



HAL
open science

Deep Learning Driven Data Processing, Modeling, and Inverse Design for Nanophotonics

Peter R Wiecha, Nicholas J. Dinsdale, Otto L. Muskens

► **To cite this version:**

Peter R Wiecha, Nicholas J. Dinsdale, Otto L. Muskens. Deep Learning Driven Data Processing, Modeling, and Inverse Design for Nanophotonics. Peng Yu; Hongxing Xu; Zhiming Wang. Integrated Nanophotonics: Platforms, Devices, and Applications, Wiley, pp.245-275, 2023, 9783527349128. 10.1002/9783527833030.ch7 . hal-04628291

HAL Id: hal-04628291

<https://laas.hal.science/hal-04628291v1>

Submitted on 28 Jun 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 Deep learning driven data processing, modeling, and inverse design for nanophotonics

Peter R. Wiecha^{1*}, Nicholas J. Dinsdale², Otto L. Muskens²

¹ LAAS, Université de Toulouse, CNRS, Toulouse, France

² Physics and Astronomy, Faculty of Engineering and Physical Sciences, University of Southampton, Southampton, UK

1.1 Introduction

The use of methods from machine learning to address problems in the design and modelling of photonic systems has grown significantly in recent years. A number of recent reviews have extensively covered the field of deep learning in (nano) photonics [1–7]. Here, we provide an introduction covering briefly some of the relevant background to deep learning neural networks, followed by a number of more in-depth applications addressing different aspects of nanophotonics research where neural networks can find application. We describe the development of a generalized predictor network for the evaluation of internal electromagnetic fields of nanostructures, which allows replacing numerical simulations in forward iterative design speeding up the design process by orders of magnitude once the network is fully trained. Next we explain the use of a tandem neural network to predict complex scattering patterns in multiport photonic integrated circuits as an example of neural network-enabled inverse design. Finally, we discuss the use of neural networks in advanced and real-time data-processing, with recent applications in optical data storage and in hyperspectral imaging.

1.2 Artificial neural networks and deep learning

Before presenting an overview of applications of deep learning to nanophotonics, we first provide a short introduction to some of the basic concepts. Deep learning (DL) is a sub-field of machine learning, which is itself a sub-field of artificial intelligence, belonging to the field of computer science. In DL, deep artificial neural networks are trained on large datasets in order to reveal complex correlations or to learn to categorize the data samples. In this context, “deep” simply means that the neural networks

*pwiecha@laas.fr

consist of several consecutive layers of artificial neurons. How individual neurons interact to form a deep artificial neural network and what are the most effective network layouts for different tasks has become a very active and fast moving topic with many significant developments made only in recent years, many of which were made through open challenge competitions [8]. The capability of deep learning networks often goes far beyond what is expected and a full understanding of how these are so effective is yet to be developed [9]. Despite the lack of a full understanding, engineers and scientists have been able to successfully apply the deep learning concepts and a wealth of practically useful principles has been established in publicly available computational toolkits.

1.2.1 Artificial neurons and neural networks

As illustrated in figure 1.1a, an artificial neuron is merely a mathematical function which reproduces the behavior of a biological neuron. If the function’s input (the stimulus) is small, the output (the activation) is also small – the neuron is inactive. If the input exceeds an activation threshold specific to the artificial neuron, it returns a high value as output – the neuron fires. In addition to the choice of activation function, an artificial neuron is defined by a set of weight parameters w_i (one for each input value) as well as by the mentioned activation threshold parameter b , called “bias”.

In principle any mathematical function which shows a neuron-like behavior can be used, however a few specific activation functions have proven particularly useful in rendering the training of ANNs fast and robust. Many tutorials on artificial neural networks introduce the “sigmoid” activation as first example of such a function, because of its direct analogy to the behavior of biological neurons. However, in practice, sigmoid neurons are usually not the first choice because of what is known as the vanishing gradients problem, which refers to the small gradient at the outer limits of the logistic function, where it is almost totally flat. This vanishing gradient of the logistic function causes the learning to slow down in many situations and imposes the requirement of careful hyperparameter tuning and sophisticated regularization techniques. A more robust and more generally applicable choice is the rectified linear unit (ReLU) activation function (or its adaptations such as the *leaky ReLU* [10]). The ReLU function is illustrated in Fig. 1.1b. It is linear if the network inputs exceed the threshold parameter b , otherwise it is zero:

$$y(\mathbf{x}) = \begin{cases} 0 & \sum_i x_i w_i \leq b \\ \sum_i x_i w_i - b & \sum_i x_i w_i > b \end{cases} \quad (1.1)$$

The kink in the piece-wise definition serves as the non-linearity, while the linear scaling in the positive region guarantees non-vanishing gradients during training also for large neuron inputs.

From the artificial neuron, it is now only a relatively simple step to the artificial neural network. An ANN is a mathematical function defined by a large number of layer-wise connected artificial neurons, thus the output of a neuron in one layer is

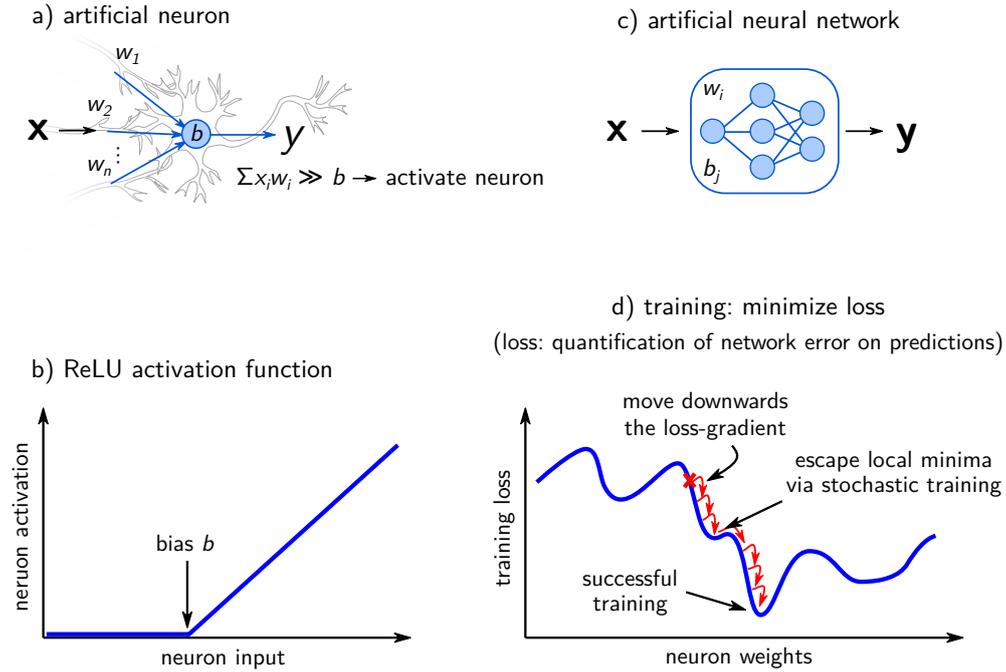


Figure 1.1: a) Sketch of an artificial neuron, taking as input a vector \mathbf{x} of values x_i and returning a scalar y , its *activation*. b) ReLU function, a possible choice for the neuron’s activation. c) simplistic sketch of an artificial neural network (ANN), composed of many artificial neurons. It takes as input a vector which feeds a set of input neurons and returns a vector \mathbf{y} , corresponding to the activations of a set of output neurons. d) training of an ANN: The network parameters are adapted by moving downwards the gradients of the loss function with respect to the network weights.

fed into the neurons of the following layer and so on, until a layer of output neurons. Modern deep ANNs can be composed of hundreds of layers of artificial neurons and can contain billions of inter-neuron connection weights [11]. We want to note here that it is important to use a *nonlinear* activation function. Otherwise, the consecutive arrangement of multiple layers in a deep neural network would be pointless, as it can be easily demonstrated that a deep neural network with linear activations can be identically represented by a single-layer ANN. Deep networks have proven to take tremendous advantage of the high number of successive layers in their process of abstraction in which the successive layers decode and capture increasingly complex correlations in the features of the input data – an advantage that would be lost using linear activations.

1.2.2 Training of artificial neural networks

Training of an ANN means to optimize the weights and bias parameters in order to make the network categorize and infer meaningful information from the input data. This is done by fitting a network on a large set of data, which implicitly contains the solution to the target problem. For instance, a network supposed to identify the species of animals in images will be trained on labeled images of those animals, where the input is the actual image and the output the corresponding species name. To fit the network and optimize its parameters, a so-called loss (or cost) function is defined, which quantifies the error of the network on predicting a set of input samples. As illustrated in figure 1.1d, using gradient-based optimization algorithms this loss is then minimized for a large set of training data, where a stochastic selection of batches of training samples in each minimization iteration avoids that the solver gets stuck in local minima of the loss. For functions of gigantic numbers of free parameters – like modern deep ANNs – these gradient-based methods can be used only thanks to numerical automatic differentiation techniques. Together with the rapid development of highly specialized computing hardware like GPUs (graphics processing units) or TPUs (tensor processing units) in the past decades, automatic differentiation was the key ingredient enabling the current tremendous success of deep learning.

A very common loss function is the cross entropy (CE):

$$L_{\text{CE}} = -\frac{1}{N_{\text{batch}}} \sum_i^{N_{\text{batch}}} y_{\text{train},i} \log(y_{\text{net},i}) \quad (1.2)$$

Here, y_{net} and y_{train} are the network output and the ground truth from the training set, respectively, and the loss is calculated via summation over a batch of N_{batch} randomly chosen training samples. The CE loss provides a particularly robust training convergence when training neural networks on classification tasks [12].

For regression tasks (for instance in physics predictor neural networks), the most commonly used loss function is the mean square error (MSE):

$$L_{\text{MSE}} = \frac{1}{N_{\text{batch}}} \sum_i^{N_{\text{batch}}} |y_{\text{train},i} - y_{\text{net},i}|^2 \quad (1.3)$$

The MSE represents the average quadratic error of the network predictions with respect to the training data. Due to the quadratic scaling, the larger the deviation the faster the MSE loss increases which helps accelerate training convergence. We note that in specific situations, other loss functions are sometimes more appropriate, we refer the interested reader to a relevant textbook for more background [13].

Only very recently, deep learning has been applied to countless problems across many scientific fields with great success. In the following section we will revisit a few applications of DL in nano-optics and nanophotonics. Specifically, we discuss two of the main applications of DL to scientific problems. The first section deals with data-driven physics predictor ANNs. In the second section, we discuss the solution of complicated inverse problems via DL, for instance in the structural design of photonic

devices or data-analysis tasks like noise-robust classification or compressive information reconstruction from sparsely sampled data.

1.3 Ultra-fast physics predictions

Physics simulations are often computationally demanding, especially in the case of systems with low symmetry and a large number of degrees of freedom. In nanophotonics, typical numerical simulations techniques are finite-difference time-domain (FDTD), finite element (FEM) or surface and volume integral methods, which take between minutes and days of computing time on state-of-the-art hardware [14, 15]. For many applications or rapid prototyping, this computational cost is severely limiting and hence simplified models are often used to approximate complex systems, even though this implies large trade-offs with respect to accuracy.

Data-driven deep learning methods promise to provide ultra-fast prediction models even in cases of highly complex photonic systems. Such a predictor is generated by analyzing a large set of expensive simulations through the training of a neural network. In this way, the ANN develops a phenomenological model from the implicit physics in the simulation data. The computational effort in this approach is outsourced to the process of the training data generation. After successful training of the ANN, its evaluation is almost instantaneous, usually in the order of milliseconds or faster. Such a different balance of efforts away from the specific evaluation stage and towards the general training stage can be beneficial in a range of scenarios, including, but not limited to, problems involving a large number of calculations or a time critical response.

