Research article

Thomas Christensen, Charlotte Loh, Stjepan Picek*, Domagoj Jakobović, Li Jing, Sophie Fisher, Vladimir Ceperic, John D. Joannopoulos and Marin Soljačić*

Predictive and generative machine learning models for photonic crystals

https://doi.org/10.1515/nanoph-2020-0197
Received March 15, 2020; accepted May 14, 2020; published online June 29, 2020

Abstract: The prediction and design of photonic features have traditionally been guided by theory-driven computational methods, spanning a wide range of direct solvers and optimization techniques. Motivated by enormous advances in the field of machine learning, there has recently been a growing interest in developing complementary data-driven methods for photonics. Here, we demonstrate several predictive and generative data-driven approaches for the characterization and inverse design of photonic crystals. Concretely, we built a data set of 20,000 two-dimensional photonic crystal unit cells and their associated band structures, enabling the training of supervised learning models. Using these data set, we demonstrate a high-accuracy convolutional neural network for band structure prediction, with orders-of-magnitude speedup compared to conventional theory-driven solvers. Separately, we demonstrate an approach to high-throughput inverse design of photonic crystals via generative adversarial networks, with the design goal of substantial transverse-magnetic band gaps. Our work highlights photonic crystals as a natural application domain and test bed for the development of data-driven tools in photonics and the natural sciences.

Keywords: generative models; inverse design; machine learning; neural networks; photonic crystals.

1 Introduction

The confluence of an exceptional abundance of data and computational resources has enabled techniques of machine learning (ML), especially deep neural networks [1, 2], to revolutionize fields across computer science, ranging from image analysis [3–6] and natural language processing [7–10] to decision making [11, 12]. Spurred by these gains, there has been a surge of interest in applying ML techniques to the natural sciences, e.g. in physics [13–18], chemistry [19–21], and material science [22–24]. Traditionally, these disciplines have been dominated by theory-driven computational tools: while extraordinarily varied, each such technique is essentially the result of a series of formal reductions and controllable approximations—e.g. discretizations, expansions, or probabilistic averaging—systematically applied to a known theoretical framework. In data-driven approaches, by contrast, a large number of numerical weights, jointly parameterizing a computational neural network, are tuned to minimize an error measure across a specific or dynamically explored (supervised or active/semi-supervised learning) labeled data space.

The field of photonics—the study of electromagnetic properties of (sub)wavelength-scale material structures—is an appealing area for the application and development of new data-driven approaches. Specifically, data on photonic systems can be generated in large quantities by numerical means, owing to a large and mature suite of computational tools, covering finite-element, boundary-element, finite-difference or discontinuous time-domain, and spectral methods [25]. Each enables high-accuracy solutions of the Maxwell equations, e.g. subject to spatially varying material response functions such as the

***Corresponding authors: Stjepan Picek**, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, E-mail: s.picek@tudelft.nl; and **Marin Soljačić**, Department of Physics, Massachusetts Institute of Technology, Cambridge, MA, USA, E-mail: soljacic@mit.edu
Thomas Christensen, Li Jing, Sophie Fisher, Vladimir Ceperic and John D. Joannopoulos: Department of Physics, Massachusetts Institute of Technology, Cambridge, MA, USA, E-mail: tchr@mit.edu (T. Christensen), jingli9111@gmail.com (L. Jing), sefisher@mit.edu (S. Fisher), vceperic@gmail.com (V. Ceperic), joannop@mit.edu (J.D. Joannopoulos). https://orcid.org/0000-0002-9131-2717 (T. Christensen)
Charlotte Loh: Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA, E-mail: cloh@mit.edu
Domagoj Jakobović: Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia, E-mail: domagoj.jakobovic@fer.hr

permittivity $\varepsilon(\mathbf{r})$. Provided the assumed material response and geometric features of the underlying structures are accurate, such calculations generally agree extremely well with optical measurements, resembling, effectively, "numerical experiments" (in contrast to e.g. electronic structure calculations that typically exploit physical approximations, i.e. not merely a truncated basis, to overcome the computational challenges posed by many-body electron–electron interactions). This makes photonic systems ideal test beds for exploring the applications of data-driven techniques in realistic physical systems; and for developing new ML techniques for the natural sciences in general.

Already, several studies have explored the application of ML techniques to photonics: neural networks have been used to accurately predict optical scattering by multilayer nanoparticles [26], far- [27] and near-field [28] spectral response of plasmonic nanostructures, topological properties [29–31], and transmission spectra of dielectric metamaterials and metasurfaces [32–35]. There has also been a growing interest in the study of generative models [36–38], i.e. models that learn the underlying distribution of the data rather than simply "discriminating" the target values given a certain input, aiming to complement more conventional techniques of optimization and inverse design, such as via gradient-based [39, 40] or evolutionary methods [41, 42]. While trained neural networks can also accelerate traditional inverse design by affording a cheap gradient calculation via backpropagation [26] or simply a cheap evaluation [32], multiple iterations are often needed to find a good candidate, and backpropagating costs through a large network can be computationally challenging. Generative models [31] offer an alternative approach that sidesteps these challenges and additionally provides the flexibility of choosing among multiple suitable design candidates.

