PHOTONICS Research

Bidirectional cascaded deep neural networks with a pretrained autoencoder for dielectric metasurfaces

WEICHAO KONG,^{1,†} JUN CHEN,^{2,†} ZENGXIN HUANG,¹ AND DENGFENG KUANG^{1,*}

¹Tianjin Key Laboratory of Micro-scale Optical Information Science and Technology, and Institute of Modern Optics, Nankai University, Tianjin 300350, China

²College of Physics and Electronic Engineering, Taishan University, Taian 271000, China

*Corresponding author: dfkuang@nankai.edu.cn

Received 21 April 2021; revised 17 June 2021; accepted 17 June 2021; posted 22 June 2021 (Doc. ID 428425); published 30 July 2021

Metasurfaces composed of meta-atoms provide promising platforms for manipulating amplitude, phase, and polarization of light. However, the traditional design methods of metasurfaces are time consuming and laborious. Here, we propose a bidirectional cascaded deep neural network with a pretrained autoencoder for rapid design of dielectric metasurfaces in the range of 450 nm to 850 nm. The forward model realizes a prediction of amplitude and phase responses with a mean absolute error of 0.03. Meanwhile, the backward model can retrieve patterns of meta-atoms in an inverse-design manner. The availability of this model is demonstrated by database establishment, model evaluation, and generalization testing. Furthermore, we try to reveal the mechanism behind the model in a visualization way. The proposed approach is beneficial to reduce the cost of computation burden and improve nanophotonic design efficiency for solving electromagnetic on-demand design issues automatically. © 2021 Chinese Laser Press

https://doi.org/10.1364/PRJ.428425

1. INTRODUCTION

In recent years, metasurfaces-planar optical elements composed of artificially fabricated photonic atoms-have emerged as a versatile platform for wavefront shaping. Compared with bulky and costly refractive optical components [1], metasurfaces have the advantages of being light weight and often low cost. A number of compact devices based on metasurfaces, such as flat lenses [2–4], holograms [5,6], and axicons [7], have been demonstrated due to their excellent ability in controlling the amplitude, phase, and polarization of incident light. As these nanostructured materials require labor intensive fabrication, an accurate prediction of the optical spectrum and structure of the envisioned metasurfaces must be preemptively articulated [8]. Therefore, these researches largely rely on iterative numerical full-wave simulations such as the nanofinite-element method (FEM), finite-difference time-domain (FDTD) method, and finite integration technique (FIT). Even though some commercial softwares [9-11] can make accurate predictions for electromagnetic response, it is time and computation consuming.

Deep learning is a relatively new machine learning subfield, and deep neural networks are widely used in image classification [12,13], semantic segmentation [14,15], and object detection [16]. It provides a promising solution to reduce time-consuming calculations and to produce results with limited computational resources [17]. A deep neural network aimed at optimization of dielectric metasurfaces was designed in 2019 [18]. A deep-learning-based model, comprising two bidirectional neural networks assembled by a partial stacking strategy, was reported to automatically design and optimize three-dimensional chiral metamaterials with a strong chiroptical response at predesignated wavelengths [19]. In these studies, bilinear tensor layers are employed to expand the size of the input tensor, transforming upsampling networks to downsampling networks. The laws implied by data are difficult to be explained with visualization methods. Other approaches on metasurface on-demand design were proposed and put in the limelight in the last few years [20–25]. However, there is still no interpretation of the physical mechanism behind the model.

An autoencoder is a deep neural network that tries to restore input through automatically acquiring meaningful features. As an unsupervised learning method, it is frequently used in data reduction and image denoising [26,27]. Here, we report a metasurface design method based on a bidirectional cascaded deep neural network (CDNN), which consists of a simulator, an autoencoder, and a translator. The simulator holds on a task of forward predicting amplitude and phase responses for metaatoms. On the other hand, the backward networks predict the shape of meta-atoms with on-demand amplitude and phase responses. To avoid an upsampling network in the backward path, the autoencoder takes the role of data reduction by representing each meta-atom with a low-dimensional eigenvector. The on-demand amplitude and phase responses are first mapped to the eigenvector by the translator network, and then decoded to patterns of meta-atoms. A database of 20,000 randomly shaped meta-atoms is generated and used to train and test our network. In addition, feature maps of each convolutional layer and activation mapping are visualized to interpret the physical mechanism behind the model. Our model will be applied as a very powerful tool in accelerating the on-demand design of nanophotonic devices, systems, and architectures for real world applications.

