



Research article

Sunae So and Junsuk Rho*

Designing nanophotonic structures using conditional deep convolutional generative adversarial networks

<https://doi.org/10.1515/nanoph-2019-0117>

Received April 16, 2019; revised May 20, 2019; accepted May 22, 2019

Abstract: Data-driven design approaches based on deep learning have been introduced in nanophotonics to reduce time-consuming iterative simulations, which have been a major challenge. Here, we report the first use of conditional deep convolutional generative adversarial networks to design nanophotonic antennae that are not constrained to predefined shapes. For given input reflection spectra, the network generates desirable designs in the form of images; this allows suggestions of new structures that cannot be represented by structural parameters. Simulation results obtained from the generated designs agree well with the input reflection spectrum. This method opens new avenues toward the development of nanophotonics by providing a fast and convenient approach to the design of complex nanophotonic structures that have desired optical properties.

Keywords: nanophotonics; inverse design; conditional deep convolutional generative adversarial network; deep learning.

1 Introduction

Progress in nanophotonics has yielded numerous extraordinary optical properties such as cloaking objects [1, 2], imaging beyond the diffraction limit [3, 4], and negative

refractive index [5, 6]. In nanophotonics, a sub-wavelength antenna interacts with light, so precisely designed components can provide useful functionalities. Although several systematic design approaches for desired optical behaviors have been introduced [7–10], the procedure for inverse design of nanophotonic structures mostly still relies on a laborious method of optimization. Such a conventional design method requires time-consuming iterative simulations.

Recently, data-driven design approaches have been proposed to overcome this problem. These approaches use artificial neural networks (NNs) to design nanophotonic structures [11–14]. Previous studies first set the shape, such as multilayers [11] or H-antenna [12], of the structures to be predicted, then trained NNs provide the output structural parameters that achieve the desired optical properties. Once the NNs are trained, they provide the corresponding design parameters without additional iterative simulations. Such attempts have greatly reduced the effort and computational costs of designing nanophotonic structures. So far, these approaches have only been applied to conditions in which the basic structures are predefined where only the structural parameters are predictable. Most recently, a generative adversarial network (GAN) model has been used to inversely design metasurfaces in order to provide arbitrary patterns of the unit cell structure [15].

In this article, we provide the first use of a conditional deep convolutional generative adversarial network (cDCGAN) [16] to design nanophotonic structures. cDCGAN is a recently developed algorithm to solve the instability problem of GAN, and provides a very stable Nash equilibrium solution. The generated designs are presented as images, so they provide essentially any arbitrary possible design for the desired optical properties that are not limited to specific structures. Our research provides designs of a 64×64 pixel probability distribution function (PDF) in a domain size of $500 \text{ nm} \times 500 \text{ nm}$, which allows $2^{64 \times 64}$ degrees of freedom of design.

*Corresponding author: **Junsuk Rho**, Department of Mechanical Engineering, Pohang University of Science and Technology (POSTECH), Pohang, South Korea; and Department of Chemical Engineering, Pohang University of Science and Technology (POSTECH), Pohang, South Korea, e-mail: jsrho@postech.ac.kr
<https://orcid.org/0000-0002-2179-2890>

Sunae So: Department of Mechanical Engineering, Pohang University of Science and Technology (POSTECH), Pohang, South Korea

2 Results and discussion

2.1 Deep-learning procedure

For deep learning, we first collect a dataset consisting of 10,150 silver antennae with six representative shapes (circle, square, cross, bow-tie, H-shaped, and V-shaped). Each entry in the dataset is composed of a reflection spectrum with 200 spectral points and its corresponding cross-sectional structural design with a 64×64 pixel image. Sixty-four coarse meshes are used for both the x - and y -directions for the simple calculation. The cross-sectional structure designs are prepared in the form of images with a physical domain size of $500 \text{ nm} \times 500 \text{ nm}$. The antenna of 30 nm thickness is placed on a 50-nm MgF_2 spacer, a 200-nm silver reflector, and a silicon substrate (Figure 1). To obtain the reflection spectrum of each structure, a finite-difference time-domain (FDTD) electromagnetic simulation is performed using the commercial program FDTD Lumerical Solutions. The simulation is conducted over the whole spectral range from $f=250\text{--}500 \text{ THz}$, and 200 spectral points are extracted. Periodic boundary conditions with the periodicity of 500 nm are used along the x - and y -directions, and perfectly matched boundary conditions are used along the z -direction. At each simulation, y -polarized light is incident on the antenna with 0 incident angle. The current deep-learning setting solves the designing structure problem in a fixed physical domain

and fixed wavelength. Designing structures with different periodicity or wavelengths requires additional data-collection or deep-learning procedures.