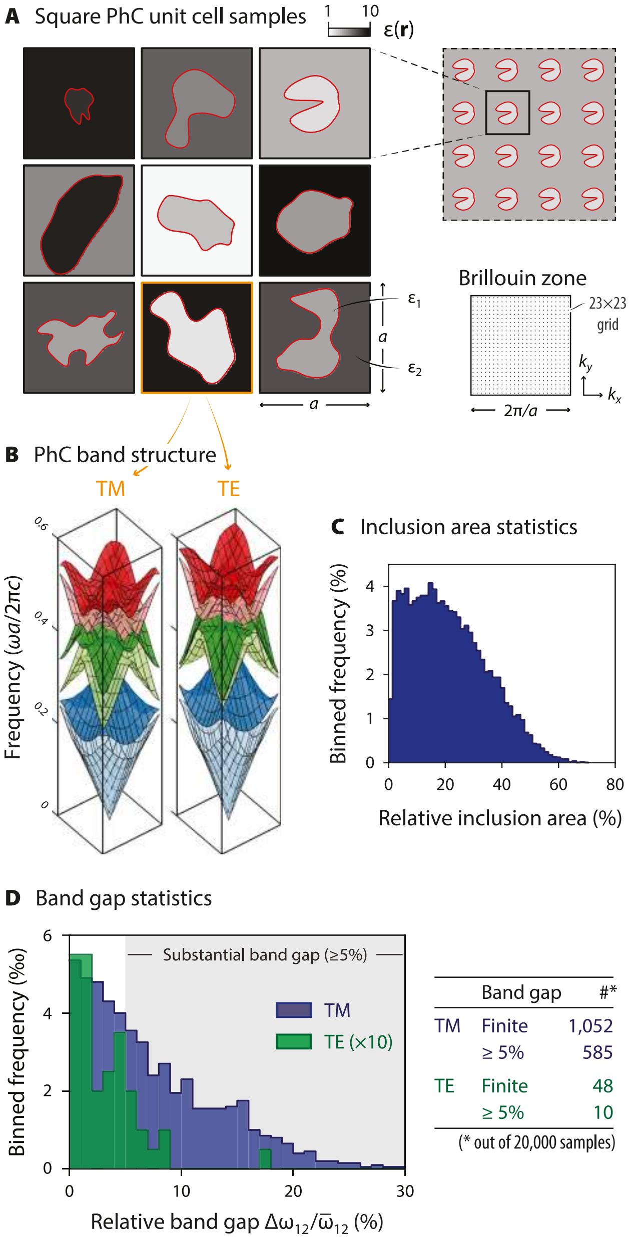
Here, we report several examples of data-driven ML techniques applied to photonic crystals (PhCs) [43, 44], that is, periodic wavelength-scale structures of dielectric material. We exploit the maturity of conventional computational approaches for PhCs to generate a data set suitable for supervised learning of 20,000 distinct two-dimensional (2D) PhCs. As a first application of these data set, we train a convolution neural network to perform band structure prediction. The trained network is highly accurate (mean test error of ~0.6%) and, once trained, orders of magnitude faster than conventional theory-based approaches. Following this, we explore two applications of generative models for data-driven inverse design of PhCs with a large band gap. In both cases, we find that a high-fidelity generative model can be trained using just ~1 000 data samples. Our results establish PhCs as a natural test bed for ML techniques applied to scientific problems and demonstrate that both forward and inverse problems in PhC-design are amenable to data-driven approaches.

2 Methods and results

2.1 Photonic crystal data set

PhCs are characterized by a periodically varying permittivity $\varepsilon(\mathbf{r})$, and the design domain is consequently restricted to a single nontrivial unit cell Ω whose tiling makes up the PhC's structure (Figure 1A). For simplicity and concreteness, we restrict our attention to 2D square lattices with two material components. Each material occupies a sub-region Ω_i of Ω, such that $\Omega_1 \cup \Omega_2 = \Omega$, with a resulting "two-tone" permittivity profile $\varepsilon(\mathbf{r}) = \begin{cases} \varepsilon_1, \mathbf{r} \in \Omega_1 \\ \varepsilon_2, \mathbf{r} \in \Omega_2 \end{cases}$. For lossless and isotropic materials, ε_i (as well as the PhC's allowed eigenfrequencies) are real quantities. As a result, each PhC is effectively characterized by a single "gray-scale image" of $\varepsilon(\mathbf{r})$. We generated 20,000 such two-tone, square unit cells. The two disjoint regions Ω_i were defined by their boundary region (Figure 1A), which in turn was procedurally generated by casting 2–8 random ellipses sequentially near each other's periphery, then spanning, smoothing, and centering an enclosing hull, and finally randomly scaling and orienting the resulting boundary. This produces unit cells that are relatively simple geometrically, host just a single inclusion, have no strongly divergent feature scales, and so exemplify realistically fabricable design candidates. We note that stricter constraints could be imposed to align more closely with experimental capabilities (minimum feature sizes could e.g. be ensured by post-processing generated inclusions with standard threshold projection techniques from topology optimization [45, 46]). Nevertheless, to retain a sufficiently varied training set we do not pursue such additional constraints here [47]. The permittivities ε_i were each drawn uniformly from the range $[1, 10]$, roughly spanning the range attainable in transparent materials in the visible spectrum (e.g. at a wavelength of 700 nm, the permittivity of air, silicon nitride, and silicon carbide is approximately 1,4.1, and 6.8, respectively).