2. METHODS

A. Establishment of Database

To design a more generalizable network, our database is made up of randomly generated meta-atoms. Figure 1(a) exhibits a combination of 100 randomly generated meta-atoms. One hundred gray images of 40×40 pixels are spliced into one picture in the form of 10 rows and 10 columns. In the figure, the black part is the substrate, and the rest is high-index dielectric nanofins; the pixel gray value is proportional to the height (255 represents 700 nm). A meta-atom model selected from Fig. 1(a) (dotted line) is shown in Figs. 1(b) and 1(c). As shown in Fig. 1(b), the all-dielectric meta-atom is generated randomly with a fixed unit size of 400 nm \times 400 nm. Four rectangular nanofins with different shapes and heights (500 nm, 600 nm, or 700 nm) are generated randomly on the left side of the unit. Then, four nanofins are placed on another side of the unit in a mirror-symmetry way to form a meta-atom. A blue margin of 20 nm is set to minimize mutual couplings. In the process, the minimum generative resolution is 10 nm, as shown on the black grid of the nanofins in Fig. 1(c), and a taller nanofin covers a shorter one. Each geometric parameter is saved to make a two-dimensional gray image, using the pixel gray value representing the height.

The FDTD method is employed to calculate phase and amplitude responses. Phase response is calculated by a meta-atom with no nanofin as a reference. For each meta-atom, periodic boundary conditions are applied in x and y directions, and perfectly matched layers are applied in z direction. The incident light is set as linearly polarized light along x direction in the range of 450–850 nm and discretized into 41 data points.

In addition, we analyze the phase and amplitude responses based on the meta-atom in Fig. 1(c). In Fig. 1(d), the red curve represents phase response and blue represents amplitude. Here, phase shifts are realized by a couple of resonant modes. The electric field distributions at the wavelengths labeled in



Fig. 1. Generation process of database. (a) Association of 100 meta-atoms. (b) Sample and its generation method. (c) Structure of the selected random meta-atom. (d) Transmission spectrum (phase and amplitude) of the meta-atom. (e) $|E_x|^2$ at several wavelengths in (d).

Fig. 1(d) are respectively plotted in Fig. 1(e). Optical coupling among dielectric nanofins is weak because of their higher refractive indices compared with the surrounding environment [28]. The resonances in nanofins show that the induced optical fields are highly concentrated inside the dielectric structures, causing negligible interaction with neighbors [1]. Although the amplitude and phase responses are simulated on a periodic grid of identical meta-atoms, it can be devoted to a given wavefront shaping.

B. Material

Material dispersion plays a paramount role in the establishment of the database and the performance of metasurfaces. Dispersive material in visible light should meet two principles: (i) the refractive index is high and scattering loss is negligible at visible wavelengths as far as possible; (ii) the coupling effects between neighboring meta-atoms are weak. Figure 2 compares four different combination materials (TiO₂ [29], GaN [30,31], a-Si [32], Si₃N₄ [33]) with the same shaped meta-atom. The refractive index is represented as n and the extinction coefficient as k. In the FDTD method, the calculation time for metaatoms largely depends on the grid step size. The more complicated propagation modes and coupling effects in nanofins require smaller step sizes and longer calculation times [34]. It can be seen that the advantages of TiO₂ are negligible absorption at visible wavelengths and a sufficiently high refractive index. However, it takes about 30 min for each TiO₂ meta-atom. Furthermore, TiO₂ is more prone to generate coupling effects, which leads to unpredictable phase shifts. Nonnegligible absorption is the main shortcoming for a-Si meta-atoms. Meta-atoms of Si₃N₄ and GaN meet the requirements of convergence speed and transmissivity. As a result, GaN nanofin on Al₂O₃ substrate is chosen because the combination takes less time in calculation, which can speed up the data collection process. From the numerical simulations, we have collected a database of 20,000 samples, and there are no data augmentation methods such as image rotation or cutting. Amplitude and phase responses are saved as vectors with the dimension of 82×1 , and the structure parameters of meta-atoms are saved as images with the dimension of 40×40 . The database is randomly divided into 18,000 (train sets) and 2000 (test sets) for training and testing [35].