As a next step, we implement a deep-learning algorithm using the Pytorch framework. Artificial intelligence has revolutionized the field of computer vision recently [17, 18]. A convolutional neural network (CNN) [18, 19] is among the most widely used techniques, inspired by the natural visual perception mechanism of the human brain. A CNN uses convolution operators to extract features from the input data, which are usually images. It greatly increases the efficiency of image recognition, because every channel extracts important features of the images. On the other hand, the development of GAN has resulted in major progress in computer vision [20]. A GAN is composed of a generator network (GN) that generates the images and a discriminator network (DN) that distinguishes the generated images from real images. GN is trained to generate authentic images to deceive the DN, and DN is trained not to be deceived by the GN. The two networks compete with each other in every training step; ultimately, the competition leads to mutual improvement of each network, so that GN can generate higher quality realistic images than when it learns alone. DCGAN combines the idea of a CNN and GAN to provide a very stable Nash equilibrium solution [16]. We employed the cDCGAN algorithm with a condition [21], which is the input reflection spectrum in this case.

The cDCGAN architecture to design nanophotonic structures is presented in Figure 2. cDCGAN is composed

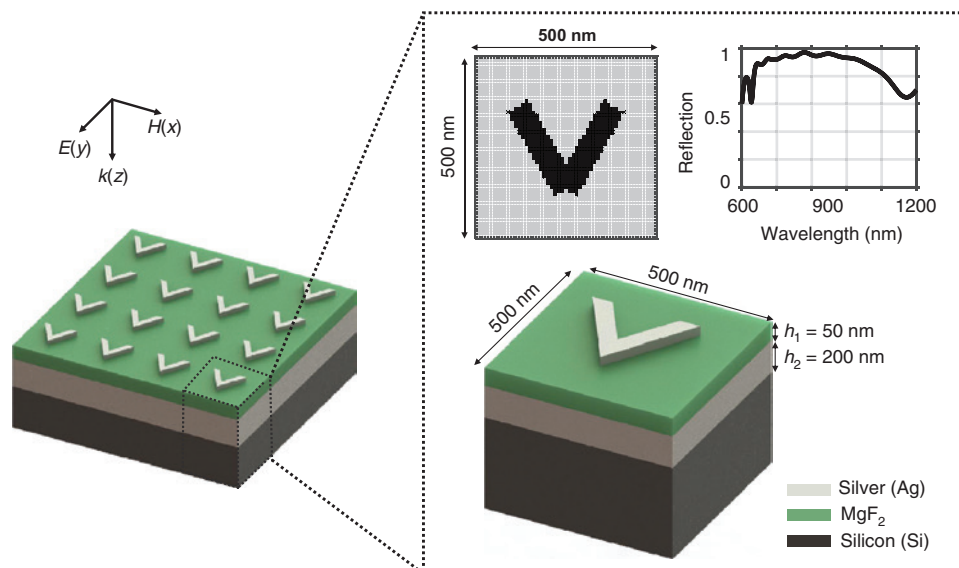


Figure 1: Schematic of data preparation for deep learning.

Each entry in the dataset is composed of the reflection spectrum obtained from FDTD simulation and its corresponding cross-sectional structural design.