For each unit cell, we computed the PhC band structure of the lowest six bands using the free MIT Photonics Bands (MPB) software [48] using 64 × 64 plane waves (equivalent, effectively, to a 64 × 64 spatial resolution). Each unit cell takes ~2 min on a single core of a 1.6 GHz Core i5-8250U CPU. The calculations are highly converged and



	Band gap	#*
TM	Finite	1,052
	≥ 5%	585
TE	Finite	48
	≥ 5%	10

(* out of 20,000 samples)

Figure 1: Photonic crystal data set. We generated a data set of 20,000 square 2D PhC unit cells, each consisting of a smooth, centered inclusion of permittivity ε_1 in a background permittivity ε_2 with $\varepsilon_i \in [1, 10]$. (A) Several representative unit cells and the BZ grid-sampling used in the calculation of band structures. (B) The TM and TE band structures of the PhC highlighted in orange in (A). (C) The generated unit cells predominately feature inclusions occupying less than half the unit cell, as illustrated by a histogramming of the relative inclusion areas across the data set. (D) TM band gaps between bands 1 and 2 consequently occur much more frequently than TE gaps, as TE gaps mainly arise in “filamentory” networks, corresponding to large relative inclusion areas.

accurate: the mean fractional deviation per band between calculations at resolutions of 64 × 64 and 32 × 32 is ~ 0.1‰, averaged over all unit cells. Figure 1B shows a set of example band structures, split into the transverse magnetic and electric (TM and TE) polarizations: it consists of the set of eigen frequencies $\omega_{n\mathbf{k}}$ indexed over band numbers $n = 1, 2, \ldots, 6$ and wave vectors $\mathbf{k}$ restricted to the Brillouin zone (BZ). For a square lattice of (arbitrary) side length a, the BZ is $[-\pi/a, \pi/a) \times [-\pi/a, \pi/a)$. Since the generated unit cells generically have no exact spatial symmetries, the band structures cannot exhibit any stable band-crossings, allowing a simple sorting of bands by their frequency alone, i.e. $\omega_{n\mathbf{k}} < \omega_{n+1,\mathbf{k}}$.

The resulting data set contains as input pixelized permittivity profiles (in either 32 × 32 or 64 × 64 resolution) and as output the computed band structure (with the BZ sampled on a 23 × 23 Γ-centered Monkhorst–Pack grid, as in Figure 1A). In addition, we computed the band gap $\Delta\omega_{12} \equiv \min\, \omega_{2\mathbf{k}} - \max\, \omega_{1\mathbf{k}}$ between bands 1 and 2. Since the generated unit cells predominately feature central inclusions with a relative area less than 50% (Figure 1C), TM band gaps are significantly more abundant than TE band gaps (Figure. 1D). In our experiments with generative models, we restricted the data set to those elements that host a substantial band gap, defined heuristically as a relative band gap $\Delta\omega_{12}/\bar{\omega}_{12}$ greater than 5% (with mid-gap frequency $\bar{\omega}_{12} \equiv \frac{1}{2}\min\, \omega_{2\mathbf{k}} + \frac{1}{2}\max\, \omega_{1\mathbf{k}}$). Since the TE band structures host only very few such examples (48 with a non-zero band gap and 10 with a band gap ≥ 5%, out of 20,000 examples), we confined our experiments with generative models to the TM polarization only.

2.2 Band prediction

A natural question is whether neural networks can be used in lieu of traditional theory-driven tools for the modeling of PhCs, e.g. to predict a PhC’s band structure. To answer this, we adopted a supervised learning approach and trained two neural networks to reproduce the TM and TE band structures, respectively, taking as input a 32 × 32-discretized unit cell and producing as output the band structure across the 23 × 23-discretized BZ for the first six bands (Figure 2). Effectively, this is a regression problem where a large input space (32 × 32 = 1024 parameters) is mapped to a large output space (23 × 23 × 6 = 3174 parameters).

The network consists of two main components: encoder and decoder (Figure 2A). Conceptually, the encoder is tasked with building an abstract representation of the PhC’s unit cell $\varepsilon(\mathbf{r})$ that spans a lower-dimensional so-called feature (or latent) space. The decoder, conversely, is tasked with reconstructing from this feature vector the band structure of the input PhC. In practice, we