C. Network Architecture

The structure of our CDNN is shown in Fig. 3(a). We divide the deep neural networks into four parts: encoder, decoder, simulator, and translator. T is phase and amplitude response simulated by the FDTD method, and T' is prediction from the simulator. Once the model is trained, the simulator can predict T' as input images with meta-atom structures more quickly than its numerical counterpart. As for backward prediction, T with the dimension of 82×1 is transformed to images with the dimension of 40 × 40, indicating a very low input dimension compared with the output dimension for a regression task. This huge mismatch makes it difficult for a network to converge and generalize well, especially when the input spectra have strong variations around resonant frequencies. Previous research has tried to avoid this problem by attaching a bilinear tensor layer [18] or generative adversarial networks [8]. Here, we first represent each meta-atom with a low-dimensional eigenvector with the dimension of 82 × 1 by a pretrained autoencoder. Then T is mapped to the low-dimensional eigenvector by the translator and decoded to the image by the decoder.

In the forward path, the simulator structure is shown in Fig. 3(b). The size of all tensors throughout the network is marked at the bottom of each block. Different layers of the convolutional neural network (CNN) are connected by convolutional operations. Kernels are represented as darker green, and the convolutional operations are illustrated as blue lines: the kernel multiplies the values of the tensor in the region of that kernel and then sums them together as a new value in tensors in the next layer. At the end of the CNN, we attach two fully connected layers (dimensions are shown underneath) to approximate a spectral tensor. In the model, a leaky ReLU of $\alpha = 0.2$ is used for each convolutional layer and tanh for each fully connected layer. The convolutional layer maps the input tensor x_k to the output tensor x_{k+1} :



Fig. 2. Comparison of four kinds of frequently used material combinations in visible light.



Fig. 3. Structure of CDNN for metasurfaces. (a) Forward and backward networks for prediction of transmission spectrum and structure of metaatoms. (b)–(d) Structures of the simulator, autoencoder, and translator, respectively.

$$x_{k+1} = \text{leaky ReLU[CONV}_{k_1}(x_k)], \quad (1)$$

where leaky ReLU(·) stands for the rectified linear unit oper-
ation, and CONV(·) stands for the convolution operator (in-
cluding the bias terms). Subscript
$$k_1$$
 denotes the number of
channels. In the simulator, $k_1 = 32$, 32, 64, 64, 128, 128.
Strides of two are used in two, four, and six convolutional layers
to replace max-pooling layers [36]. A dropout layer with 0.1
drop rate behind each fully connected layer except for the out-
put layer is employed to avoid overfitting networks [37], as
shown by blue crosses in Fig. 3(b). Mean absolute error
(MAE) is adopted to calculate the gradient and weight.
MAE is defined as

$$MAE = \frac{\sum_{i} |T_{\text{predicted}} - T_{\text{simulated}}|}{N},$$
 (2)

where N is the number of the entries of $T_{\text{predicted}}$. As a cost function, MAE is not sensitive to outliers, but it is not conducive to the convergence of the model [38]. To ensure the stability of the model, the learning rate will decline with the number of iterations.

As for the backward path, we pretrain the autoencoder to represent each meta-atom with a low-dimensional eigenvector first. The structure of the autoencoder is shown in Fig. 3(c). The shapes of all tensors throughout the network are labeled underneath. The activation function of leaky ReLU is used for each layer except for output layers of the encoder and

decoder. Similarly, a dropout layer with 0.1 drop rate is attached behind each fully connected layer except for the output layer. The image with structure information is flattened into a tensor with a dimension of 1600×1 . Then the tensor is encoded to an eigenvector with the dimension of 82×1 by the encoder and is decoded to a tensor with the dimension of 1600×1 . Finally, we get a restored image by reshaping. Based on the traditional design of an autoencoder [26], two points make the encoded code have features of amplitude and phase responses instead of advanced structure features. One is that the network adopts fully connected layers instead of convolutional layers. The other is a choreographed cost function. The cost function is defined by