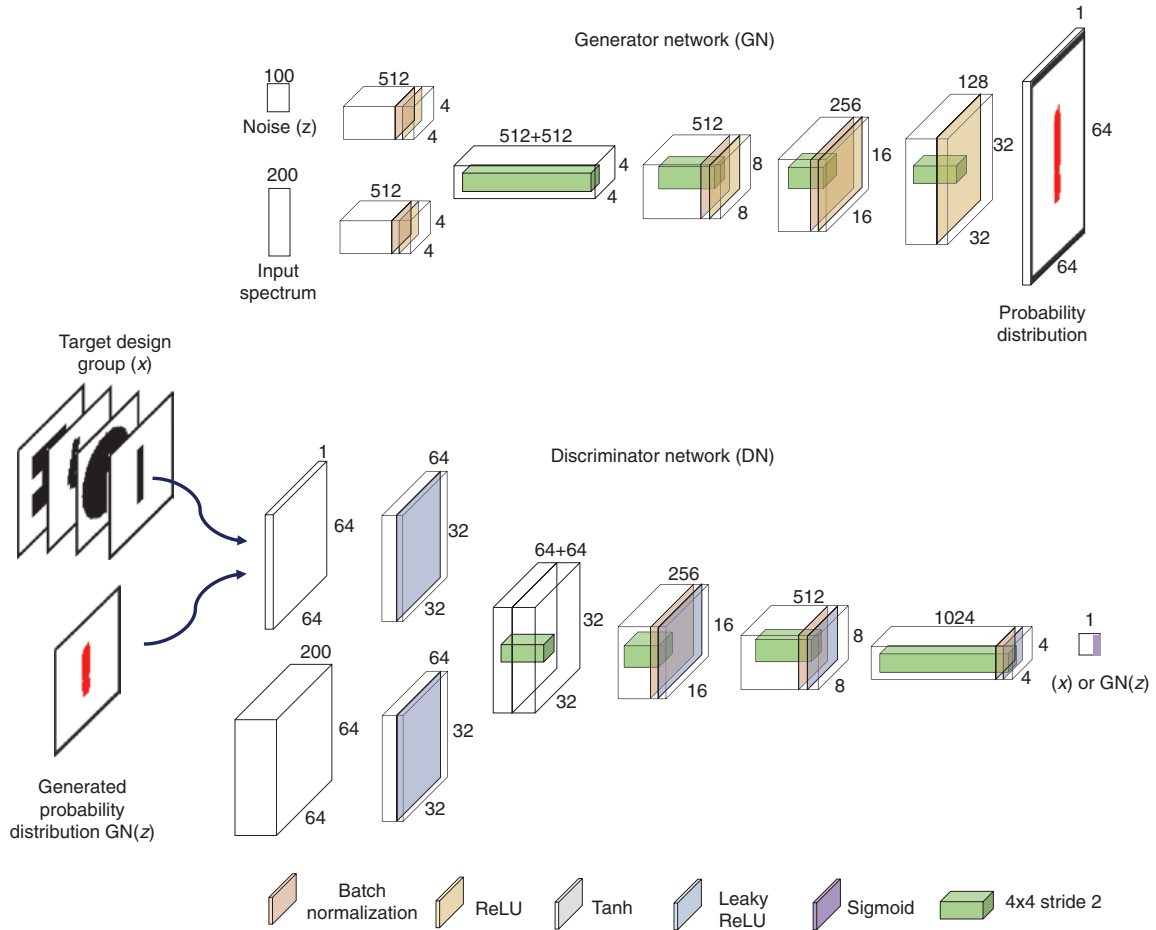


Figure 2: Schematic of the cDCGAN architecture to suggest the designs of structures.

GN is composed of a transposed CNN to generate the structural images, and DN is composed of conventional CNN to distinguish target structural designs from the generated designs. Each layer introduces nonlinear activation functions (ReLU, Tanh, Leaky ReLU, and Sigmoid) according to the guideline of Radford et al. [16].