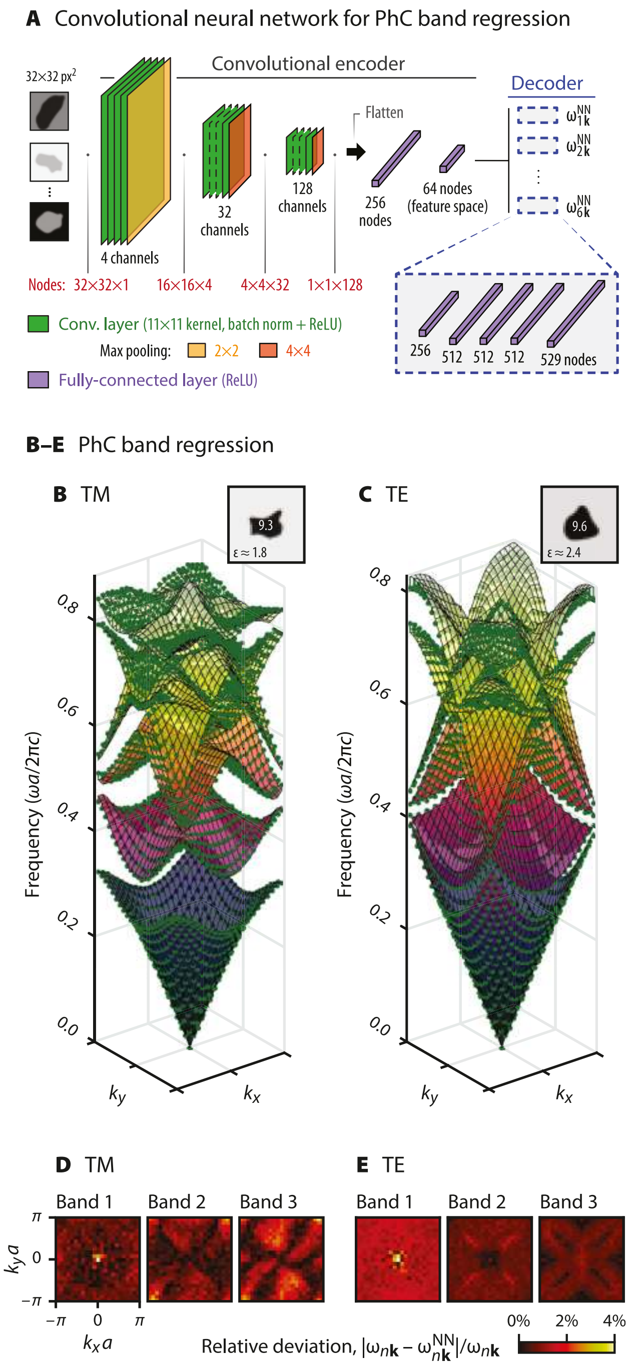


Figure 2: Band prediction with convolutional neural networks. (A) Network architecture showing the convolutional encoder and fully-connected decoder (described in detail in the main text). Numbers in red indicate the data size after every network layer. (B–C) Example applications of the trained band-prediction network on test set unit cells in both TM and TE polarizations (green markers, network predictions; surfaces, reference MPB calculations). The chosen unit cells represent worst-case examples due to their large permittivity contrast. (D–E) The relative deviation between network predictions and reference calculations. The relative error is typically very small, on the order of ≲ 2%.

implement and train the network using the popular PyTorch framework [49]. Training is accomplished by minimizing the mean square error between the training data $\omega_{n\mathbf{k}}$ and network output $\omega_{n\mathbf{k}}^{NN}$ across n, $\mathbf{k}$, and the entire training set (the cost function) using adaptive gradient descent optimization (RMSprop [50]) with an adaptive learning rate scheduler. We implement the encoder using three convolutional layers, each of (zero-padded) 11 × 11 kernels, followed by two fully-connected layers, essentially mapping the 32 × 32 input space into a linear 64-dimensional feature space. The convolutional layers were subjected to max-pooling and increasing channel depths to collapse the 2D input into a simple 1D vector that could be directly fed to the fully-connected layers of the encoder. The decoder was implemented with six feed-forward networks, each consisting of five fully-connected layers that were separately optimized for each band. All layers were followed by ReLU activations and batch normalization [51] was used for the convolutional layers. Our implementation (with optimized hyper-parameters) is available online, see Ref. 52, and summarized in Figure 2A.

We followed the standard training–validation–test approach and split the data set into training, validation, and test sets (in 70, 15, and 15% proportions). The training set was used to update the network's weights, the validation set to evaluate training convergence and select hyper-parameters, and the test set to determine the network's ability to generalize to new data (i.e. assess eventual network performance). We performed a simple grid-search to determine hyper-parameters, searching across kernel sizes of convolution layers $\in [5, 7, 9, \mathbf{11}]$, batch sizes $\in [\mathbf{32}, 64, 128]$, initial learning rates $\in [10^{-5}, \mathbf{10^{-4}}, 10^{-3}]$, and total number of training epochs $\in [20, 30, \mathbf{40}]$ (optimal hyper-parameters indicated in boldface). In addition, we searched across several network architectures consisting of varying convolution layer channel depths to arrive at the optimal configuration shown in Figure 2A. Application of the optimally tuned network on two examples from the test set is shown in Figure 2B–E, in absolute (Figure 2B–C) and relative scales (Figure 2D–E). Both examples are characterized by a large permittivity contrast between inclusion and background and consequently reflect extremal elements in the data set, whose band structures deviate substantially from the trivial empty-lattice approximation. Averaged across the entire validation and test sets, both the band-specific and the band-averaged relative mean errors $\text{mean}_{\mathbf{k}}(|\omega_{n\mathbf{k}}^{NN} - \omega_{n\mathbf{k}}|/\omega_{n\mathbf{k}})$ are generally very low, on the order of 0.5%, as shown in Table 1. We conclude that a simple convolution neural network can predict the band

Table 1: Neural network performance. Mean relative error, $\text{mean}_{\mathbf{k}}|\omega_{n\mathbf{k}}^{\text{NN}} - \omega_{n\mathbf{k}}|/\omega_{n\mathbf{k}}$, of the trained TE and TM networks on validation and test samples, shown for each band separately as well as band-averaged ($\overline{1-6}$).