It has been shown that various effects and properties in nanophotonic devices can indeed be predicted by deep ANNs. For instance, gratings or metasurfaces [16–18], waveguides [19], integrated power-splitters or other photonic devices [20, 21] but also the entire image formation in microscopes [22] have been successfully addressed using deep ANNs. Recent works on nanophotonics predictor networks have demonstrated that go much further than prediction of physical observables. By describing the predicted effects in a physical model description one can improve the fidelity of the ANNs and additionally provide insight in physical mechanisms that lead to the predicted effects. It has been demonstrated for example that instead of directly predicting the reflectivity of a grating, a neural network which predicts the optical response in a multi-oscillator model outperforms the direct prediction and is particularly strong in extrapolating outside the range of the training data [16].

A further recent concept for physics prediction is formed by so-called “physics informed neural networks” (PINNs), which do not rely on simulations for training. Instead, a loss function based on the partial differential equation (PDE) of a physical theory is implemented symbolically in the deep learning toolkit. This can be used to quantify the error of the network’s output with respect to the PDE solution without the requirement of data. Using the automatic differentiation capabilities of modern deep learning toolkits, the physics PDEs can be numerically evaluated very efficiently and a network can be trained directly on their solution. This has been proven to allow

extremely precise ANNs with very low prediction errors, on problems in, for example, heat diffusion, fluid mechanics or seismology [23–26]. However, since the training directly implements the PDEs, the solution is limited to the PDE, which may not yield the measured physical observable. In addition, the boundary conditions are generally fixed which means that each configuration or geometry requires training of a new model [26]. This obviously limits the flexibility of PINNs quite drastically. Efforts to overcome this limitation are so far usually accompanied by a significant reduction of the network accuracy [27], which is why, in situations close to real-life applications, simulations-based ANNs are typically used.

1.3.1 Specialized physics predictors: Fully connected vs. convolutional ANNs

Leaving aside PINNs, in most implementations of physics predictor networks the ANN is trained on a selected physical observable. Examples are transmission and/or reflection spectra of metasurfaces [28–30], power-splitting ratios in photonic devices [20] or extinction cross sections of nanoparticles [31]. This approach has been exhaustively applied to various optical properties of nanophotonics systems [2, 4, 6, 7]. In general, the more specific a physics problem is, the better a neural network can learn its prediction.

In many works, so-called fully connected neural networks (FCNNs) are used, in which abstract geometrical parameters like sizes, positions or numerated materials are used as input to the network. However, since the publication of the so-called “AlexNet” in 2012 [32], convolutional neural networks (CNNs) have attracted tremendous attention and have proven to outperform FCNNs on virtually any task. Unlike FCNNs, CNNs take as inputs higher-dimensional data. The most common form, for instance, is the 2D-CNN which takes images as input. Instead of connecting every input value to every neuron like in the case of FCNNs, only the convolutional kernels of the CNN consist of artificial neurons that are subject to learning. In consequence, CNNs usually have far lower numbers of neuron-connections and thus lesser free parameters. This enables very deep network layouts, consisting of up to several hundred layers of neurons. Enabling very deep network designs is the main reason for the superior performance of CNNs. Additionally, the image-like representation of the input data is often more easily understood by an ANN – compared to abstract geometric parameters, which need to be decoded additionally before the ANN can start to do the physics interpretation. Furthermore, CNNs implicitly learn translational invariance – hence the fact that an image shifted by a few pixels still represents the same image. This is not the case in FCNNs, where it needs to be learned, resulting in the requirement of larger training sets.

In order to demonstrate the performance difference between FCNNs and CNNs, we develop here a simple toy problem, aimed at predicting the extinction cross section spectra of silicon nanostructures under linear polarization plane wave illumination. As illustrated on the left in figure 1.2a, the nanostructures are composed of 2 or 3 silicon cuboids of lengths l_i and widths w_i , both varying between 120 nm and 400 nm, and placed at positions (x_i, y_i) on a $2 \times 2 \mu\text{m}^2$ large area. The sizes and positions are discretized by steps of 40 nm to allow a simple 2D representation in the CNN.

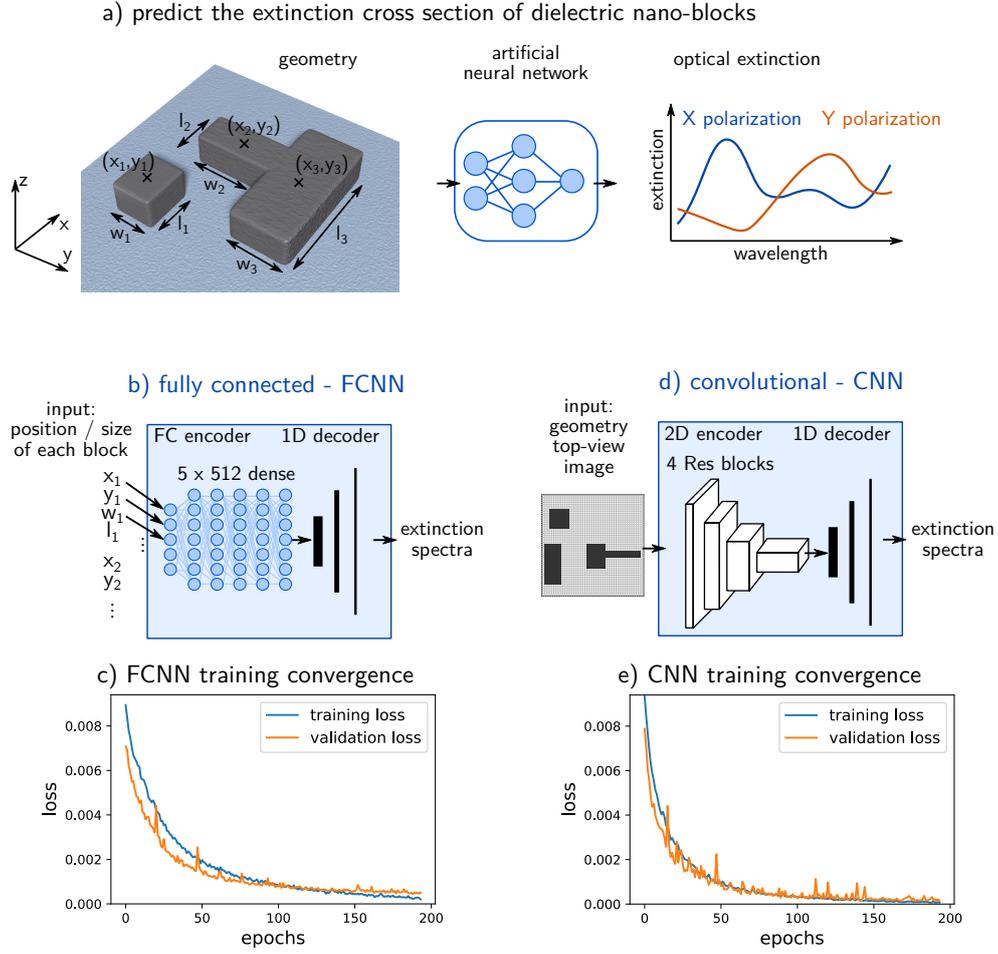


Figure 1.2: a) Sketch of the extinction cross section predictor neural network. The geometry of a dielectric nano-structure composed of multiple cuboids of size (w_i, l_i) (fixed height) and positions on a plane (x_i, y_i) is fed into an ANN which returns the optical extinction spectra for X- and Y-polarized plane wave illumination under normal incidence. b) fully connected neural network with ≈ 1.5 million free parameters, taking as input the geometrical design parameters and c) its training convergence. d) convolutional neural network with also ≈ 1.5 million free parameters, taking as input the top-view image of the geometry and e) its training convergence.

The height is set constant to $h = 160$ nm and overlapping cuboids are fused. For the training set, we generate 10000 random cuboids of which we use 9000 for training and 1000 for validation and benchmarking. We simulate the extinction spectra using

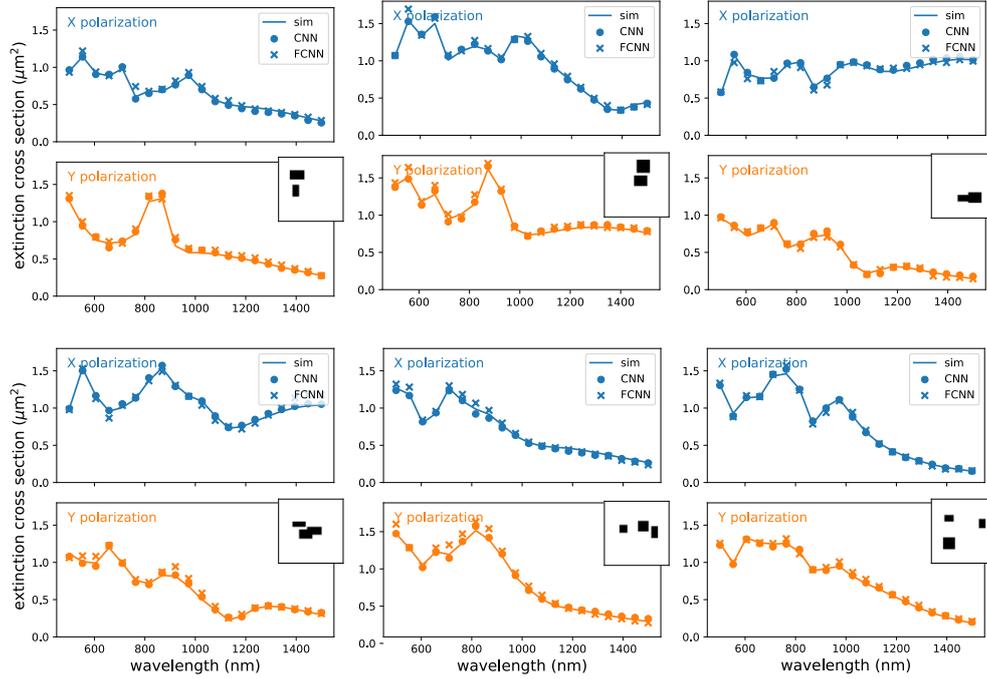


Figure 1.3: Comparison of simulations (solid lines) and neural-network predicted extinction cross sections for random nanostructures. Cross markers correspond to the FCNN predicted extinction, filled circular markers to the CNN. Blue and orange curves show the response for X , respectively Y polarized illumination (plane wave at normal incidence). Insets show the top view of the respective geometries on $2 \times 2 \mu\text{m}^2$ large areas.

our homemade toolkit “pyGDM” [33]. Our benchmark problem consists of predicting the optical extinction spectra simultaneously for X and Y polarized illumination, as illustrated in figure 1.2a. Using these data, we now compare the FCNN with the CNN network architecture. We design the two ANNs such that they are composed of an approximately identical number of artificial neurons (≈ 1.5 million fitting parameters each). We use leaky ReLU activations in all layers except for the output layer which is linear. Furthermore, to avoid overfitting and to accelerate the training convergence, we apply batch normalization after each layer [34]. The spectral reconstruction parts (“1D decoder” in figure 1.2) are also identical in both cases and are composed of a 1D convolutional ResNet. We thus evaluate only the impact of different network layouts in the “information extraction” part of the ANNs. This is the region following the input layer, trying to “understand” the meaning of the input data.

The first architecture, the FCNN (figure 1.2b), takes as input the positions (x_i, y_i) and sizes (w_i, l_i) of four silicon blocks. If a block is not present, the according param-

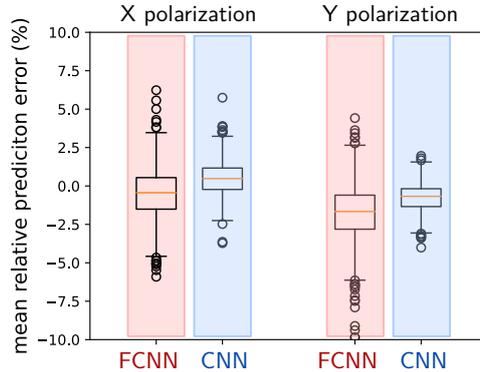


Figure 1.4: Statistics of the network predictions on 1000 random structures which were not included in the training set for X (left) and Y polarization (right). The red highlighted box plots are the statistics for the fully connected network, the blue highlighting corresponds to the convolutional neural network.

eters are set to -1 , so the network can learn that a negative block size means actually “no block”. Note that we normalize all input and output data to a range between 0 and 1. Following the $3 \times 4 = 12$ input neurons, the main part of the FCNN is composed of 5 dense layers, each of which is 512 neurons large. The last of these dense layers is connected to the 1D CNN spectra decoder. The training convergence is shown in figure 1.2c.