of two networks: a GN that generates structural cross-sectional images, and a DN that distinguishes the generated images given by the GN from user-given target designs group. GN is composed of four transposed CNN layers consisting of 1024, 512, 256, 128, and 1 channel, respectively; DN is a CNN with four layers. GN takes inputs of both the 100×1 size random noise (z) and the 200×1 size input spectrum. GN provides a probability distribution function (PDF) of the antenna as output, which is generated from the random noise. The input spectrum guides the GN to generate a PDF that has such optical properties. On the contrary, DN takes input as a structural image from either a user-provided target designs group (x) or the generated PDF images by GN, $GN(z)$. DN plays the role of discriminating $GN(z)$ from the target designs group. Ultimately, GN and DN are simultaneously trained competitively: GN is trained to generate an authentic structural design to deceive the DN, and DN

is trained to distinguish target designs from the design generated by GN. Mathematically, GN and DN are trained in the direction to minimize or maximize the objective function:

$$\min_{GN} \max_{DN} l(DN, GN) = E_{x \leftarrow P_{data}(x)} [\log DN(x)] + E_{z \leftarrow P_z(z)} [\log(1 - DN(GN(z)))], \quad (1)$$

where $DN(x)$ represents the probability of a structural image coming from the target design group (x), and $DN(GN(z))$ represents that coming from generated design $G(z)$ by GN. In terms of DN, the network is trained to give maximized expectation values of E with $E_{x \leftarrow P_{data}(x)} [\log DN(x)]$ for a given image coming from the target design, and $E_{z \leftarrow P_z(z)} [\log(1 - DN(GN(z)))]$ for a given image generated by GN. On the other hand, GN is trained to give minimized expectation values to deceive DN. This

adversarial training allows GN to generate high-quality structural images.

In addition to adversarial training, we further modify the loss function of GN [22, 23] in the cDCGAN to fit our problem to

$$l_{\text{GN}} = (1 - \rho) \times l_{\text{GN,design}} + \rho \times l_{\text{GN,adv}}, \quad (2)$$

where $l_{\text{GN,design}}$ is the design loss, $l_{\text{GN,adv}}$ is the adversarial loss defined in Eq. (1), and ρ is the ratio of the adversarial loss. The design loss is introduced to explicitly guide the GN to generate structural images well. It directly measures the quantitative difference between two probability distributions of the target design (x_i) and the generated design (\hat{x}_i) using a binary cross-entropy criterion:

$$l_{\text{GN,design}} = -(x_i \log \sigma(\hat{x}_i) + (1 - x_i) \log(1 - \sigma(\hat{x}_i))), \quad (3)$$

where a σ is a Sigmoid function.

We optimized ρ to make the GN generate high-quality realistic designs. For a low ρ , a competition effect cannot be expected, whereas a high ρ can cause confusion in the learning process. Therefore, an appropriate value of $\rho = 0.5$ was chosen to maximize the ability of GN to produce convincing structural designs. During each training step, the network is trained to optimize the weights to describe the mapping between the input spectrum and the PDF (see Supporting Information for details about deep-learning procedure and network optimization).

After training, cDCGAN suggests designs on a 64×64 pixel PDF $p(i, j)$, which represents the probability that a silver antenna exists at the location (i, j) . To reduce the PDF to a binary image representing the existence of antennae at the locations, we employed a post-processing step according to Otsu [24]. This method determines the binary threshold t that minimizes the intra-class variance σ_ω^2 of the black and white pixels as

$$\sigma_\omega^2(t) = \omega_0(t) \sigma_0^2(t) + \omega_1(t) \sigma_1^2(t), \quad (4)$$

where ω_0 and ω_1 represent the weights for the probabilities of two classes separated by t , σ_0^2 is the variance of the black pixels, and σ_1^2 is the variance of the white pixels. In summary, for a given reflection spectrum, cDCGAN produces a PDF which is then converted to a binary design image in the post-processing step. At each training step, 2000 validation samples are used to validate the trained network. The average loss of the validation set converged to 5.564×10^{-3} after 1000 training steps. Using a single GPU of GTX 1080-Ti, the training a network for one epoch

requires about 4 min. However, once a network is trained, the trained network can generate a design for a desired spectrum within 3 s.