$$Loss_{a} = -\frac{1}{N} \sum_{i} [y_{i} \ln(p_{i}) + (1 - y_{i}) \ln(1 - p_{i})] + \frac{\mu \sum_{i} |\text{code} - T|}{N},$$
(3)

where y_i represents the label of the sample *i*. (Positive samples are one and negative samples are zero. Positive samples mean the predicted value is the same as input, and negative samples mean the predicted value is different from input.) p_i represents the probability that sample *i* is predicted to be positive. μ is the parameter to balance losses of two parts, and we choose 0.01 here.

Once the autoencoder is pretrained, the encoder is not involved in the backward path. The backward path is cascaded networks of the translator and decoder. The translator is a deep neural network consisting of fully connected layers, and its structure is shown in Fig. 3(d). It is a classical neural network, and the output tensor shape is labeled underneath. Leaky ReLU is used for each layer except for output layers. Similarly, A dropout layer with 0.1 drop rate is attached behind each fully connected layer except for the output layer. The cost function is MAE due to the regression task. The input T with the dimension of 82×1 is translated to code with the dimension of 82×1 by the translator.

3. RESULTS AND DISCUSSION

A. Model Evaluation of Forward Networks

We build our networks using the TensorFlow v2.0 (Google Inc.) framework. The Adam optimizer is adopted with a learning rate of 1×10^{-5} and decay rate of 1×10^{-4} . The cost functions of training and test sets are sampled once every 20 iterations, as shown in Fig. 4(a). The red curve represents the cost function of test sets and blue of training sets.

The cost function decreases rapidly at the beginning of network training because of a larger gradient. As the gradient decreases, the descending rate of the cost function is slower and the curve tends to be stable. After about 1000 iterations, the descent direction is guided by gradient loss, and an MAE of 0.03 is obtained. The smooth downward trend and the consistency between test and training sets show that our simulator has great generalization performance.

To investigate the effect of convolutional layer depth on prediction performance, the process is implemented by simply changing the simulators' depth as five, eight, and eleven; removing the first, third, and fifth layers to get a five-layer network;



Fig. 4. Evaluation of the simulator. (a) Training and test loss functions along with epochs. (b) Loss functions of different depth. (c) Counts of MAE for whole test sets.

and adding a convolutional layer after the first, third, and fifth layers to get an eleven-layer network. These models are iterated 1000 times under the same conditions, and loss functions of the test set are sampled once every 20 iterations, as shown in Fig. 4(b). The MAE of the simulator with a depth of five layers finally stops near 0.06 and 0.03 for the simulator with depths of eight and eleven, respectively. Increasing the network depth will improve the performance, but too many layers will lead to network degradation. As the depth of the network increases, the singular values of the product matrix become more and more concentrated, while the small number of singular values with very low frequency become arbitrarily large [39]. Therefore, the eight-layer simulator saves computing resources while ensuring prediction accuracy. After completing the training process, all loss values between prediction and simulation in the test sets are shown in Fig. 4(c). The histogram is produced by counting the loss values, which are divided into 15 parts. Among all the test sets, there are 16 samples whose errors fall into the range of 0-0.001, indicating these samples have distinguishing features in the processing of forward networks. Most samples are in the range of 0.002-0.003, which is consistent with the average error of the network. The maximum error occurs in the range of 0.010-0.011, with a total of five samples. This is because some unique features of these samples are lost in the extraction process of convolutional layers, which is an unavoidable phenomenon in a regression task. The simulator is able to predict phase and amplitude responses with an MAE of 0.03, comparable to numerical softwares. In general, the network has a good prediction ability for the sample of the test set.

We select four samples with typical losses in addition to the database and exhibit them in Figs. 5(a)-5(d). The red dots represent the phase response predicted by the network, and the red curve refers to the phase response calculated with the



Fig. 5. Four samples for the simulator. Inset is corresponding meta-atom. (a) MAE of 0.0104. (b) MAE of 0.0327. (c) MAE of 0.0356. (d) MAE of 0.0550.