Sample	Polarization	Band index *n* (‰ error)						
		1	2	3	4	5	6	$\overline{1-6}$
Validation	TM	4.8	5.3	6.3	6.3	6.5	6.7	**6.0**
	TE	6.0	4.8	4.9	5.3	5.4	5.6	**5.3**
Test	TM	4.7	5.2	6.1	6.1	6.4	6.5	**5.8**
	TE	6.0	4.9	4.8	5.3	5.4	5.8	**5.4**

structures of PhCs with very high accuracy and generalizes excellently to examples not seen during training. While we have confined our attention to 2D square lattices, this conclusion appears likely to apply generally across different lattice types and dimensionalities.

It is worth noting that while generation of a suitable data set—and, to a lesser extent, network training (taking ~3 min for fixed hyper-parameters on an Nvidia 1080 Ti GPU)—requires substantial computing resources, once trained, a neural network can predict band structures orders of magnitude faster than conventional theory-driven simulations (network evaluation of a single input takes ≈ 0.02 s on an Nvidia 1080 Ti GPU). While these gains are not sufficiently attractive to merit the training of regression networks for one- or few-off calculations, they can be relevant in inverse-design problems [26, 27] or high-throughput searches [53], where a very large number of distinct system configurations must be considered.

2.3 Generative adversarial networks

While ML techniques for classification and regression problems (such as band structure prediction) are naturally complementary to traditional theory-based approaches to forward problems, the field of generative modeling stands to complement conventional techniques of optimization and inverse design. Rather than learning a mapping from input to output data (e.g. from the unit cell to band structure), generative models generally seek to learn the statistical distribution of data samples. Once learned, many new elements can then be drawn from this distribution—a highly attractive option for optimization problems characterized by a non-unique solution space (in sharp contrast to conventional gradient-based approaches where the retrieval of diverse design candidates can be nontrivial).

Generative adversarial networks (GANs) have become a singularly prominent direction in generative models [54], due to their ability to seemingly generalize "creatively" beyond training data, with applications spanning e.g. autonomous driving systems [55], natural image synthesis [56], and anomaly detection [57]. The training of GANs mimics an adversarial game between two networks (Figure 3): while one network, the discriminator, is tasked with deciding whether a given input belongs to the training data ("real") or not ("fake"); the other, the generator, is tasked with producing (from an input vector sampled from a predefined probabilistic feature space) candidates that fool the discriminator. During training, their joint cost function—whose contributions are adversarial in nature, i.e. generally opposing—is optimized.

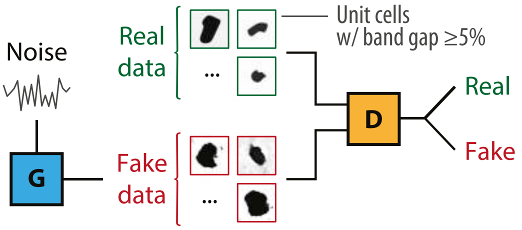


Figure 3: Generative adversarial network. Through an adversarial game between a generative (G) and a discriminative (D) network, new synthetic examples (fake) of 2D unit cells with a TM band gap can be generated from a genuine data set (real).

We explored the use of GANs for synthesizing new candidate unit cells that host a substantial TM band gap. To do so, we extracted the 585 unit cells with $\Delta\omega_{12}/\overline{\omega}_{12} \geq 5\%$ from the data set for use as training data. We tested three different GAN-variants [58]: a conventional GAN [53], a least squares GAN (LSGAN) [59], and Deep Regret Analytic GAN (DRAGAN) [60], each distinguished essentially by their respective generator and discriminator cost functions [61]. In each case, we adapted standard off-the-shelf implementations [62] to take a single-channel, 64 × 64 pixelized $\varepsilon(\mathbf{r})$ profile as training data. Training across 400 epochs took on the order of 5–10 min for each GAN on an Nvidia 1080 Ti GPU.

Figure 4A illustrates the improvement during the training of each GAN-variant's ability to generate convincing unit cells that exhibit the desired characteristics (i.e. well-defined, high-contrast, two-tone inclusions). We also evaluated the models' performance relative to the design goal of exhibiting a substantial band gap by computing the band gap sizes of the generated unit cells with MPB (Figure 4B). Concretely, we trained 10 distinct networks for each GAN-variant (distinguished only by network initialization), outputting at each epoch 16 generated unit cells. From these samples, we evaluated a notion of "generation fidelity", defined as the relative fraction of generated unit cells that