2.2 Network evaluation

The trained cDCGAN is evaluated on test data that were not used in the previous training or validation steps. Randomly chosen test results are shown in Figure 3. Target designs of various nanophotonic antennae (upper-right panel in Figure 3) and the corresponding suggested PDFs (lower-right panel in Figure 3) show good qualitative agreement. For a quantitative evaluation of the suggested PDFs, FDTD simulation based on those suggested designs were conducted. The PDFs were converted to binary designs to be imported into the simulations. Reflection spectra of the suggested images agree well with the given input spectra. We introduce a mean absolute error (MAE) criterion

$$l_{\text{error}} = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i|, \quad (5)$$

to quantitatively measure the average error of the reflection spectrum per spectral point between the FDTD simulation result (Y_i) obtained from the suggested design and the input spectrum (\hat{Y}_i) that is originally fed into the network. The average MAE error of 12 test samples is 0.0322, which supports that the trained network can essentially provide an appropriate structural design that has the desired reflection spectrum. Interestingly, even if the antennae have similar shapes, there can be a discrepancy between the predicted and input spectra, as shown in the second column and bottom row of Figure 3. This is due to the small artifacts in the provided PDF. These artifacts can be removed using a different image filter in the post-processing step (see Supporting Information for the effect on post-processing).

We also test our cDCGAN with completely new structures of triangle and star antennae whose shapes never existed in the training and validation dataset (Figure 4A, B). cDCGAN generated new designs that had distorted forms of the antennae, which were used for training. The results imply that our cDCGAN can suggest any designs, unconstrained by structural parameters. The generated images are different from the target designs, but the generated reflection spectra are similar to the input reflection spectra. This is because of the non-uniqueness of the correlation between the optical properties and designs: several different designs can have the same optical property. Among the several possible designs, the generated

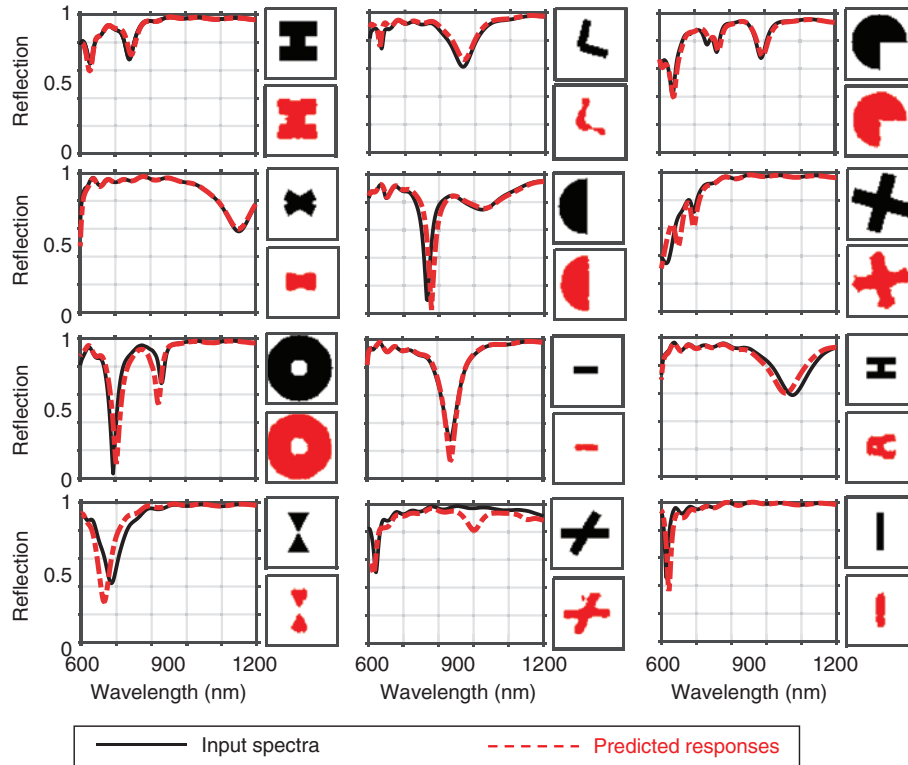


Figure 3: Twelve examples of cDCGAN-suggested images and their simulation results. Each panel is composed of reflection spectra and their corresponding structural cross-sectional images. The upper-right structural images are the target designs (black), and the lower-right images are the suggested images by cDCGAN (red). The left spectra show the desired input spectra (black solid lines) that we fed into the network and the predicted responses obtained from the suggested designs (red dotted lines).