FDTD method; the blue cubes represent the amplitude predicted by the network, and the blue curve refers to the amplitude calculated with the FDTD method. For the sample in Fig. 5(a), the error between the network prediction and ground truth is 0.0104. Most of the points of phase and amplitude fall on the curve, and even the 2π phase shifts at 600 nm are accurately predicted. The accurate predictions also can be seen in Figs. 5(b) and 5(c). The maximum error is the sample in Fig. 5(d), which is 0.0550. The slightly larger error is caused by the large difference between the predicted phase and the ground truth at 630 nm. It is not a phase shift because the predicted phase point is a remainder of 2π . Therefore, it can be concluded that the network can accurately predict phase and amplitude responses, whether as a continuation or shift. The great performance demonstrates that the simulator has learned FDTD calculated methods accurately. More importantly, it takes only less than 0.01 s for network prediction; in contrast, it takes more than 3 min for numerical simulation. In addition, compared with the input vector composed of structural data, the input images have a better performance in robustness and intuitive understanding. Therefore, it can be used to intuitively study the mechanism of electromagnetic response due to meta-atoms structure.

B. Visualization for Simulator

To study the process of network data fitting, the underlying mechanism is understood by visualizing the feature map and gradient-weighted class activation mapping (Grad-CAM) extracted from each convolutional layer [40]. We select a meta-atom, as shown in Fig. 6(a), and make it into gray image.



Fig. 6. Visualization of forward network. (a) Meta-atom and its structure parameters. (b) Feature maps extracted from the first and second convolutional layers. (c) Feature maps extracted from the third and fourth convolutional layers. (d) Activation thermal maps.

Then the image is fed to the simulator with intermediate convolutional layers as output. The feature maps of the first and second convolutional layers are shown in Fig. 6(b). The number of feature maps of the first and second convolutional layers is 32 because of 32 convolutional kernels. The lower convolutional layers extract basic morphological information, and different convolutional kernels have different responses and weights for meta-atoms. Some focus on the nanopillar itself or part of it, while others focus on parts other than nanopillars. The feature maps of the third and fourth convolutional layers are shown in Fig. 6(c). The middle convolutional layers extract more abstract features. A pixel often represents multiple features, which are fusion features of nanopillars and substrates. The highest layer would put attention on global features instead of local features.

Grad-CAM uses gradient information flowing into the last convolutional layer of the simulator to assign weights to each neuron to make specific decisions of interest. In the method, the predicted electromagnetic response is backpropagated to the final convolutional feature map, which is used to calculate the rough gradient positioning (blue thermal map). It indicates that the model must make a specific decision according to the location. Finally, the thermal map is multiplied by the guided backpropagation point by point to obtain high-resolution and concept specific visualization images. We input 24 samples to make the activation mapping, as shown in Fig. 6(d). The highlights in the figure indicate the important regions in the final decision-making process. It can be seen that the highlighted regions are always not in the center of the nanopillars, but on the edge or between the nanopillars. It can be considered that the network pays more attention to the interaction between the nanopillars than the nanopillars themselves in the training process, which is similar to the coupling effect of FDTD calculation. These are hidden features behind data learned by convolutional neural networks under electromagnetic response monitoring.

C. Model Evaluation of Backward Networks

In the same way, we build the autoencoder using the TensorFlow v2.0 framework. The autoencoder is trained using a batch size of 64 and an Adam optimizer. After more than 10,000 iterations, the cost function stabilizes. Sixty-four samples in test sets are fed into the autoencoder, and restored images are shown in Fig. 7(a). The corresponding original images are displayed in Fig. 7(b). The restored images basically restore the global geometric features of meta-atoms, except for some details after comparison. From the visualization analyses, global features play a major role in the final decision for the transmission spectrum. More importantly, compared with the image of (40,40,1), the feature vector of (82,1) realizes about 80 times feature compression. Therefore, we avoid the problem of network convergence and generalization, which are caused by a severe mismatch, indicating a very low input dimension compared with the output dimension for a regression task.