The convolutional network takes as input a 2D array which is, in our case, the discretized top-view of the nanostructure on a 50×50 grid with 40 nm stepsize (see left of figure 1.2d). The CNN is composed of four residual blocks (Res-blocks) each implementing 3 consecutive convolutional layers of 16, 32, 64 and 128 kernels and a skip-connection through an additional convolution [35]. After each of the first three Res-blocks, we perform a 2×2 max pooling to reduce the input data from 50×50 down to 6×6 , before injection into the upsampling 1D-CNN decoder. The training convergence is shown in figure 1.2e. We observe that the CNN converges faster and to lower loss values compared to the FCNN.

Note that with approximately the same number of artificial neurons, the CNN can implement $4 \times 4 = 16$ convolutional layers, against 5 dense layers in the FCNN. Increasing the number of layers in the FCNN while reducing the number of neurons in each layer does actually deteriorate its performance. This trend suggests that increasing the depth of the FCNN beyond a certain point results in a reduced efficiency of learning, which is a known problem in FCNNs. The skip-connections in the Res-blocks avoid this problem and allow the training of very deep architectures [35].

Figure 1.3 shows 6 random examples of predictions from the FCNN (cross markers), the CNN (dot markers) and the corresponding full-field simulation (solid lines). X and Y polarization are shown in the top (blue lines) and bottom panels (orange lines), respectively. The top view of the nanostructure is shown in the insets in the Y -

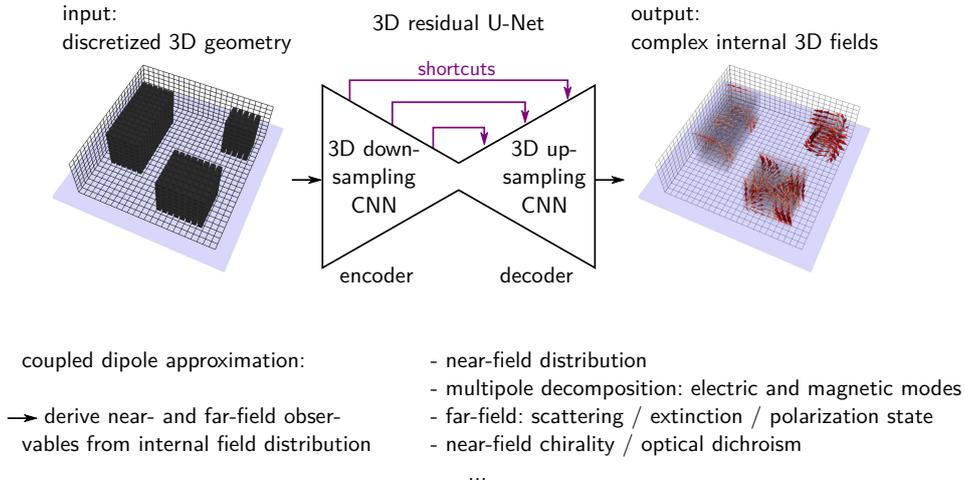


Figure 1.5: Simplified scheme of the generalized nano-optics predictor network concept. A 3D fully convolutional U-Net takes as input a volumetric discretized representation of a nanostructure of arbitrary shape. It returns 6 channels of the same dimensionality as the input, predicting real and imaginary part of E_x , E_y and E_z inside the nanostructure. The internal fields can be used in the framework of a coupled dipole approximation to calculate basically all possible near- and far-field effects.

polarization subplots. First of all, we find that both ANNs provide good predictions for the multi-cube nanostructures. However, the predictions of the CNN are consistently slightly closer to the numerical simulation. Figure 1.4 shows a boxplot containing statistics of the predictions of 1000 random nanostructures, from the FCNN (red highlighted) and from the CNN (blue highlighted). The statistics confirm the globally better performance of the CNN. Interestingly the FCNN tends to underestimate the extinction, while the error of the CNN is centered around zero.

1.3.2 Generalized nanophotonics predictor network

As discussed above, artificial neural networks are used in many recent examples as surrogate models to predict specific physical quantities such as the optical extinction in the above example, nearfield enhancements [36], or the reflectance or transmittance of metamaterials [28]. Recently we demonstrated how a more generalized predictor network can work [37]. When the interaction of light with a nanostructure is numerically described in a coupled dipole approximation, the volume of the structure is discretized on a regular grid of which each meshpoint is considered to interact with electromagnetic fields like an oscillating dipole [38]. In this approximation, once the electric field inside the discretized structure is known, so is the dipole moment of each

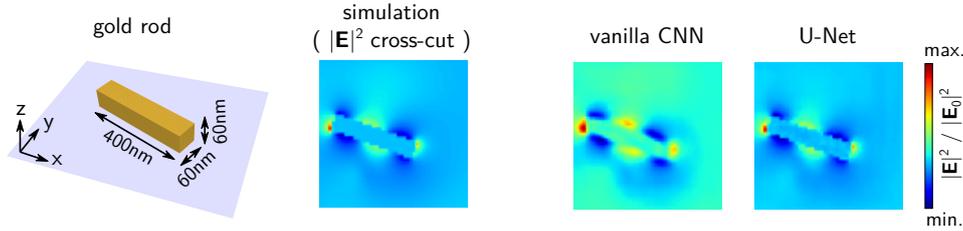


Figure 1.6: Vanilla CNN vs. U-Net illustrated by the example of near-field prediction for a plasmonic nano-rod. The gold rod of $400 \times 60 \times 60 \text{ nm}^3$ size is in air on a glass substrate ($n_s = 1.45$), illuminated by an X -polarized plane wave of $\lambda_0 = 700 \text{ nm}$. Simulation and predictions show the electric field intensity on a cross-cut through the rod center, parallel to the XY plane. In comparison to the vanilla CNN, the short-cut connections in the U-Net allow the network to bypass the information about the geometric shape around the network’s bottleneck layer. This helps significantly in the correct reconstruction of the sharp rod borders and in consequence leads to a far better prediction of electric field hot-spots in the close nearfield.

discretization meshcell. Because the electromagnetic emission behavior of an oscillating dipole is perfectly known, the electromagnetic field at any position in space can be calculated from the distribution of oscillating dipole moments [39, 40]. This allows the calculation of, for example, electric [39] and magnetic near-fields [41], the far-field radiation pattern [42], a multipole decomposition of the optical response [43], optical forces [44], optically induced heat generation [45] or the dynamic charge distribution in plasmonic nanostructures [46]. The method can also be generalized to periodic structures [47, 48]

An artificial neural network capable of approximating the discretized internal electric field of a nanostructure upon interaction with light represents therefore a generalized nano-optics predictor, since the fields inside can be used to obtain the effective dipole moments of the cells of a discretization grid. This concept is illustrated in figure 1.5 and is realized in Ref. [37] by a three-dimensional fully convolutional U-Net [49] consisting of residual blocks. The U-Net concept includes shortcut connections between the downscaling and upsampling residual blocks, which allow to bypass high-resolution spatial information around the bottleneck in the center of the network. U-Nets are therefore often applied in image segmentation [49]. As illustrated in figure 1.6, they are beneficial for the electromagnetic field prediction, helping to accurately reconstruct the sharp borders of the nanostructure geometry, which improves for instance the prediction of strongly localized near-field hot-spots.

The generalized predictor achieves accuracies of a few percent on various derived physical observables such as near-field intensities, scattering cross section or scattering polarization state [37]. However, a similarly complex neural network explicitly trained on a specific observable instead of the complex internal fields will most likely

outperform the generalized predictor network in terms of accuracy. This is, on the one hand, a result of the error propagation when calculating a physical effect based on the predicted internal field distribution. On the other hand it is due to the trend in deep learning that highly specialized ANNs perform better than networks trained on more general problems [12]. However, the slightly worsened accuracy comes with a tremendous gain in flexibility and, in many cases, an error of a few percent is still tolerable. In particular, for complicated inverse design problems, a perfect accuracy of the prediction might not be of the highest importance as long as the method is capable of finding a close-to-optimal solution.

1.4 Photonics inverse design

Applications of the deep-learning driven inverse design to nanophotonics have recently received a great deal of attention. Inverse design – the problem of finding a nanophotonics device or structure which implements a specific user-defined functionality – is a highly difficult problem. The most prominent challenge is that the solution to an inverse problem is generally not unique. This means, for instance, that several configurations or nanostructures can lead to the same optical response. Because the problems are usually ill-posed, analytical methods fail in their solution, which is why deep learning is currently receiving immense attention, since it is capable to approach these problems efficiently. Applications of ANN inverse design in nanophotonics range from individual nanostructure geometries [31, 50, 51] over metasurface inverse design [17, 52–55] to the design of photonics devices such as waveguides, gratings, grating couplers, power-splitters, periodic arrays, etc. [21, 56–59].

1.4.1 Predictor network as a surrogate model for optimization

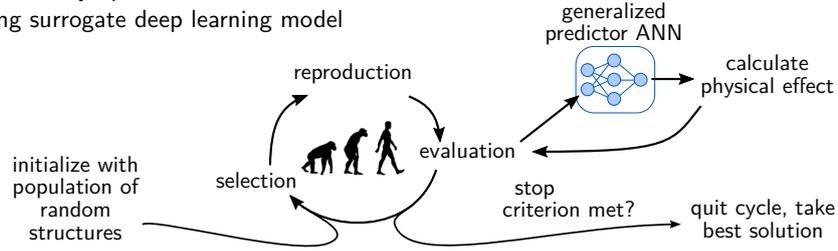
Two distinct approaches for deep learning inverse design can be discerned. The first technique is based on exploiting the extremely fast evaluation speed of physics predictor neural networks, which usually offer accelerations of many orders of magnitude compared to numerical physics simulations.

The ultra-fast physics predictor ANN is coupled to a global optimizer such as evolutionary optimization (EO) or genetic algorithms [56, 58]. In principle, this technique can employ all common global optimization methods [60] while gaining tremendous acceleration through the ANN surrogate model. A practical advantage of the surrogate model optimization approach is the possibility to replace the neural network predictor by the full numerical simulations for occasional verifications of intermediate solutions and / or once the ANN-accelerated optimization has converged [61]. In this way, the ANN serves as a rapid tool to find a good initial guess while the slow conventional simulations methods lead to the very optimum solution. Another approach to deal with the approximate character of the ANN predictions has been proposed by Jiang et al. with the “GloNET” model [57]. The GloNET is directly trained using a physics-simulation based loss function where the gradients for error backpropagation in ANN training are obtained by adjoint simulations. Therefore the method requires

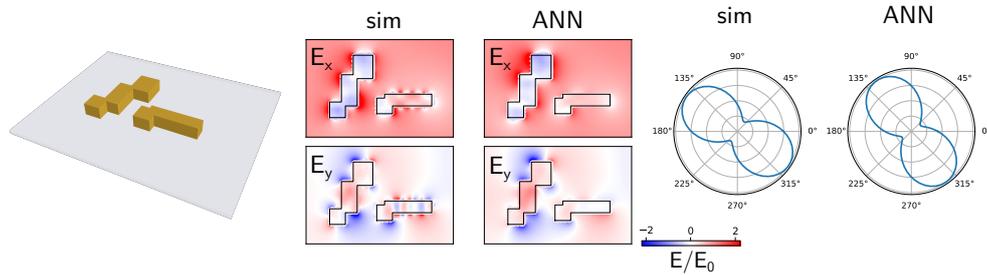
1 Deep learning driven data processing, modeling, and inverse design for nanophotonics

a) evolutionary optimization

using surrogate deep learning model



b) maximize polarization conversion by gold structure



c) maximize magnetic near-field enhancement above silicon structure

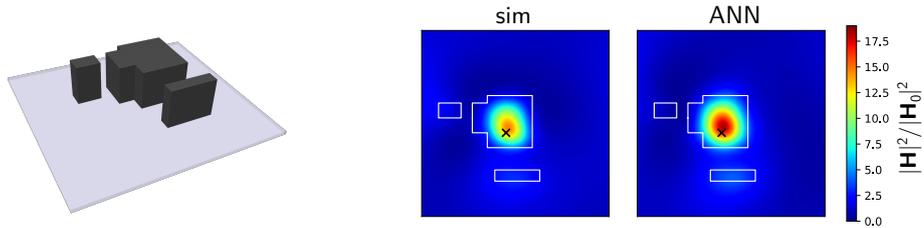


Figure 1.7: a) scheme of coupling a physics predictor neural network to an evolutionary optimization algorithm for inverse design of photonic structures. Using the generalized nanophotonics predictor, arbitrary nanophotonic target effects can be optimized without re-training of the neural network. b) example of an inverse designed plasmonic gold nanostructure for conversion from X - to Y -polarized field, using the generalized predictor network for internal field approximation. The polarization conversion is then calculated in a second step from the internal field distribution. c) example of the design of a silicon nanostructure for magnetic field intensity maximization. The black cross indicates the target position. Again the internal fields are obtained by the generalized predictor during optimization, and the magnetic field intensity is then derived from this prediction.

no pre-simulated dataset and the ANN training procedure corresponds instead to the iterative optimization process during which the numerical simulations are executed on-the-fly. On the other hand, this means that each inverse design requires a full training run. However, the GloNET implicitly optimizes its design strategy for each given problem during network training and therefore converges with less simulation evaluations compared to other heuristic optimizers. We note that recently Jiang et al. demonstrated that a multi-objective description of the problem can be used to generalize the GloNET concept to solve a broader spectrum of problems concurrently in a single training sequence [62].