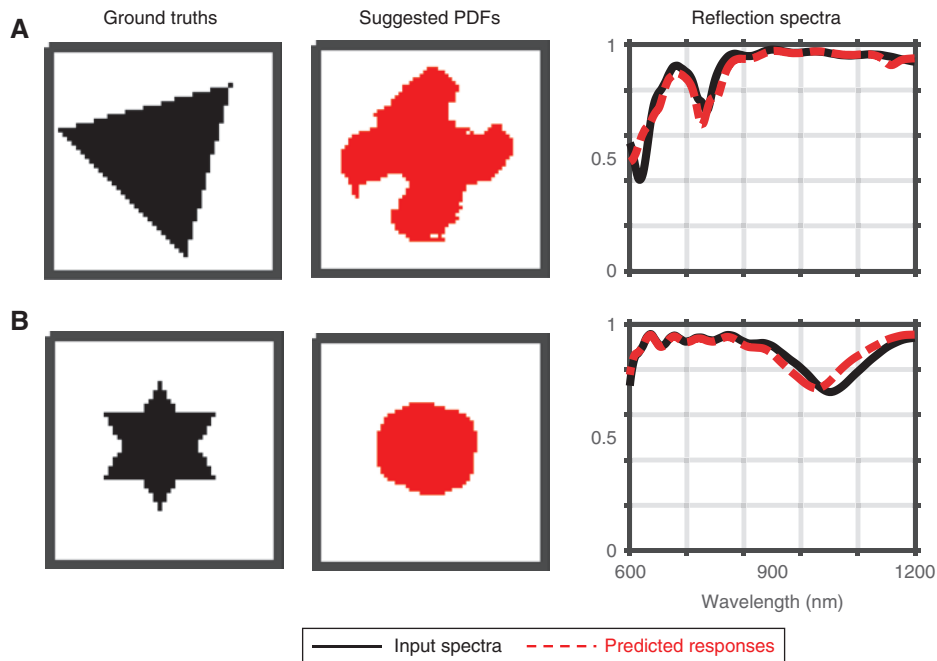


Figure 4: cDCGAN suggestion results of completely new structures. New structures of (A) triangle and (B) star shape antennae, whose shapes have never been used in training, are used. The first columns are the target of desired designs and the second columns are PDFs suggested by cDCGAN. The last columns show the reflection spectra of the input (black solid lines) and the simulation result obtained from the suggested designs (red dotted lines), respectively.