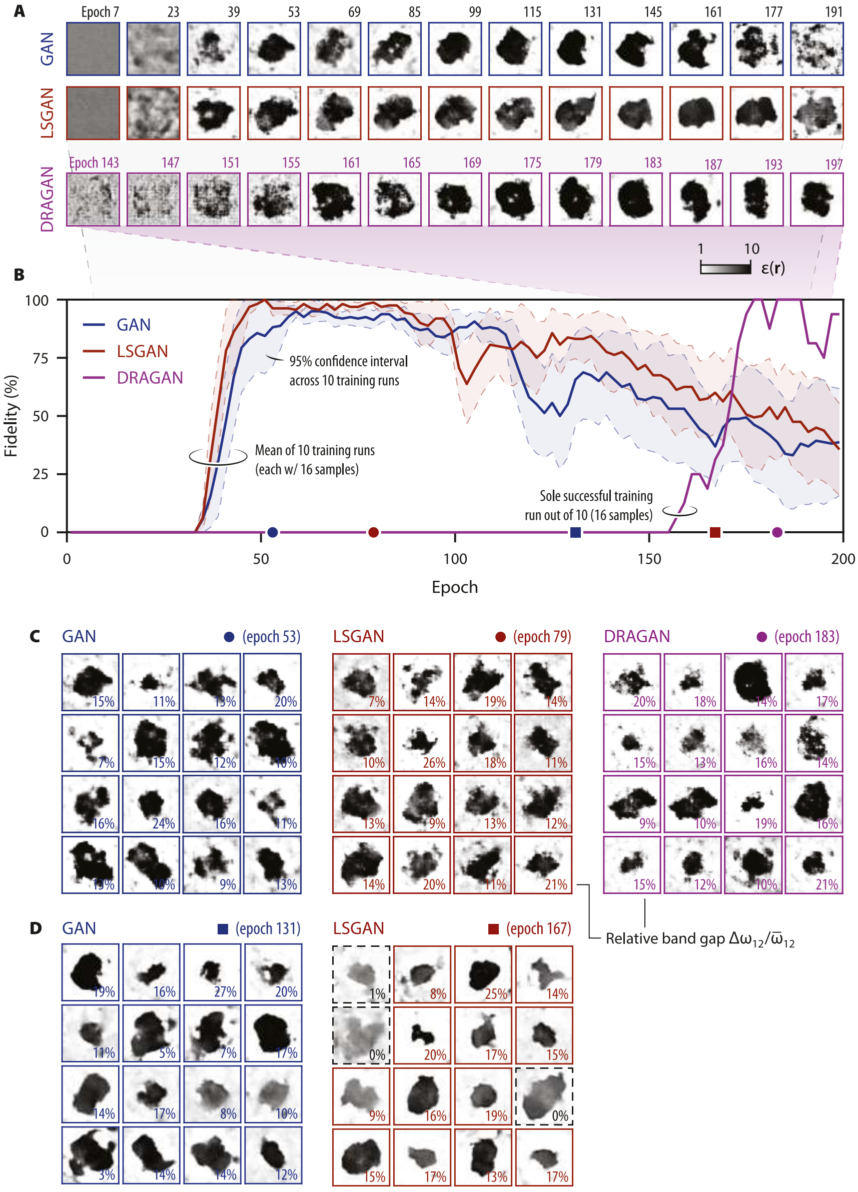


Figure 4: GAN, LSGAN, and DRAGAN for generation of unit cells with substantial TM band gaps. (A) The mapping of fixed feature vectors to generated unit cells during training. Note the differing epoch steps and ranges for DRAGAN versus GAN and LSGAN. (B) Fidelity of generated unit cells (the fraction hosting a band gap ≥5%). For GAN and LSGAN, fidelity is averaged over 16 distinct feature vectors and 10 training runs (uncertainty across training runs is indicated by shaded regions). Only a single DRAGAN training run was successful (averaged over 16 outputs). (C–D) Examples of generated unit cells at selected epochs (indicated by matching markers in B). GAN and LSGAN produce more well-defined but lower-fidelity unit cells at later epochs (text-insets give $\Delta\omega_{12}/\bar{\omega}_{12}$ evaluated with MPB; dashed borders highlight cases where $\Delta\omega_{12}/\bar{\omega}_{12} < 5\%$).

indeed exhibit a band gap ≥5%. Both metrics—visual “quality” and fidelity—exhibit much the same evolution: initially, performance is poor, reflecting essentially randomly initialized networks; then, within a few epochs, performance improves dramatically; and finally, performance slowly deteriorates, typical of the saturation problem [63]. While GAN and LSGAN achieve convincing performance within ~50 epochs, DRAGAN takes significantly longer, apparently passing through a phase of “fractured” inclusions. Further, across our 10 training experiments, we identified only a single successful DRAGAN trial (others not shown).

Figure 4C shows 16 examples of generated unit cells for each GAN-variant, evaluated at epochs and training runs of 100% fidelity. The generative models have clearly “learned” the key elements necessary to host a TM band gap, namely an inclusion of high permittivity embedded in a low-permittivity background [43]. Interestingly, although the fidelity of GAN and LSGAN generally decreases after peaks around range 5070 epochs, the visual quality—especially the well-definedness of inclusion boundaries—improves at higher epochs as shown in Figure 4D. The apparent cost of moving to higher epochs appear to be an increase in low-contrast examples without (or with smaller) band gaps. More generally, both visual quality and fidelity alike could likely be improved by simply enlarging the training set’s size. Finally, we note that regularization and filtering techniques from topology optimization [45, 46] could be leveraged to further reduce noise or ensure minimum feature sizes in the generated designs, either as a post-processing step or during training.