In addition to the problem of introducing approximation errors by the use of surrogate ANN models, most physics predictor neural networks are highly specialized on a specific physical effect (see section 1.3.1). In nanophotonics, the generalized predictor ANN [37] can be used as a flexible alternative to training separate neural networks for each design target. As illustrated in figure 1.7a, in combination with a global optimizer like EO, the generalized predictor is used as a simulation surrogate where its prediction of the photonic nanostructure’s internal field distribution is used in an intermediate step to calculate the actual target physical observable. While the approach introduces a numerically slower post-processing step of the internal fields, the slowest part – the main physics simulation – is still replaced by the ultra-fast ANN model, allowing total accelerations of a factor of 20-30. Most importantly, the generalized predictor is highly versatile and can be used without re-training for the inverse design of countless nano-optics functionalities.

Example: Polarization conversion maximization

As a first demonstration, we derive the polarization state of scattered light in the far-field from the generalized predictor model of a gold nanostructure. The wavelength of the plane wave illumination is $\lambda_0 = 700$ nm and the polarization is linear along X under normal incidence. We feed the polarization state into an evolutionary optimization algorithm (for details on the optimization see Ref. [42]) with the goal to maximize the ratio of the Y -polarized electric field component over the X -polarized component. Consequently, we aim to obtain a plasmonic nanostructure with a high polarization conversion ratio. Figure 1.7b shows the optimized gold nanostructure after 200 EO iterations of a population of 50 structures. The ANN-driven optimization found an arrangement of L-shape type nanostructures, which corresponds to the type of structures commonly used for plasmonic polarization conversion applications [63–65]. When comparing the optical nearfield in and around the gold structure (Fig. 1.7b center panels), we observe that the geometry introduces indeed a conversion of the X -aligned incident linear polarization into a Y -oriented field of significant amplitude. The right panels of Fig. 1.7b show that this Y component is also effectively radiated to the far-field, leading to an elliptic polarization with a polarization angle of approximately 45° (in the polar plots 90° correspond to the Y direction). However, we observe a quite large quantitative error in the optimization target, which is derived from the ANN prediction. While the ANN-based optimizer found a Y/X polarization ratio of around 1.5, the full-field simulation of the final structure reveals that the true Y/X ratio is in fact

only around 0.7. This quite large discrepancy is probably a result of a disadvantageous error propagation in this specific case. Nevertheless, the general trend is correct and the structure may serve at least as a very good initial guess either for simulation-based optimization or for manual tuning of the geometry.

Example: Maximize magnetic near-field

As a second example, we aim to find a silicon nanostructure that maximizes the magnetic near-field intensity 50 nm above its top surface. The magnetic near-field can be obtained from the internal electric field using the according dyadic Green’s tensors [41, 66]. We use again the generalized predictor network which approximates the internal field distribution inside the nanostructure, which now serves to derive the H -field intensity at a fixed location. The illumination is again an X -polarized, normally incident plane wave of wavelength $\lambda_0 = 700$ nm. The ANN-driven EO algorithm finds a nanostructure that does induce a magnetic near-field hotspot at the target position (see Fig. 1.7c). Again the numerical simulation confirms, in general, the optimization result, however with some discrepancy of $|\mathbf{H}|^2/|\mathbf{H}_0|^2 \approx 14$ vs $|\mathbf{H}|^2/|\mathbf{H}_0|^2 \approx 19$. Still, the found enhancement is comparable to formerly reported values in optimized nanostructures [67].

The reason why both examples display discrepancies of far more than the average 5% error of the generalized predictor network [37] is not only a result of unfortunate error enhancement during the calculation of derived physics observables. A key role is played by the optimization itself. It is important in this context to realize that the 5% ANN error is the statistical average, there are many cases in which the ANN error is significantly higher. The optimization algorithm naturally tries to push the structures to the limits of the parameter space, when it maximizes a response. Typically in these regions, the prediction error is higher than average. Therefore, the optimization results need to be validated with particular care. We note that the problem can be alleviated at the cost of additional computational expense by occasionally performing regular full-field simulations in the evaluation step of the EO algorithm [61].

Despite the undeniable drawback of limited accuracy, the flexibility of the generalized predictor-based optimization can be very useful for rapid prototyping and, in particular, for the inverse design of multi-functional nanostructures, which optimize several physical effects at a time. Without any re-training or re-calculation of training data, the approach can be directly applied to any physical observable as well as on multi-objective optimization problems by formulating the problem accordingly and by using appropriate global optimization algorithms [68].

1.4.2 Direct inverse design networks

The inverse design using ultra-fast physics predictor ANNs as surrogate models for computationally expensive simulations requires countless evaluations of the predictor and coupling to a complex optimization algorithm. Furthermore, the “no free lunch” theorem for optimization states that the choice of the optimizer algorithm can be a crucial component in the workflow and can drastically limit the performance of the

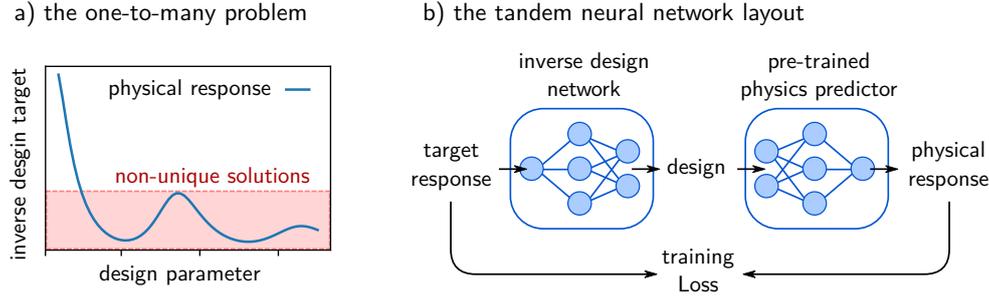


Figure 1.8: a) Illustration of the *one to many problem* in inverse design. A target property can often be realized by different designs, hence the solution of the problem is non-unique, which poses problems to a straight-forward neural network implementation. b) The *tandem neural network* layout as a typical approach to solve the one to many problem. Rather than using the design geometry, it is based on a loss function of the physical response, predicted by a pre-trained forward network.

method [69]. Thus, while straightforward to implement, the ANN+EO technique is computationally not fully optimized. A very exciting other approach using deep learning is offered by the possibility of training a neural network directly on solving the inverse problem, without detour via an iterative approach. This potential of ANNs for direct inverse design is in part responsible for the great attention which is currently placed upon artificial intelligence applications in various scientific fields.

However, the direct inversion using a naive network implementation works only on very simple problems, where for each design target only a single solution exists. A few such examples have been reported in the literature [20, 70], however these cannot be adapted to even slightly more complicated problems or design models. The reason why in general the training of a standard neural network on the inverse problem is not working without more sophisticated techniques is that inverse design problems are typically so-called *one to many problems*, hence several different designs exist that yield the desired effect [71]. This is illustrated in figure 1.8a, where the y -axis indicates the design target (*i.e.* a physical observable) and the x -axis corresponds to the design parameter (like a geometrical configuration). Generally, several design parameters can yield the same physical response of the system, thus the problem is non-unique. Training an ANN directly on such problems yields unstable training convergence because the loss function is not well defined as it can take both, small and large values for the same inverse design target, depending on the training data-sample [72].

The main challenge is therefore to stabilize training in cases of non-unique problems. Several approaches have been suggested to this end. One of the most frequently used concepts is the so-called *tandem neural network* [17, 72, 73], which is schematized in figure 1.8b. In this architecture, a forward solver is trained in a first step, which

predicts the inverse design target from the design. This is a well-defined problem with unique solutions. In the second step, the inverse network is trained, but rather than using the inverse design geometry in comparison with the training data to calculate the training loss, the inverse design ANN is trained using the physical response for the loss function. To assess the physical response of the network’s inverse design suggestion, the pre-trained forward network is used as a surrogate model. Usually, the pre-trained network is fixed and not subject to training during the second step, but it has been shown that it is also possible to train forward and inverse networks simultaneously [28,31].

While we will not go into detail, we still want to mention that several further network architectures exist to approach one to many problems, such as nanophotonics inverse design. Using so-called mixture density or probability density networks which return probability distributions of the design parameters, different possible designs that lead to the same response can be predicted simultaneously [74,75]. Conditional autoencoders (cAEs) or conditional generative adversarial neural networks (cGANs) add additional degrees of freedom, the so-called latent variables, to the target response (the “condition”), to render each non-unique solution different. In contrast to the supervised part of the data (the “condition”, for which training data exists), the latent variables are trained in an unsupervised manner (hence are “freely chosen” by the optimizer during training). This effectively transforms the one to many problem into a well-defined, unambiguous problem [76–78]. A similar approach has been demonstrated using invertible neural networks which learn the forward and inverse problems concurrently [79].

1.4.3 Optimizing inverse design performance

While deep learning-based inverse design offers tremendous advantages in terms of computation speed, the quality of the structures obtained by inverse design is so far often not ideal. As already discussed above (see section 1.4.1), this can be a result of various problems such as convergence to a point of low accuracy in the ANN representation or singular points in the neural network, but also convergence problems in the simulations used to generate the training data will play a role. Important efforts are therefore spent to develop techniques that improve the inverse design fidelity of deep learning approaches.