There is a normal design in the translator training after the pretrained autoencoder. After the training of the backward path is finished, we make a model evaluation for the backward model. To verify the gap between an on-demand transmission spectrum and a network prediction spectrum, the images



Fig. 7. Evaluation of backward networks. (a) Restored images and (b) original images. (c) Counts of MAE for whole test sets.

obtained in backward networks are fed into the simulator. On-demand amplitude and phase responses are sent into the translator to acquire advanced features of meta-atom structure. Then the advanced features are restored to images by the pretrained decoder. Last, the restored images are fed in the simulator to predict transmissions and phase shifts. Figure 7(c) shows the MAE between on-demand and predicted transmission spectra in test sets. There are a total of 1128 samples whose MAEs are less than 0.1. The larger error is the inevitable result of accumulation losses of CDNN, and the details are lost in data reduction. The decline of predicted accuracy is an obligatory phenomenon owing to the feasibility of dimensionality reduction.

We input four samples in addition to the database and compare predicted results with targets, as shown in Fig. 8. The red dots represent the phase response designed by the network, and the red curve refers to the on-demand phase response; the blue cubes represent the amplitude response designed by the network, and the blue curve refers to on-demand amplitude. The minimum error is 0.0317, and the predicted point has a larger error only at a point with 2π phase shifts, as shown in Fig. 8(a). Similar miss points appear in samples in Figs. 8(b) and 8(c). The details of determining 2π phase shifts are lost in the cascade process of the network. The maximum error is the sample in Fig. 8(d). Most of the predicted points of the network are not on the curve; only the trend of the electromagnetic response is predicted. Compared with forward networks, this is due to the accumulated error of the network cascade, and the dimension compression process leads to the loss of some details. However, it is also a successful model for transforming upsampling to a downsampling module. Equipped with backward networks, we can conveniently and efficiently search the entire design space based on the prescribed



requirements to uncover the complex evolution of the transmission response as the geometric parameters change. It also provides a workable method for metasurface on-demand design and studying complicated light-matter interactions. It is worth noting that our method is inherently domain interpolative and only as accurate as the data that we feed into it. Thus, the model we have developed could be readily extended to not only multifunctional metasurfaces/meta-atoms, but various types of electromagnetic devices, such as optical antennas, microwave components, and integrated photonic devices. For example, by expanding transmission responses or using reflection responses instead, our model can be adopted to design reflective and refractive metasurfaces at arbitrary wavelengths. In this work, we restrict the meta-atoms pattern in a multiple nanofins form. Meta-atoms with more complicated structures can be designed and optimized as well by simply revising the database used here and choosing hyperparameters of our model.

4. CONCLUSION

In summary, we propose a new model of a deep neural network with a pretrained autoencoder for dielectric metasurface design. The model consists of four deep neural networks: simulator, encoder, decoder, and translator. By using this model, forward and backward designs can be realized in a businesslike and available way. The forward model can give phase and amplitude responses with an MAE of 0.03 in millisecond time scale, circumventing the computational burden required for numerical simulation. After about 80 times data reduction by the pretrained autoencoder, the backward model allows one to retrieve metasurfaces' geometric parameters from specific requirement responses, solving the on-demand problems. Furthermore, we try to reveal the physical mechanism behind the model. Activation mappings of the simulator are visualized by Grad-CAM, indicating that our model puts more attention on global features in the final decision-making process. Our model provides a novel approach for solving complex light– matter interaction such as multifunctional meta-atoms and broadband achromatic metalenses.

Funding. Fundamental Research Funds for the Central Universities, Nankai University (63201178); Natural Science Foundation of Tianjin City (18JCZDJC38200); Nankai University-Taishan University cooperative project (F1033851).

Disclosures. The authors declare no conflicts of interest.

[†]These authors contributed equally to this work.