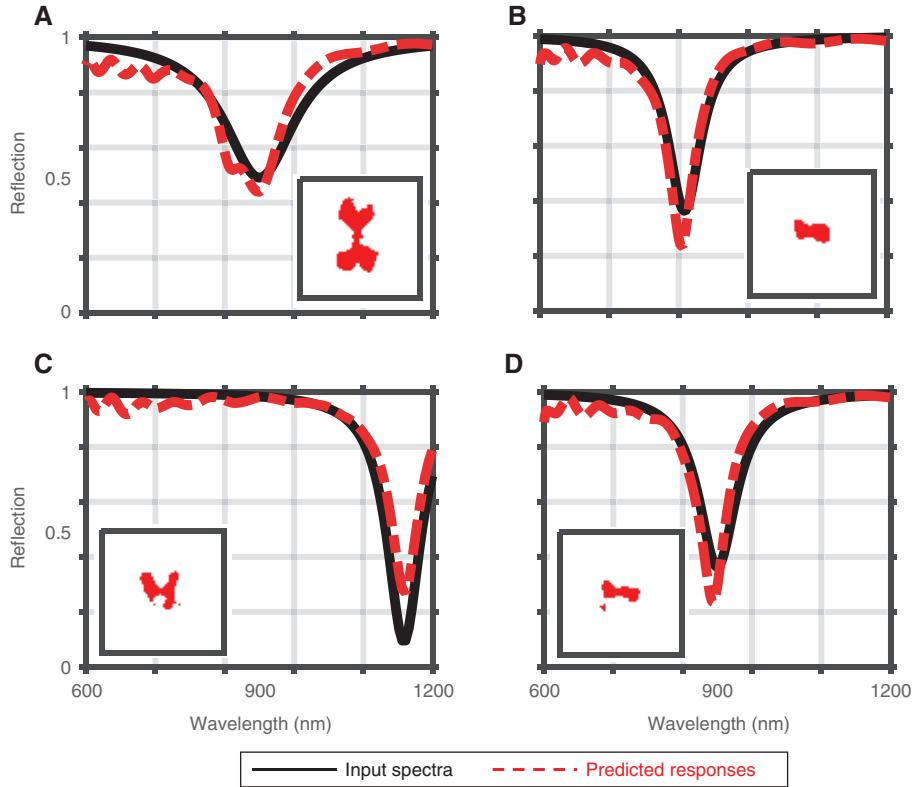


Figure 5: cDCGAN suggestion results of hand-drawn spectra.

Lorentzian-like function of (A) $a=120$, $b=900$, $c=150$, (B) $a=70$, $b=850$, $c=70$, (C) $a=100$, $b=1150$, $c=70$, and (D) $a=80$, $b=900$, $c=80$. The inset in each figure shows the suggested PDFs by cDCGAN. Black solid lines indicate the input Lorentzian-like function that we fed to the network, and red dotted lines indicate the simulation results obtained from the suggested PDFs.

results are most likely to be found in areas that do not deviate much from the trained dataset space.

Finally, our cDCGAN was further tested with randomly generated, hand-drawn spectra of the Lorentzian-like function

$$f(x) = \frac{2a}{\pi} \frac{c}{4(x-b)^2 + c^2}, \quad (6)$$

with three cases: (A) $a=120$, $b=900$, $c=150$, (B) $a=70$, $b=850$, $c=70$, (C) $a=100$, $b=1150$, $c=70$, and (D) $a=80$, $b=900$, $c=80$. For each case, the generated image and its corresponding reflection spectrum are shown in Figure 5. The MAE of the reflection spectrum for the four examples are (a) 0.0496, (b) 0.0396, (c) 0.0409, and (d) 0.0408. The predicted responses show reasonably good agreement with the input spectrum in terms of the overall behavior. Most interestingly, the generated images (insets in Figure 5A–D) deviate much from the shapes that are used for training. Such extraordinary structural shapes are not constrained by predefined structures and are even not describable; this is the key advantage of our method over previous ones which can only suggest given structural parameters. The

results also indicate that cDCGAN actually learns well the correlation between structural designs and their overall optical responses and hence can be widely used to systematically design nanophotonic structures.

3 Conclusion

In conclusion, we demonstrated the first use of cDCGAN to design nanophotonic structures. The two networks of GN and DN in cDCGAN competitively learn to suggest appropriate designs of nanophotonic structures that have the desired optical properties of reflection. Our cDCGAN is not limited to suggesting predefined structures but can also generate new designs. It has numerous design possibilities with $2^{64 \times 64} = 2^{4096}$ degrees of freedom. The 7.8 nm pixel size used in the current design makes it very difficult to fabricate the suggested design within this resolution. This fabrication difficulty can be overcome by increasing the pixel size up to the feasible fabrication scale of 20–30 nm. Here, we limited the input to a single reflection spectrum. This method can also be extended in the

diffraction regime, where additional diffraction orders may appear, if multiplexed input spectra with several diffraction orders are used. Because of the limitation on the used training dataset, it is not always possible to generate structural images with extraordinary reflection spectra. Such weakness may be overcome by collecting additional data that can represent extraordinary reflection spectra. Although our examples set the thickness of each layer and the material type of the antenna, they can also be added as output parameters to be suggested. This modification would allow artificial intelligence to be used to design nanophotonic devices completely independently and would thereby greatly reduce the time and computational cost of designing them manually. We believe that our research findings will lead to the rapid development of nanophotonics by solving the main problem of designing the structures.

4 Supplementary Material

Supplementary Material is available online on the journal's website or from the author.