In the following section we want to illustrate some of these approaches on the specific example of photonic device inverse design via a tandem neural network. We aim to design a multi-mode interference device (MMI) which routes light from one or multiple inputs to several outputs. Here we will illustrate the techniques for inverse design improvement on the most simple example of a 1x2 MMI (one input, two outputs, see figure 1.9a). The MMI is patterned with etched holes of $750 \times 750 \text{ nm}^2$. Using appropriate patterns of these perturbations, the light routing behavior of the MMI can be controlled. We want to note that an important advantage of tandem neural network inverse design is its inherent capability to treat multi-objective problems. Without any major modification, the same ANN approach used for the single-objective 1x2 MMI design can be applied to the inverse design of, for instance, 3×3 MMIs (3 inputs and

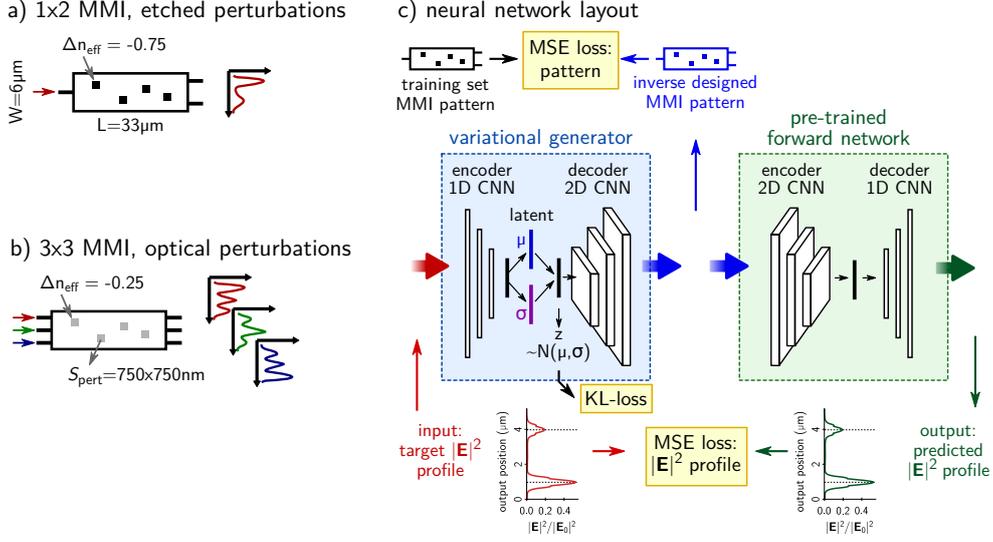


Figure 1.9: MMI inverse design via tandem neural network. a) 1x2 MMI model with etched holes as perturbations to control light routing. b) A more complex 3x3 MMI model with optically induced perturbations to control the light flow [80]. c) The tandem network layout used to inverse design MMI perturbation patterns that achieve specific light routing control. A pre-trained forward network predicts the light routing of MMI patterns which is used to calculate an unambiguous loss function, capable of effectively training the inverse ANN.

3 outputs, see figure 1.9b). In the latter case three possible input states need to be optimized concurrently, for details see Ref. [80].

In order to find perturbation patterns that control the light-flow through the MMI in a desired way, we use the tandem network shown in figure 1.9c. A pre-trained forward network (green box) predicts the light routing of an MMI patterns, this physical attribute of the MMI is used to calculate an unambiguous loss function, capable of effectively training the inverse ANN. Furthermore, we also add a loss which compares input and output geometry with a small weight of 1:50 compared to the pre-trained ANN loss. This pattern-loss teaches the network to favor the specific solution given by the training data, since during training we can be certain that the simulated sample represents the exact optimum. Finally, we add another 1:50 weighted loss, the Kullback–Leibler divergence (KL-loss). This makes the inverse network a “variational” generator, and ensures that the compressed latent representation of the physical response in the generator is following a standard distribution, and hence physically similar responses are clustered in the latent space. We will not go more into detail about the network itself at this point, so for all network and training details please see Ref. [80].

Optimizing the network layout

In general, the optimization of a network layout is relatively straightforward. Usually, one should follow the design guidelines of recent literature. In our case, we choose a fully convolutional ANN over a dense network since the representation is more natural and it allows to train a deeper network. We use batch normalization instead of dropout for regularization and use leaky ReLU activations throughout the network except for the output layers for which we use sigmoid functions. We do the latter because both networks (MMI inverse design ANN and forward predictor) yield data in well-defined ranges: the binary perturbation pattern has values of either 0 or 1 and the physics predictor network returns the electric field intensity profile across the output waveguides, which is normalized to the input field and hence cannot exceed one or become negative. By using sigmoid neurons, the network is restricted to the range between 0 and 1 of physically possible results, which facilitates the training procedure.

Once an architecture like our tandem network is chosen and implemented, configuration parameters are manually tried until training starts converging. From this first rough network design, the most crucial parameters like the number of layers and kernels or the learning rate can be tested and optimized systematically, for instance, via a grid scan, but even simple manual testing often leads to a satisfying performance.

Quality of the initial dataset

Much more important than fine-optimizing the ANN hyperparameters, in many cases, is the quality of the training data [7]. In our example of designing MMI perturbation patterns for controlled light routing, we found that a dataset of purely randomized patterns contains very few devices with significant splitting ratios. In other words, MMIs that route the light specifically towards one output port. In fact, random patterns mostly lead to high losses due to scattering out of the device. Consequently, the ANN trained on a purely random dataset learns to predict the low transmission and weak splitting ratios of those patterns but fails on the prediction and inverse design of “useful” MMIs. A neural network that is supposed to inverse design actual light routing devices requires training data containing suitable examples.

To solve this issue with the ANNs in our patterned MMI design, we use a randomized, iterative optimization scheme in order to generate a large variety of patterns that contain mainly MMIs with high splitting ratios. To this end, we first define a fitness function that is maximum if the target transmission values are met at all ports, as illustrated in the top plots of figure 1.10. Fig 1.10a shows the fitness function as a function of the two output transmission values when light routing towards the top port is targeted. Fig 1.10b shows the case of a target splitting ratio of 22:68. After generating a random set of target transmission values for each output port, perturbations are iteratively added at random positions and the MMI is simulated after each new perturbation is added. If a perturbation leads to a higher fitness value, hence an improvement in the optimized pattern, it is kept. If the pixel does not improve the fitness value, it is discarded. We continue adding perturbations either until no further improvement can be made after testing each available position or until a defined maxi-

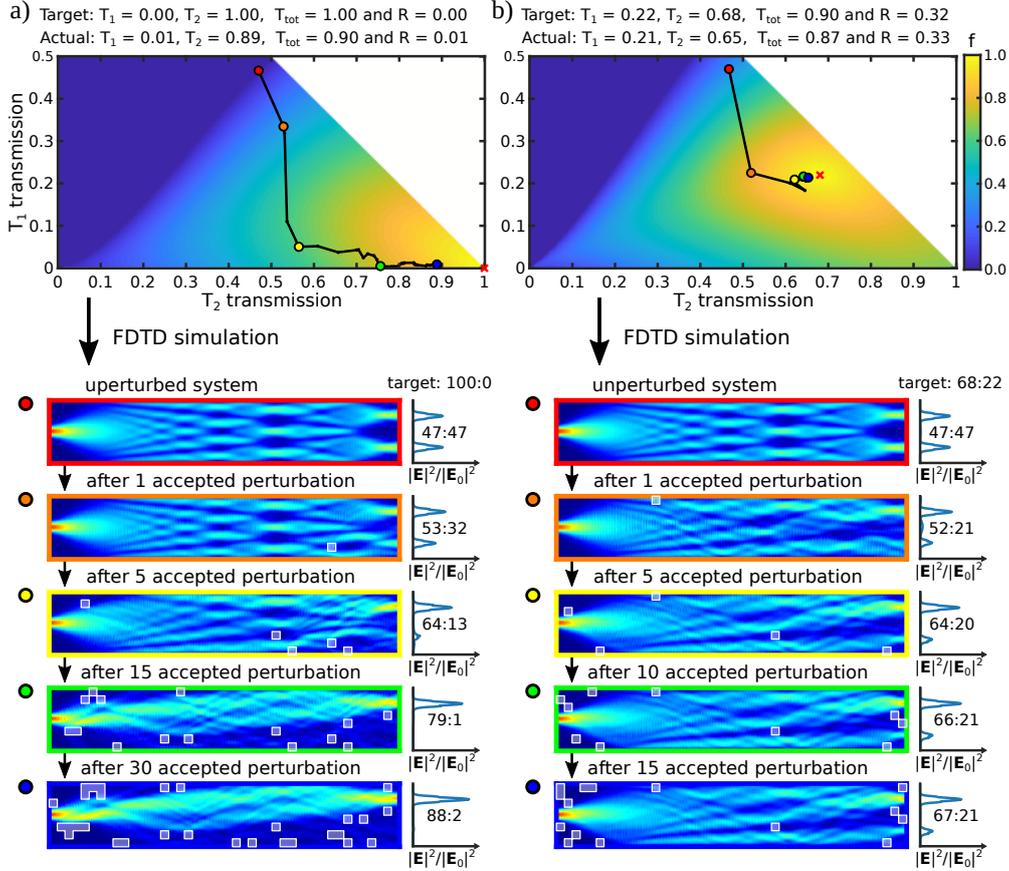


Figure 1.10: Example of initial dataset generation for optimizing coupling to a) a single output port and b) a specific target splitting ratio, R . The top section shows the iterative perturbation pattern optimization process using an aperiodic Fourier Modal Method model. The colored circles indicate the FDTD simulated patterns below, which show the respective perturbation pattern and its corresponding electric field amplitude distribution map as well as the output intensity profile (right columns) [80].

imum number of perturbations is reached. The positions of the consecutively improved patterns in the fitness landscape are shown in the respective top plots of Fig. 1.10 along a black line. Selected MMI patterns and their respective absolute electric field amplitude are shown below the optimization space in Fig. 1.10 and are indicated by their frame color in the fitness plots.

Using this optimized set of training data, the ANN finally does learn to correctly predict low-loss and arbitrary splitting ratio MMI devices. This shows that high-quality data is probably the most important ingredient, which holds for most deep

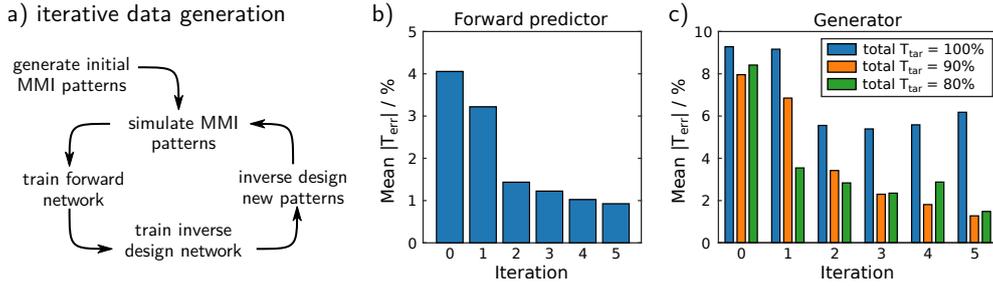


Figure 1.11: Effect of iterative training data generation on a 1×2 MMI inverse design network. a) Scheme of the iterative training procedure, where the inverse design network is used to generate new samples which are being simulated and incorporated in the training data. b-c) Mean absolute transmittance error between the ANNs and simulations for the consecutive data-generation iterations, evaluated using a fixed test dataset. b) Forward network error from prediction against simulation. c) Generator network error of design-targets against simulations of inverse-designed patterns.

learning applications.

Iterative training

When benchmarking the ANNs which are trained on the initial dataset described above, we obtain errors in the order of 5% for the forward network and 8 – 10% for inverse-designed MMIs. While this means that the method works in principle, it is still far from ideal to concede such large error margins. A closer inspection of the predictions reveals that the network underestimates scattering losses in many cases and that the inverse design ANN often suggests patterns that barely transmit any light. This is a symptom of having over-optimized the training dataset. It contains almost no lossy or truly random patterns, hence the network assumes that every MMI has high overall transmission.

A possible approach to improve the performance of inverse design neural networks is to iteratively generate more training data by re-simulating designs produced by the generator ANN, appending those samples to the data and by subsequently retraining the network on the enriched dataset [80, 81]. In this way, the network is trained on the correct simulations of formerly incorrect predictions. A scheme of the concept is shown in figure 1.11a. Fig. 1.11b and 1.11c show the improvement of the forward and inverse network, respectively, over 5 iterations of training data generation. The iterative process helps considerably to improve the prediction accuracies for the forward as well as for the inverse design network. The high remaining inverse design error for targets with 100% total transmission (blue bars in Fig. 1.11c) can be explained by the physical impossibility of reaching lossless transmission with the chosen MMI model.

We note that it has been recently demonstrated that a progressively growing neural network layout complexity together with iterative training-data generation can boost accuracies even further [82].