REFERENCES

- S. Wang, P. C. Wu, V. Su, Y. Lai, M. Chen, H. Y. Kuo, B. H. Chen, Y. H. Chen, T. Huang, J. Wang, R. Lin, C. Kuan, T. Li, Z. Wang, S. Zhu, and D. P. Tsai, "A broadband achromatic metalens in the visible," Nat. Nanotechnol. **13**, 227–232 (2018).
- W. T. Chen, A. Y. Zhu, V. Sanjeev, M. Khorasaninejad, Z. Shi, E. Lee, and F. Capasso, "A broadband achromatic metalens for focusing and imaging in the visible," Nat. Nanotechnol. 13, 220–227 (2018).
- M. Khorasaninejad, A. Y. Zhu, C. Roques-Carmes, W. T. Chen, J. Oh, I. Mishra, R. C. Devlin, and F. Capasso, "Polarization-insensitive metalenses at visible wavelength," Nano Lett. 16, 7229–7234 (2016).
- T. Cai, G. Wang, X. Fu, J. Liang, and Y. Zhuang, "High-efficiency metasurface with polarization-dependent transmission and reflection properties for both reflect array and transmit array," IEEE Trans. Antennas Propag. 66, 3219–3224 (2018).
- G. Lee, G. Yoon, S. Lee, H. Yun, J. Cho, K. Lee, H. Kim, and J. R. B. Lee, "Complete amplitude and phase control of light using broadband holographic metasurfaces," Nanoscale 10, 4237–4245 (2018).

- X. Ni, A. V. Kildishev, and V. M. Shalaev, "Metasurface holograms for visible light," Nat. Commun. 4, 2807 (2013).
- F. Aieta, P. Genevet, M. A. Kats, N. Yu, R. Blanchard, Z. Gaburro, and F. Capasso, "Aberration-free ultrathin flat lenses and axicons at telecom wavelengths based on plasmonic metasurfaces," Nano Lett. 12, 4932–4936 (2012).
- Z. Liu, D. Zhu, S. P. Rodrigues, K. Lee, and W. Cai, "Generative model for the inverse design of metasurfaces," Nano Lett. 18, 6570–6576 (2018).
- O. Avayu, E. Almeida, Y. Prior, and T. Ellenbogen, "Composite functional metasurfaces for multispectral achromatic optics," Nat. Commun. 8, 14992 (2017).
- Q. Ma, G. D. Bai, H. B. Jing, C. Yang, L. Li, and T. J. Cui, "Smart metasurface with self-adaptively reprogrammable functions," Light Sci. Appl. 8, 98 (2019).
- 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," Adv. Mater. 26, 5031–5036 (2014).
- D. Sarwinda, R. H. Paradisaa, A. Bustamam, and P. Anggia, "Deep learning in image classification using residual network (ResNet) variants for detection of colorectal cancer," Procedia Comput. Sci. 179, 423–431 (2021).
- J. Zhang, Y. Xie, Q. Wu, and Y. Xia, "Medical image classification using synergic deep learning," Med. Image. Anal. 54, 10–19 (2019).
- A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," Appl. Soft Comput. **70**, 41–45 (2018).
- R. Kemker, R. Luu, and C. Kanan, "Low-shot learning for the semantic segmentation of remote sensing imagery," IEEE Trans. Geosci. Remote Sens. 56, 6214–6223 (2018).
- F. Abdurahman, K. A. Fante, and M. Aliy, "Malaria parasite detection in thick blood smear microscopic images using modified YOLOV₃ and YOLOV₄ models," BMC Bioinf. 22, 112 (2021).
- P. R. Wiecha, A. Arbouet, C. Girard, and O. L. Muskens, "Deep learning in nano-photonics: inverse design and beyond," Photon. Res. 9, B182–200 (2021).
- S. An, C. Fowler, B. Zheng, M. Y. Shalaginov, H. Tang, H. Li, L. Zhou, J. Ding, A. M. Agarwal, C. Rivero-Baleine, K. A. Richardson, T. Gu, and J. Hu, "A deep learning approach for objective-driven all-dielectric metasurface design," ACS Photon. 6, 3196–3207 (2019).
- W. Ma, F. Cheng, and Y. Liu, "Deep-learning-enabled on-demand design of chiral metamaterials," ACS Nano 12, 6326–6334 (2018).
- J. Jiang, D. Sell, S. Hoyer, J. Hickey, J. Yang, and J. A. Fan, "Freeform diffractive metagrating design based on generative adversarial networks," ACS Nano 13, 8872–8878 (2019).
- C. Qian, B. Zheng, Y. Shen, L. Jing, E. Li, L. Shen, and H. Chen, "Deep-learning-enabled self-adaptive microwave cloak without human intervention," Nat. Photonics 14, 383–390 (2020).
- J. Peurifoy, Y. Shen, L. Jing, Y. Yang, F. Cano-Renteria, B. G. DeLacy, J. D. Joannopoulos, and M. Tegmark, "Nanophotonic particle simulation and inverse design using artificial neural networks," Sci. Adv. 4, eaar4206 (2018).