2.4 Image-to-image translation

Image-to-image translation can be viewed as a subset of generative modeling concerned with translating (i.e. mapping) between distinct representations of images. Effectively, this translation can often be viewed simply as implanting the "style" or characteristics of a given representation A onto another B; say, mapping from an outline, or even a sketch, to a photorealistic representation (e.g. of cats [64]). Following the introduction of the `pix2pix` software [65], conditional GANs [66] have emerged as a powerful tool to achieve this translation. The underlying principle is illustrated in Figure 5A: the generator of a conditional GAN takes, in addition to the standard random feature vector x, a "conditional input" y (of representation A) from which a fake output $G(x,y)$ is generated—the discriminator, conversely, seeks to distinguish between genuine pairings of y and real output z (of representation B) from faked pairings.

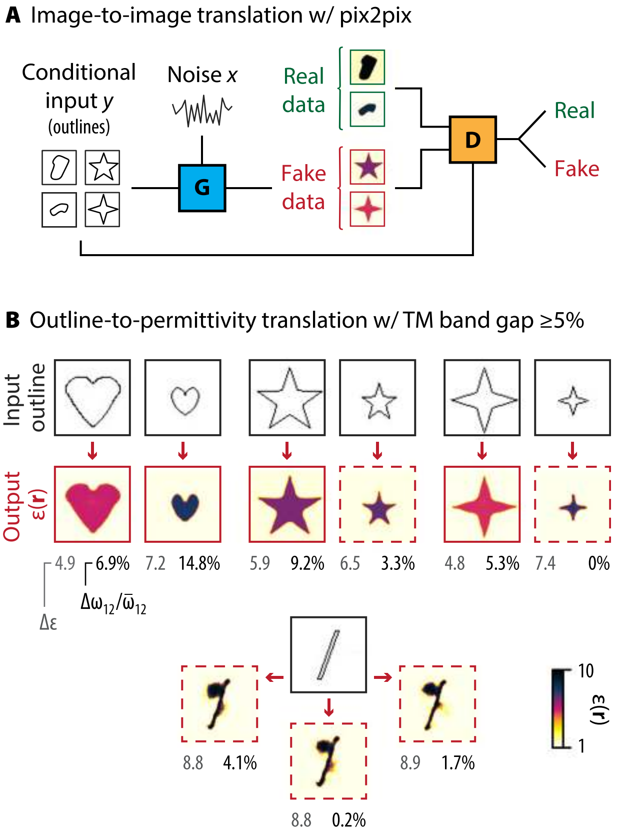


Figure 5: Image-to-image translation of photonic features. (A) Conditional GANs, as implemented e.g. by `pix2pix` [65], facilitate image-to-image translation by augmenting a conventional GAN (Figure 3) with a conditional input. (B) Using `pix2pix`, we trained a model to translate a discretized inclusion outline (black borders) to a permittivity profile (red borders) hosting a TM band gap. The permittivity contrast $\Delta\varepsilon \equiv \max \epsilon(\mathbf{r}) - \min \epsilon(\mathbf{r})$ and the relative band gap $\Delta\omega_{12}/\overline{\omega}_{12}$, evaluated with MPB, are indicated below each design (dashed borders highlight cases where $\Delta\omega_{12}/\overline{\omega}_{12} < 5\%$).

A natural application of image-to-image translation, and `pix2pix` specifically, for photonics is "guided inverse design", i.e. inverse design subject to conditional input. Figure 5B illustrates one such application (using a PyTorch implementation of `pix2pix` [65, 67, 68]): by taking again the set of unit cells with a TM band gap ≥5% and choosing as conditional input the corresponding inclusion outlines, we can learn a mapping from outlines to permittivity profiles supporting a TM band gap. We trained the model using just 256 samples (each of 64×64 pixels) over 200 epochs (requiring less than 1 h on an Nvidia 1080 Ti GPU). We tested the trained model on conditional input of several distinct shapes (heart and five- and four-pointed stars) and scales. The trained model successfully translates each large inclusion to a permittivity profile with a TM band gap ≥5%. Notably, this translation is successful—and maintains the outline's shape—even though the training data does not contain examples that resemble the chosen outlines. Further, when the scale of a shape is reduced, we observe that the contrast in the generated profile is increased; in exact agreement with the basic design-principle suggested by perturbation theory [43]. While the small five- and four-pointed stars translations do not achieve a TM band gap ≥5%, it is clear that the design approach (i.e. increasing contrast) is valid. Indeed, for sufficiently small or irregular inclusions, designs with $\varepsilon(\mathbf{r}) \in [1, 10]$ and a ≥5% band gap may not exist. We can explore this latter point by feeding the trained model a stick too narrow to host a TM band gap (Figure 5B, bottom). We sampled three generated designs (distinct feature vectors): in each case, the design "breaks out" of the outline and maximizes contrast. The resulting rupture varies slightly in extent and so hosts differently sized band gaps, though in each case ≤5%.

3 Conclusions

In conclusion, we have explored predictive and generative models for data-driven approaches to PhC analysis and design. Within predictive modeling, we demonstrated that convolutional neural networks can be trained to predict the band structure of a square 2D PhCs with high accuracy and with orders of magnitude speedup across both TE and TM polarizations. Within generative modeling, we demonstrated that standard techniques, namely GANs and conditional GANs, can be readily adapted for high-throughput unguided and guided inverse design; here, for the inverse design of PhCs with sizable TM band gaps. A key advantage of data-driven approaches to inverse design is that otherwise hard-to-quantify constraints, such as notions of fabricability, can be encoded implicitly by a representative selection of training data (here, smooth two-tone inclusions). Such data-driven approaches to inverse design could also make appealing alternatives to traditional inverse design tools in scenarios where a large number of design candidates are desired for a fixed design goal. Encouragingly, high-fidelity generative models could be trained even with relatively modest data quantities; here, just ~ 250 – 600 unit cells.