Post-processing

Depending on the inverse design problem, post-processing of the ANN output may be necessary. Since the neural network requires differentiable neuron activation functions, we naturally obtain a greyscale image of the inverse design MMI perturbation maps (as shown in the top panel of figure 1.12a). Since, in our case, the perturbations are binary (hole or no hole), these greyscale images need to be converted into patterns of zeros and ones. Hence the application of a threshold step is necessary, which raises the question of how to choose the best possible threshold. As shown in figure 1.12a, the wrong threshold can have a significant impact on the performance of the device. We also see that low threshold values are usually a good choice, since the pattern already has a high contrast. This is a consequence of our choice of sigmoid activations for the output layer of the ANN, which helps to increase the contrast and therefore reduces the impact of the threshold over a relatively large range, as shown in figure 1.12b (blue line).

We note that instead of a fixed threshold, it is possible to use the pre-trained forward network to test MMI patterns for several threshold values. In that way, it is possible to dynamically determine the best threshold value for each single inverse design [80].

1.5 Advanced data-processing for photonics applications

In this final section, we want to provide a brief insight into another important application of deep learning in nanophotonics, namely the analysis and interpretation of complex data. Deep learning data analysis can be applied to experimental results as well as on numerical simulations. Results from simulations are often equally difficult to analyze or interpret, for which deep learning based methods have emerged as promising tool. For instance, dimensionality reduction techniques have recently been used for knowledge discovery from simulated data [83–85]. However, we want to present here briefly two examples of ML for the analysis of experimental data.

1.5.1 Optical data storage below the diffraction limit

As first example, we discuss an application of ANNs for the recognition and classification of experimental optical spectra from individual nanostructures. The goal is to retrieve digital information, encoded in bar-code-shaped silicon nanostructures [86]. In contrast to conventional optical storage, where a single bit of information (0 or 1) is encoded in a diffraction limited area, the silicon nanostructures of the size of a diffraction limited focused spot are designed such that they encode several bits of information in their geometry. This is illustrated on the left of figure 1.13. The structures are furthermore designed such that under polarized illumination, each nanostructure representing

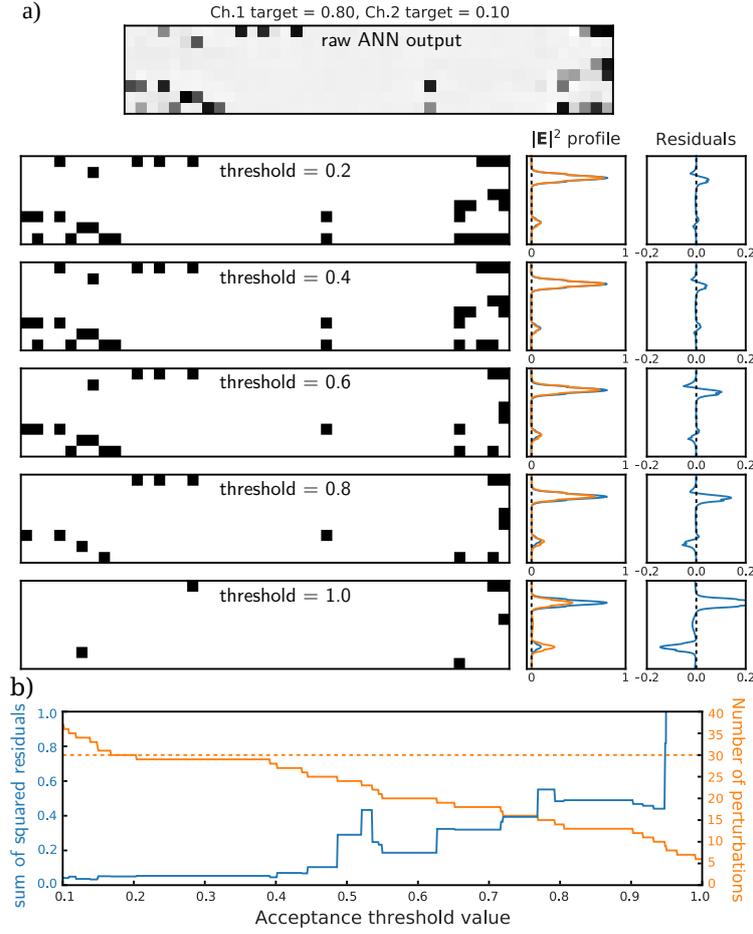


Figure 1.12: Example of the effect of perturbation acceptance threshold on the inverse design pattern. a) The predicted perturbation patterns (left column), the forward network predicted intensity profiles (central column) and the residuals between the prediction and target (right column) for a selection of different perturbation acceptance threshold values. The target intensity splitting was set to 80% and 10% in channels 1 and 2. b) The sum of squared residuals between the prediction and target for acceptance threshold values between 0.10 and 1.00 in steps of 0.01 and the corresponding number of predicted perturbations. The orange dashed line represents the maximum number of perturbations considered in the initial training dataset.

a certain bit-sequence scatters visible light with a unique spectral signature. Through an analysis of the scattering spectrum, the information can in principle be retrieved

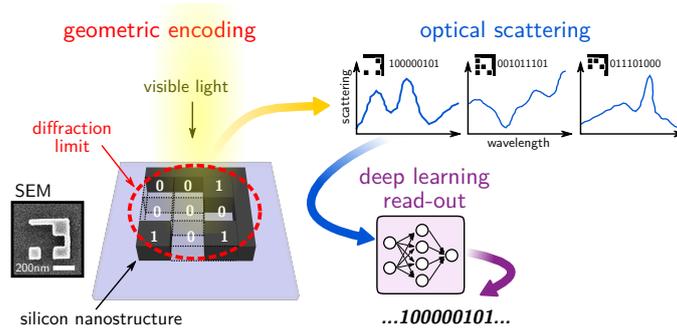


Figure 1.13: By structuring diffraction-limit sized silicon nanostructures in the illustrated bar-code like way, the optical scattering spectra of each bit-sequence encoding geometry can be rendered unique (using illumination of linear polarization). Training an neural network on experimental data for the classification of the spectra according to the respective bit-sequences, noise-robust read-out of ultra-high data densities can be enabled.

even though the individual bits of information are packed on an area smaller than the optical diffraction limit. However, in a realistic implementation, measurements are noisy (see illustration on the right in figure 1.13). Moreover, the nanostructures cannot be fabricated with infinite precision and hence will possess a certain amount of defects. Therefore the scattering spectra will be deteriorated and especially in the case of high information densities, error-tolerant data retrieval will be very challenging. We have demonstrated recently, that an artificial neural network can be trained on the recognition of noisy scattering spectra and is capable of retrieving almost error-free at least 9 bits of information encoded per diffraction limited area – it is hence able to distinguish 512 possible geometries from noisy experimental spectra. By training the network on a set of actual measurements, the ANN implicitly learns to deal with the experimental noise and with spectral fluctuations due to geometric variations. For more information on the explicit geometry design, the neural network architecture and detailed benchmarks, we refer the reader to our former publication [86].

1.5.2 Speckle reconstruction for real-time hyperspectral imaging

As a second example, we discuss the ultra-fast deconstruction of optical speckle patterns, enabling real-time hyperspectral imaging through a bundle of multimode fibers. The concept is illustrated in figure 1.14a. A light source is spectrally filtered (here by an acousto-optical tunable filter, AOTF), then an intensity-mode spatial light modulator (SLM) is used to project an image (or a video) using the light. The video is projected onto the input of a multicore, multimode fiber bundle (MCMMF) at a small angle (here 4°), and the light propagated from the output of the MCMMF is projected on a CMOS camera. The fiber outputs of the MCMMF form a mosaic of the origi-

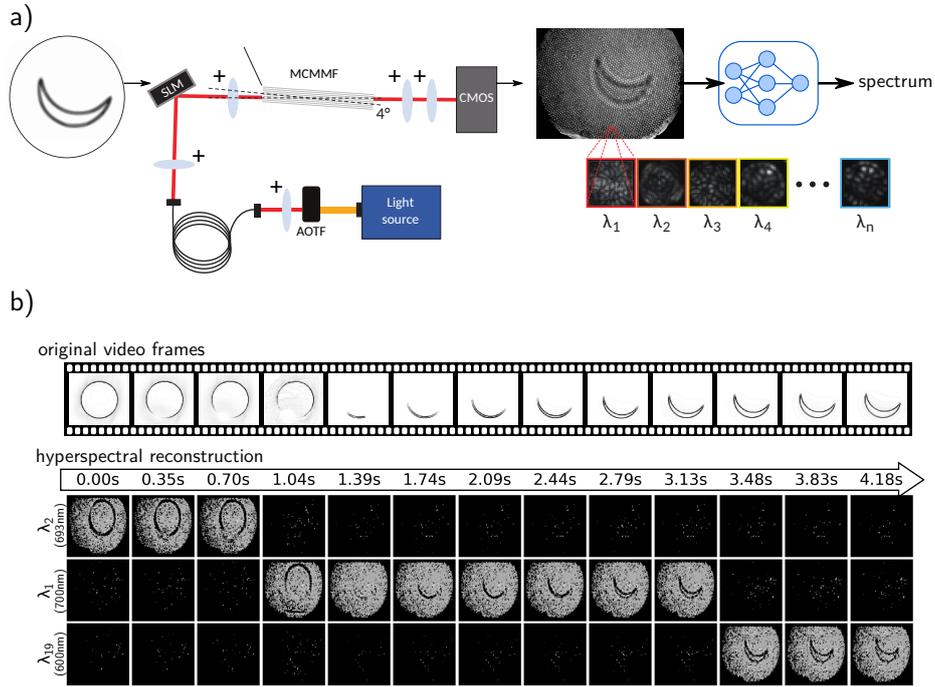


Figure 1.14: a) Scheme of the experimental setup for speckle-based hyperspectral imaging. A broadband supercontinuum laser is spectrally filtered by an acousto-optical tunable filter (AOTF). Subsequently, a spatial light modulator (SLM) is used for image generation. The light is then focused on a slightly tilted multi-core, multimode fiber (MCMMF), which serves as imaging system. Each fiber core of the bundle represents a pixel of the image, and the speckle pattern produced by each fiber encodes the spectral information. The speckles are finally deconstructed by an artificial neural network. b) Demonstration of real-time hyperspectral video reconstruction via DL. A video is projected on the fiber bundle, the filtered wavelength is changed by the AOTF during playback. Top: Original video frames. Bottom: Temporal evolution of the spectral reconstructions of three wavelength channels of the approx. 2700 fiber cores of the MCMMF. a) adapted from [87], copyright (2019) Optical Society of America.

nal image which is overlaid with a characteristic speckle pattern for each fiber. The speckle patterns generated by the fibers are deterministic and wavelength dependent within a characteristic spectral decorrelation width of a few nm, therefore allowing to extract the spectral information separately for each element of the fiber bundle. Different methods for extraction can be applied, ranging from matrix inversion tech-

niques (pseudo-inverse or Tikhonov regularization), to compressive sensing and DL. By training an artificial neural network on a experimental set of speckle patterns, the ANN can indeed learn to decode the spectrum which is encoded in the speckle pattern [87]. The DL-based reconstruction is far more noise-tolerant than transmission matrix approaches [88], and hugely faster in an undersampling configuration in comparison with compressive sensing techniques [89]. As shown in figure 1.14b, the concept allows reconstruction rates of several frames per second. It maintains a high reconstruction fidelity in undersampling conditions and it can be taught robustness against noise by adding according disturbed data to the training set [87].

1.6 Conclusion and Outlook

In conclusion, in this chapter we have provided an overview of deep-learning applications in nanophotonics, including ultra-fast physics predictors, inverse design neural networks and data characterization techniques based on deep ANNs. We put a particular emphasis on discussing practical aspects concerning these deep learning applications. For the example of an optical response predictor for individual photonic nanostructures, we explained how to choose configurations such as the network layout, the neuron activations or the loss function as well as which representations of the data are favorable for efficient deep learning. On the single nanostructure, we then demonstrated inverse design of the geometry based on a deep learning surrogate model as well as photonics device inverse design using a tandem network. Illustrated by the inverse design problem of an MMI power splitter, we discussed practical methods to improve ANN inverse design performance. Finally, we gave a brief overview of nanophotonic data processing and analysis using deep learning, which we illustrated by two specific examples, namely deep learning enabled robust readout of nanophotonic information storage and speckle based hyperspectral imaging through multimode fiber-bundles, accelerated by deep learning.