- I. Malkiel, M. Mrejen, A. Nagler, U. Arieli, L. Wolf, and H. Suchowski, "Plasmonic nanostructure design and characterization via deep Learning," Light Sci. Appl. 7, 60 (2018).
- Z. Liu, L. Raju, D. Zhu, and W. Cai, "A hybrid strategy for the discovery and design of photonic structures," IEEE J. Emerging Sel. Top. Circuits Syst. 10, 126–135 (2020).
- M. V. Zhelyeznyakov, S. Brunton, and A. Majumdar, "Deep learning to accelerate scatterer-to-field mapping for inverse design of dielectric metasurfaces," ACS Photon. 8, 481–488 (2021).
- P. Mehran, A. Purang, and N. Robert, "Adaptive augmentation of medical data using independently conditional variational autoencoders," IEEE Trans. Med. Imaging 38, 2807–2820 (2019).
- H. Xu, "Generate faces using ladder variational autoencoder with maximum mean discrepancy (MMD)," J. Intell. Inf. Syst. 10, 108–113 (2018).
- A. J. Ollanik, J. A. Smith, M. J. Belue, and M. D. Escarra, "Highefficiency all-dielectric Huygens metasurfaces from the ultraviolet to the infrared," ACS Photon. 5, 1351–1358 (2018).
- S. Sarkar, V. Gupta, M. Kumar, J. Schubert, P. T. Probst, J. Joseph, and T. A. F. König, "Hybridized guided-mode resonances via colloidal plasmonic self-assembled grating," ACS Appl. **11**, 13752–13760 (2019).
- T. Kawashima, H. Yoshikawa, and S. Adachi, "Optical properties of hexagonal GaN," J. Appl. Phys. 82, 3528–3535 (1997).
- S. Logothetidis, J. Petalas, M. Cardona, and T. D. Moustakas, "Optical properties and temperature dependence of the interband transitions of cubic and hexagonal GaN," Phys. Rev. B 50, 18017–18029 (1994).
- D. T. Pierce and W. E. Spicer, "Electronic structure of amorphous Si from photoemission and optical Studies," Phys. Rev. B 5, 3017–3029 (1972).
- 33. K. Luke, Y. Okawachi, M. R. E. Lamont, A. L. Gaeta, and M. Lipson, "Broadband mid-infrared frequency comb generation in a Si_3N_4 microresonator," Opt. Lett. **40**, 4823–4826 (2015).
- B. Wang, F. Dong, Q. Li, D. Yang, C. Sun, J. Chen, Z. Song, L. Xu, W. Chu, Y. Xiao, Q. Gong, and Y. Li, "Visible-frequency dielectric metasurfaces for multiwavelength achromatic and highly dispersive holograms," Nano Lett. 16, 5235–5240 (2016).
- P. Prahs, V. Radeck, C. Mayer, Y. Cvetkov, N. Cvetkova, H. Helbig, and D. Märker, "OCT-based deep learning algorithm for the evaluation of treatment indication with anti-vascular endothelial growth factor medications," Graff. Arch. Clin. Exp. 256, 91–98 (2018).
- J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: the all convolutional net," in *ICLR* (2015).
- C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: an empirical study of their impact to deep learning," Multimedia Tools Appl. **79**, 12777–12815 (2020).
- H. B. McMahan, "Follow-the-regularized-leader and mirror descent: equivalence theorems and L1 regularization," in *Proceedings of Machine Learning Research (PMLR)* (2011), Vol. 15, pp. 525–533.
- E. Orhan and X. Pitkow, "Skip connections eliminations singularities," in *ICLR* (2018).
- R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: visual explanations from deep networks via gradient-based localization," Int. J. Comput. Vis. **128**, 336–359 (2020).