Acknowledgment: This work was financially supported by the National Research Foundation grants (NRF-2017R1E1A1A03070501, NRF-2019R1A2C3003129, CAMM-2019M3A6B3030637, NRF-2018M3D1A1058998, and NRF-2015R1A5A1037668) funded by the Ministry of Science and ICT, Korea. S.S. acknowledges a global Ph.D. fellowship (NRF-2017H1A2A1043322) from the NRF-MSIT, Korea.

References

- [1] Cai W, Chettiar UK, Kildishev AV, Shalaev VM. Optical cloaking with metamaterials. *Nat Photon* 2007;1:224.
- [2] Schurig D, Mock J, Justice B, et al. Metamaterial electromagnetic cloak at microwave frequencies. *Science* 2006;314:977–80.
- [3] Zhang X, Liu Z. Superlenses to overcome the diffraction limit. *Nat Mater* 2008;7:435.
- [4] Betzig E, Trautman JK. Near-field optics: microscopy, spectroscopy, and surface modification beyond the diffraction limit. *Science* 1992;257:189–95.
- [5] Shelby RA, Smith DR, Schultz S. Experimental verification of a negative index of refraction. *Science* 2001;292:77–9.
- [6] Valentine J, Zhang S, Zentgraf T, et al. Three-dimensional optical metamaterial with a negative refractive index. *Nature* 2008;455:376.
- [7] Matlack KH, Serra-Garcia M, Palermo A, Huber SD, Daraio C. Designing perturbative metamaterials from discrete models. *Nat Mater* 2018;17:323.
- [8] Kalinin SV, Sumpter BG, Archibald RK. Big-deep-smart data in imaging for guiding materials design. *Nat Mater* 2015;14:973.
- [9] Coulais C, Teomy E, de Reus K, Shokef Y, van Hecke M. Combinatorial design of textured mechanical metamaterials. *Nature* 2016;535:529.
- [10] Molesky S, Lin Z, Piggott AY, Jin W, Vucković J, Rodriguez AW. Inverse design in nanophotonics. *Nat Photon* 2018;12:659.
- [11] Liu D, Tan Y, Khoram E, Yu Z. Training deep neural networks for the inverse design of nanophotonic structures. *ACS Photon* 2018;5:1365–9.
- [12] Malkiel I, Nagler A, Mrejen M, Arieli U, Wolf L, Suchowski H. Deep learning for design and retrieval of nano-photon structures. 2017; arXiv preprint, arXiv:1702.07949.
- [13] Kabir H, Wang Y, Yu M, Zhang QJ. Neural network inverse modeling and applications to microwave filter design. *IEEE Trans Microw Theory Tech* 2008;56:867–79.
- [14] Ma W, Cheng F, Liu Y. Deep-learning enabled on-demand design of chiral metamaterials. *ACS Nano* 2018;12:6326–34.
- [15] Liu Z, Zhu D, Rodrigues S, Lee KT, Cai W. A generative model for the inverse design of metasurfaces. *Nano Lett* 2018;18:6570–6.
- [16] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015; arXiv preprint, arXiv:1511.06434.
- [17] Russell SJ, Norvig P. Artificial intelligence: a modern approach. Malaysia: Pearson Education Limited, 2016.
- [18] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 2012:1097–105.
- [19] Lawrence S, Giles CL, Tsoi AC, Back AD. Face recognition: a convolutional neural-network approach. *IEEE Trans Neural Netw* 1997;8:98–113.
- [20] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. *Adv Neural Inf Process Syst* 2014:2672–80.
- [21] Mirza M, Osindero S. Conditional generative adversarial nets. 2014; arXiv preprint, arXiv:1411.1784.
- [22] Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA. Context encoders: feature learning by inpainting. *Proc IEEE Conf Comput Vis Pat Recogn* 2016:2536–44.
- [23] Isola P, Zhu JY, Zhou T, Efros AA. Image-to-image translation with conditional adversarial networks. 2017; arXiv preprint.
- [24] Otsu N. A threshold selection method from gray-level histograms. *IEEE Trans Syst Man Cybern* 1979;9:62–6.

Supplementary Material: The online version of this article offers supplementary material (<https://doi.org/10.1515/nanoph-2019-0117>).