The field of deep learning applications in nanophotonics is still young and many research groups are currently working on the use of machine learning techniques to solve their open challenges. Data-driven methods offer an entirely new perspective on many problems, particularly in situations where analytical data analysis fails. In such cases deep learning can be capable to unveil hidden correlations and physical phenomena, which emerge only in a statistical picture.

Still, the current excitement around ANN methods sometimes tempts us to disregard the difficulties and dangers associated with deep learning. For instance, the fact that large amounts of high-quality data are necessary is actually often a major obstacle. Poor data can render deep learning models useless, or in many situations it is impossible to obtain sufficient amounts of data. In this context, interesting propositions have been made recently. For instance, it might be possible to use knowledge migration techniques to access situations where training data is expensive, e.g. to transfer knowledge which is obtained from relatively low-cost simulations to experimental scenarios where large amounts of data are difficult to collect [90]. Another example of important limitations of deep learning techniques is the requirement of

hard-coded geometric models in ANN based inverse design approaches. This imposes to pre-define the complexity of a configuration prior to data generation and training. If a geometric model turns out to be unable to fulfill specific inverse design targets, the computationally highly expensive data generation and training procedure need to be repeated from scratch. Recent works have started to approach this issue by using deep learning based knowledge discovery to determine the physical limitations of geometric models. In nanophotonics this can be done, for instance, by an analysis of a neural network's learned compressed representation of structural inputs and of its optical responses [84]. By evaluating in this way the physical feasibility of design targets for several geometrical models, an assessment of the required structural complexity in an inverse design problem can be obtained [91]. In conclusion, thanks to the current immense research interest for those techniques, in the near future many more exciting results are to be expected.

Abstract

In the past few years, methods of artificial intelligence and in particular deep learning (DL) have been broadly discussed in the context of nanophotonics and manifold formerly impossible applications, were unexpectedly enabled thanks to DL. [4]

Probably the most extensively discussed application of deep artificial neural networks (ANNs) in nanophotonics is the inverse design of photonic devices and nanostructures [80]. But an abundance of further applications exists, ranging from enhanced and robust data-processing and “knowledge discovery”, [86] over the ultra-rapid surrogate modeling for the acceleration of physics-simulations, [37] to efficiently solving experimental inverse problems like sparse data reconstruction. [87]

In this chapter we will discuss selected pioneering applications which are enabled thanks to DL-based techniques. We provide a critical review of some of the aforementioned novel possibilities, and we discuss in particular possible pitfalls and weaknesses that are inherent to the data-driven nature of numerical methods based on ANNs.

Keywords

Nanophotonics inverse design; deep learning; generalized predictor; tandem network; U-Net; compressive sensing; data storage; hyperspectral imaging

Bibliography

- [1] Davide Piccinotti, Kevin F. MacDonald, Simon Gregory, Ian Youngs, and Nikolay I. Zheludev. Artificial Intelligence for Photonics and Photonic Materials. *Reports on Progress in Physics*, 2020.
- [2] Lujun Huang, Lei Xu, and Andrey E. Miroshnichenko. Deep Learning Enabled Nanophotonics. *Advances in Deep Learning*, July 2020.
- [3] Wei Ma, Zhaocheng Liu, Zhaxylyk A. Kudyshev, Alexandra Boltasseva, Wenshan Cai, and Yongmin Liu. Deep learning for the design of photonic structures. *Nature Photonics*, pages 1–14, October 2020.
- [4] Jiaqi Jiang, Mingkun Chen, and Jonathan A. Fan. Deep neural networks for the evaluation and design of photonic devices. *arXiv:2007.00084 [physics]*, June 2020.
- [5] Ravi S. Hegde. Deep learning: A new tool for photonic nanostructure design. *Nanoscale Advances*, 2(3):1007–1023, March 2020.
- [6] Jie Fang, Anand Swain, Rohit Unni, and Yuebing Zheng. Decoding Optical Data with Machine Learning. *Laser & Photonics Reviews*, n/a(n/a):2000422, 2020.
- [7] Peter R. Wiecha, Arnaud Arbouet, Christian Girard, and Otto L. Muskens. Deep learning in nano-photonics: Inverse design and beyond. *Photonics Research*, 9(3), 2021.
- [8] Richard Baraniuk, David Donoho, and Matan Gavish. The science of deep learning. *Proceedings of the National Academy of Sciences*, 117(48):30029–30032, 2020.
- [9] Terrence J. Sejnowski. The unreasonable effectiveness of deep learning in artificial intelligence. *Proceedings of the National Academy of Sciences*, 117(48):30033–30038, 2020.
- [10] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, USA, 2013. Citeseer.
- [11] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford,

Bibliography

- Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 300, pages 1877–1901, July 2020.
- [12] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015.
- [13] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [14] Dominique Barchiesi, Christian Girard, Olivier J. F. Martin, Daniel Van Labeke, and Daniel Courjon. Computing the optical near-field distributions around complex subwavelength surface structures: A comparative study of different methods. *Physical Review E*, 54(4):4285–4292, October 1996.
- [15] Benjamin Gallinet, Jérémy Butet, and Olivier J. F. Martin. Numerical methods for nanophotonics: Standard problems and future challenges. *Laser & Photonics Reviews*, 9(6):577–603, November 2015.
- [16] André-Pierre Blanchard-Dionne and Olivier J. F. Martin. Teaching optics to a machine learning network. *Optics Letters*, 45(10):2922–2925, May 2020.
- [17] Sensong An, Clayton Fowler, Bowen Zheng, Mikhail Y. Shalaginov, Hong Tang, Hang Li, Li Zhou, Jun Ding, Anuradha Murthy Agarwal, Clara Rivero-Baleine, Kathleen A. Richardson, Tian Gu, Juejun Hu, and Hualiang Zhang. A Deep Learning Approach for Objective-Driven All-Dielectric Metasurface Design. *ACS Photonics*, 6(12):3196–3207, December 2019.
- [18] Sensong An, Bowen Zheng, Mikhail Y. Shalaginov, Hong Tang, Hang Li, Li Zhou, Yunxi Dong, Mohammad Haerinia, Anuradha Murthy Agarwal, Clara Rivero-Baleine, Myungkoo Kang, Kathleen A. Richardson, Tian Gu, Juejun Hu, Clayton Fowler, and Hualiang Zhang. Deep Convolutional Neural Networks to Predict Mutual Coupling Effects in Metasurfaces. *arXiv:2102.01761 [physics]*, February 2021.
- [19] Tian Zhang, Jia Wang, Qi Liu, Jinzan Zhou, Jian Dai, Xu Han, Yue Zhou, and Kun Xu. Efficient spectrum prediction and inverse design for plasmonic waveguide systems based on artificial neural networks. *Photonics Research*, 7(3):368–380, March 2019.
- [20] Mohammad H. Tahersima, Keisuke Kojima, Toshiaki Koike-Akino, Devesh Jha, Bingnan Wang, Chungwei Lin, and Kieran Parsons. Deep Neural Network Inverse Design of Integrated Photonic Power Splitters. *Scientific Reports*, 9(1):1368, February 2019.
- [21] Alec M. Hammond and Ryan M. Camacho. Designing integrated photonic devices using artificial neural networks. *Optics Express*, 27(21):29620–29638, October 2019.

Bibliography

- [22] Yair Rivenson, Zoltán Göröcs, Harun Günaydin, Yibo Zhang, Hongda Wang, and Aydogan Ozcan. Deep learning microscopy. *Optica*, 4(11):1437–1443, November 2017.
- [23] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, February 2019.
- [24] Han Gao, Luning Sun, and Jian-Xun Wang. PhyGeoNet: Physics-Informed Geometry-Adaptive Convolutional Neural Networks for Solving Parameterized Steady-State PDEs on Irregular Domain. *Journal of Computational Physics*, 428:110079, 2020.
- [25] Ben Moseley, Andrew Markham, and Tarje Nissen-Meyer. Solving the wave equation with physics-informed deep learning. *arXiv:2006.11894 [physics]*, June 2020.
- [26] Zhiwei Fang and Justin Zhan. Deep Physical Informed Neural Networks for Meta-material Design. *IEEE Access*, 8:24506–24513, 2020.
- [27] Shehryar Malik, Usman Anwar, Ali Ahmed, and Alireza Aghasi. Learning To Solve Differential Equations Across Initial Conditions. *arXiv:2003.12159 [cs, stat]*, April 2020.
- [28] Itzik Malkiel, Michael Mrejen, Achiya Nagler, Uri Arieli, Lior Wolf, and Haim Suchowski. Plasmonic nanostructure design and characterization via Deep Learning. *Light: Science & Applications*, 7(1):60, September 2018.
- [29] Christian C. Nadell, Bohao Huang, Jordan M. Malof, and Willie J. Padilla. Deep learning for accelerated all-dielectric metasurface design. *Optics Express*, 27(20):27523–27535, September 2019.
- [30] Jonathan Trisno, Hao Wang, Hong Tao Wang, Ray J. H. Ng, Soroosh Daqiqeh Rezaei, and Joel K. W. Yang. Applying Machine Learning to the Optics of Dielectric Nano-Blobs. *Advanced Photonics Research*, page 2000068, 2020.
- [31] Sunae So, Jungho Mun, and Junsuk Rho. Simultaneous Inverse Design of Materials and Structures via Deep Learning: Demonstration of Dipole Resonance Engineering Using Core–Shell Nanoparticles. *ACS Applied Materials & Interfaces*, 11(27):24264–24268, July 2019.
- [32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25*, pages 1097–1105, 2012.
- [33] Peter R. Wiecha. pyGDM—A python toolkit for full-field electro-dynamical simulations and evolutionary optimization of nanostructures. *Computer Physics Communications*, 233:167–192, December 2018.

Bibliography

- [34] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv:1502.03167 [cs]*, February 2015.
- [35] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4278–4284, February 2016.
- [36] Yongzhong Li, Yinpeng Wang, Shutong Qi, Qiang Ren, Lei Kang, Sawyer D. Campbell, Pingjuan L. Werner, and Douglas H. Werner. Predicting Scattering From Complex Nano-Structures via Deep Learning. *IEEE Access*, 8:139983–139993, 2020.
- [37] Peter R. Wiecha and Otto L. Muskens. Deep Learning Meets Nanophotonics: A Generalized Accurate Predictor for Near Fields and Far Fields of Arbitrary 3D Nanostructures. *Nano Letters*, 20(1):329–338, January 2020.
- [38] Bruce T. Draine and Piotr J. Flatau. Discrete-dipole approximation for scattering calculations. *Journal of the Optical Society of America A*, 11(4):1491, April 1994.
- [39] Christian Girard. Near fields in nanostructures. *Reports on Progress in Physics*, 68(8):1883–1933, August 2005.
- [40] C. Girard, E. Dujardin, G. Baffou, and R. Quidant. Shaping and manipulation of light fields with bottom-up plasmonic structures. *New Journal of Physics*, 10(10):105016, October 2008.
- [41] Christian Girard, Jean-Claude Weeber, Alain Dereux, Olivier J. F. Martin, and Jean-Pierre Goudonnet. Optical magnetic near-field intensities around nanometer-scale surface structures. *Physical Review B*, 55(24):16487–16497, June 1997.
- [42] Peter R. Wiecha, Peter R. Wiecha, Clément Majorel, Christian Girard, Aurélien Cuche, Vincent Paillard, Otto L. Muskens, and Arnaud Arbouet. Design of plasmonic directional antennas via evolutionary optimization. *Optics Express*, 27(20):29069–29081, September 2019.
- [43] Andrey B. Evlyukhin, Carsten Reinhardt, and Boris N. Chichkov. Multipole light scattering by nonspherical nanoparticles in the discrete dipole approximation. *Physical Review B*, 84(23):235429, December 2011.
- [44] P. C. Chaumet and M. Nieto-Vesperinas. Coupled dipole method determination of the electromagnetic force on a particle over a flat dielectric substrate. *Physical Review B*, 61(20):14119–14127, May 2000.
- [45] Guillaume Baffou, Romain Quidant, and Christian Girard. Thermoplasmonics modeling: A Green’s function approach. *Physical Review B*, 82(16):165424, October 2010.

Bibliography

- [46] R. Marty, G. Baffou, A. Arbouet, C. Girard, and R. Quidant. Charge distribution induced inside complex plasmonic nanoparticles. *Optics Express*, 18(3):3035–3044, February 2010.
- [47] Patrick C. Chaumet, Adel Rahmani, and Garnett W. Bryant. Generalization of the coupled dipole method to periodic structures. *Physical Review B*, 67(16):165404, April 2003.
- [48] Bruce T. Draine and Piotr J. Flatau. Discrete-dipole approximation for periodic targets: Theory and tests. *JOSA A*, 25(11):2693–2703, November 2008.
- [49] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597 [cs]*, May 2015.
- [50] Li Gao, Xiaozhong Li, Dianjing Liu, Lianhui Wang, and Zongfu Yu. A Bidirectional Deep Neural Network for Accurate Silicon Color Design. *Advanced Materials*, 31(51):1905467, 2019.
- [51] Sunae So, Trevon Badloe, Jaebum Noh, Jorge Bravo-Abad, and Junsuk Rho. Deep learning enabled inverse design in nanophotonics. *Nanophotonics*, 0(0), 2020.
- [52] Zhaocheng Liu, Dayu Zhu, Kyu-Tae Lee, Andrew S. Kim, Lakshmi Raju, and Wenshan Cai. Compounding Meta-Atoms into Metamolecules with Hybrid Artificial Intelligence Techniques. *Advanced Materials*, 32(6):1904790, 2020.
- [53] Maksym V. Zhelyeznyakov, Steven L. Brunton, and Arka Majumdar. Deep learning to accelerate Maxwell’s equations for inverse design of dielectric metasurfaces. *arXiv:2008.10632 [physics]*, August 2020.
- [54] Raphaël Pestourie, Youssef Mroueh, Thanh V. Nguyen, Payel Das, and Steven G. Johnson. Active learning of deep surrogates for PDEs: Application to metasurface design. *arXiv:2008.12649 [physics]*, August 2020.
- [55] Wei Ma, Feng Cheng, Yihao Xu, Qinlong Wen, and Yongmin Liu. Probabilistic Representation and Inverse Design of Metamaterials Based on a Deep Generative Model with Semi-Supervised Learning Strategy. *Advanced Materials*, 31(35):1901111, 2019.
- [56] Victor Kalt, Alma K. González-Alcalde, Soukaina Es-Saidi, Rafael Salas-Montiel, Sylvain Blaize, and Demetrio Macías. Metamodeling of high-contrast-index gratings for color reproduction. *JOSA A*, 36(1):79–88, January 2019.
- [57] Jiaqi Jiang and Jonathan A. Fan. Global Optimization of Dielectric Metasurfaces Using a Physics-Driven Neural Network. *Nano Letters*, 19(8):5366–5372, August 2019.
- [58] Alma K. González-Alcalde, Rafael Salas-Montiel, Victor Kalt, Sylvain Blaize, and Demetrio Macías. Engineering colors in all-dielectric metasurfaces: Metamodeling approach. *Optics Letters*, 45(1):89–92, January 2020.

Bibliography

- [59] J. R. Thompson, J. R. Thompson, J. A. Burrow, P. J. Shah, P. J. Shah, J. Slagle, E. S. Harper, A. Van Rynbach, I. Agha, and M. S. Mills. Artificial neural network discovery of a switchable metasurface reflector. *Optics Express*, 28(17):24629–24656, August 2020.
- [60] Mahmoud M. R. Elsawy, Stéphane Lanteri, Régis Duvigneau, Jonathan A. Fan, and Patrice Genevet. Numerical Optimization Methods for Metasurfaces. *Laser & Photonics Reviews*, 14(10):1900445, 2020.
- [61] R. S. Hegde. Photonics Inverse Design: Pairing Deep Neural Networks With Evolutionary Algorithms. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(1):1–8, January 2020.
- [62] Jiaqi Jiang and Jonathan A. Fan. Multiobjective and categorical global optimization of photonic structures based on ResNet generative neural networks. *Nanophotonics*, 10(1):361–369, September 2020.
- [63] Mikhail A. Kats, Patrice Genevet, Guillaume Aoust, Nanfang Yu, Romain Blanchard, Francesco Aieta, Zeno Gaburro, and Federico Capasso. Giant birefringence in optical antenna arrays with widely tailorable optical anisotropy. *Proceedings of the National Academy of Sciences*, 109(31):12364–12368, July 2012.
- [64] Leo-Jay Black, Yudong Wang, C. H. de Groot, Arnaud Arbouet, and Otto L. Muskens. Optimal Polarization Conversion in Coupled Dimer Plasmonic Nanoantennas for Metasurfaces. *ACS Nano*, 8(6):6390–6399, June 2014.
- [65] Peter R. Wiecha, Leo-Jay Black, Yudong Wang, Vincent Paillard, Christian Girard, Otto L. Muskens, and Arnaud Arbouet. Polarization conversion in plasmonic nanoantennas for metasurfaces using structural asymmetry and mode hybridization. *Scientific Reports*, 7:40906, January 2017.
- [66] Peter R. Wiecha, Arnaud Arbouet, Aurélien Cuche, Vincent Paillard, and Christian Girard. Decay rate of magnetic dipoles near nonmagnetic nanostructures. *Physical Review B*, 97(8):085411, February 2018.
- [67] Nicolas Bonod, Sébastien Bidault, Geoffrey W. Burr, and Mathieu Mivelle. Evolutionary Optimization of All-Dielectric Magnetic Nanoantennas. *Advanced Optical Materials*, 7(10):1900121, 2019.
- [68] Peter R. Wiecha, Arnaud Arbouet, Christian Girard, Aurélie Lecestre, Guilhem Larrieu, and Vincent Paillard. Evolutionary multi-objective optimization of colour pixels based on dielectric nanoantennas. *Nature Nanotechnology*, 12(2):163–169, February 2017.
- [69] D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, April 1997.

Bibliography

- [70] Li Jiang, Li Jiang, Xiaozhong Li, Xiaozhong Li, Qingxin Wu, Lianhui Wang, Li Gao, and Li Gao. Neural network enabled metasurface design for phase manipulation. *Optics Express*, 29(2):2521–2528, January 2021.
- [71] Yashar Kiarashinejad, Sajjad Abdollahramezani, and Ali Adibi. Deep learning approach based on dimensionality reduction for designing electromagnetic nanostructures. *arXiv:1902.03865 [physics, stat]*, April 2019.
- [72] Dianjing Liu, Yixuan Tan, Erfan Khoram, and Zongfu Yu. Training Deep Neural Networks for the Inverse Design of Nanophotonic Structures. *ACS Photonics*, 5(4):1365–1369, April 2018.
- [73] Abhishek Mall, Abhijeet Patil, Dipesh Tamboli, Amit Sethi, and Anshuman Kumar. Fast design of plasmonic metasurfaces enabled by deep learning. *Journal of Physics D: Applied Physics*, 53(49):49LT01, October 2020.
- [74] Rohit Unni, Kan Yao, and Yuebing Zheng. Deep Convolutional Mixture Density Network for Inverse Design of Layered Photonic Structures. *ACS Photonics*, 7(10), September 2020.
- [75] Ying-Tao Luo, Peng-Qi Li, Dong-Ting Li, Yu-Gui Peng, Zhi-Guo Geng, Shu-Huan Xie, Yong Li, Andrea Alù, Jie Zhu, and Xue-Feng Zhu. Probability-Density-Based Deep Learning Paradigm for the Fuzzy Design of Functional Metastructures. *Research*, 2020:8757403, September 2020.
- [76] Zhaocheng Liu, Dayu Zhu, Sean P. Rodrigues, Kyu-Tae Lee, and Wenshan Cai. Generative Model for the Inverse Design of Metasurfaces. *Nano Letters*, 18(10):6570–6576, September 2018.
- [77] Sensong An, Bowen Zheng, Hong Tang, Mikhail Y. Shalaginov, Li Zhou, Hang Li, Myungkoo Kang, Kathleen A. Richardson, Tian Gu, Juejun Hu, Clayton Fowler, and Hualiang Zhang. Multifunctional Metasurface Design with a Generative Adversarial Network. *Advanced Optical Materials*, page 2001433, 2020.
- [78] Jiaqi Jiang, David Sell, Stephan Hoyer, Jason Hickey, Jianji Yang, and Jonathan A. Fan. Free-Form Diffractive Metagrating Design Based on Generative Adversarial Networks. *ACS Nano*, 13(8):8872–8878, August 2019.
- [79] Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W. Pellegrini, Ralf S. Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing Inverse Problems with Invertible Neural Networks. *arXiv:1808.04730 [cs, stat]*, February 2019.
- [80] Nicholas J. Dinsdale, Peter R. Wiecha, Matthew Delaney, Jamie Reynolds, Martin Ebert, Ioannis Zeimpekis, David J. Thomson, Graham T. Reed, Philippe Lalanne, Kevin Vynck, and Otto L. Muskens. Deep learning enabled design of complex transmission matrices for universal optical components. *ACS Photonics*, 8(1):283–295, 2021.

Bibliography

- [81] Andre-Pierre Blanchard-Dionne and Olivier J. F. Martin. Successive training of a generative adversarial network for the design of an optical cloak. *OSA Continuum*, 4(1):87–95, January 2021.
- [82] Fufang Wen, Jiaqi Jiang, and Jonathan A. Fan. Robust Freeform Metasurface Design Based on Progressively Growing Generative Networks. *ACS Photonics*, 7(8):2098–2104, August 2020.
- [83] Yashar Kiarashinejad, Sajjad Abdollahramezani, Mohammadreza Zandehshahvar, Omid Hemmatyar, and Ali Adibi. Deep Learning Reveals Underlying Physics of Light–Matter Interactions in Nanophotonic Devices. *Advanced Theory and Simulations*, 2(9):1900088, 2019.
- [84] Yashar Kiarashinejad, Mohammadreza Zandehshahvar, Sajjad Abdollahramezani, Omid Hemmatyar, Reza Pourabolghasem, and Ali Adibi. Knowledge Discovery in Nanophotonics Using Geometric Deep Learning. *Advanced Intelligent Systems*, 2(2):1900132, 2020.
- [85] Raban Iten, Tony Metger, Henrik Wilming, Lídia del Rio, and Renato Renner. Discovering Physical Concepts with Neural Networks. *Physical Review Letters*, 124(1):010508, January 2020.
- [86] Peter R. Wiecha, Aurélie Lecestre, Nicolas Mallet, and Guilhem Larrieu. Pushing the limits of optical information storage using deep learning. *Nature Nanotechnology*, 14:237–244, January 2019.
- [87] Ulas Kürüm, Peter R. Wiecha, Rebecca French, and Otto L. Muskens. Deep learning enabled real time speckle recognition and hyperspectral imaging using a multimode fiber array. *Optics Express*, 27(15):20965–20979, July 2019.
- [88] Sébastien Popoff, Geoffroy Lerosey, Mathias Fink, Albert Claude Boccara, and Sylvain Gigan. Image transmission through an opaque material. *Nature Communications*, 1:81, September 2010.
- [89] Rebecca French, Sylvain Gigan, and Otto L. Muskens. Snapshot fiber spectral imaging using speckle correlations and compressive sensing. *Opt. Express*, 26:32302–32316, 2018.
- [90] Yurui Qu, Li Jing, Yichen Shen, Min Qiu, and Marin Soljačić. Migrating Knowledge between Physical Scenarios Based on Artificial Neural Networks. *ACS Photonics*, 6(5):1168–1174, May 2019.
- [91] Mohammadreza Zandehshahvar, Yashar Kiarashi, Muliang Zhu, Hossein Maleki, Tyler Brown, and Ali Adibi. Manifold Learning for Knowledge Discovery and Intelligent Inverse Design of Photonic Nanostructures: Breaking the Geometric Complexity. *arXiv:2102.04454 [physics]*, February 2021.