We note that the relative ease with which standard ML techniques can be adapted and applied to PhCs, as shown here, suggests a promising application space for data-driven approaches in photonics more generally. Especially within generative modeling, a large suite of ML techniques exists that point to several opportunities for data-driven inverse photonic design, some of which have already been explored: among them, variational autoencoders [69] exemplify a natural alternative [70] to GANs for photonic inverse design [71, 72], as does the related approach of bidirectional neural networks [73, 74]. Further, the ML application-space for PhCs extends beyond the periodic settings considered here: for instance, both isolated and aperiodic systems, such as PhC defect cavities and quasiperiodic PhCs, may be explored with similar ML techniques, e.g. by an appropriate augmentation of the input space. Even with this outlook, the appeal of data-driven computational photonics—and science more broadly—will remain closely correlated with the required quantities of data needed to train networks, and the ease with which it may be generated. Given the performance and maturity of state-of-the-art theory-driven methods for PhCs, we believe PhCs will make an ideal test bed to explore and develop new ML techniques, e.g. ideas from transfer- and meta-learning, for photonics and the natural sciences.

Acknowledgments: We thank Yichen Shen, Rumen Dangovski, Samuel Kim, and Peter Lu for fruitful discussions. Research supported in part by the Army Research Office through the Institute for Soldier Nanotechnologies under contract No. W911NF-18-2-0048, in part by the MIT–SenseTime Alliance on Artificial Intelligence, in part by the MRSEC Program of the National Science Foundation under award No. DMR–1419807, and in part upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00111890042. Research was sponsored in part by the United States Air Force Research Laboratory and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. T. C. was supported in part by the Danish Council for Independent Research (Grant No. DFF–6108-00667). C. L. acknowledges financial support from the DSO National Laboratories, Singapore. D. J. acknowledges the donation of GPU resources by the Nvidia Corporation.

References

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

[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, p. 436, 2015.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, *IEEE Conference on Computer Vision and Pattern Recognition*, 2016;770. https://doi.org/10.1109/CVPR.2016.90.

[5] R. Girshick, “Fast R-CNN”, in *IEEE International Conference on Computer Vision*, 2016, pp. 1440–1448. https://doi.org/10.1109/ICCV.2015.169.

[6] D. Shen, G. Wu, and H. Suk, “Deep learning in medical image analysis,” *Annu. Rev. Biomed. Eng.*, vol. 19, p. 221, 2017.

[7] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 6645–6649.

[8] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and tell: a neural image caption generator,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3156–3164.

[9] K. Cho, B. van Merrienboer, C. Gulcehre, et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *Conference on Empirical Methods in Natural Language Processing*, 2014.

[10] J. Devlin, M. Chang, K. Lee, and K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, arXiv:1810.04805, 2019.

[11] D. Silver, A. Huang, C. J Maddison, et al., “Mastering the game of Go with deep neural networks and tree search,” *Nature*, vol. 529, p. 484, 2016.

[12] V. Mnih, K. Kavukcuoglu, D. Silver, et al., “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, p. 529, 2015.

[13] L. Arsenault, A. Lopez-Bezanilla, O. A. von Lilienfeld, and A. J. Millis, “Machine learning for many-body physics: the case of the Anderson impurity model,” *Phys. Rev. B*, vol. 90, p. 155136, 2014.

[14] M. Schuld, I. Sinayskiy, and F. Petruccione, “An introduction to quantum machine learning,” *Contemp. Phys.*, vol. 56, p. 172, 2015.

[15] J. Carrasquilla and R. G. Melko, "Machine learning phases of matter," *Nat. Phys.*, vol. 13, p. 431, 2017.

[16] M. Raissi and G. E. Karniadakis, "Hidden physics models: machine learning of nonlinear partial differential equations," *J. Comput. Phys.*, vol. 357, p. 125, 2017.

[17] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," *Rep. Phys. Prog.*, vol. 81, p. 074001, 2018.

[18] G. Carleo, I. Cirac, K. Cranmer, et al., "Machine learning and the physical sciences," *Rev. Mod. Phys.*, vol. 91, p. 045002, 2019.

[19] G. Pilania, A. Mannodi-Kanakkithodi, B. P. Uberuaga, R. Ramprasad, J. E. Gubernatis, and T. Lookman, "Machine learning bandgaps of double perovskites," *Sci. Rep.*, vol. 6, p. 19375, 2016.

[20] Y. Zhuo, A. M. Tehrani, and J. Brgoch, "Predicting the band gaps of inorganic solids by machine learning," *J. Phys. Chem. Lett.*, vol. 9, p. 1668, 2018.

[21] G. Montavon, M. Rupp, V. Gobre, et al., "Machine learning of molecular electronic properties in chemical compound space," *New J. Phys.*, vol. 15, p. 095003, 2013.

[22] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques, "Recent advances and applications of machine learning in solid-state materials science," *Comput. Mater.*, vol. 5, p. 83, 2019.

[23] Y. Liu, T. Zhao, W. Ju, and S. Shi, "Materials discovery and design using machine learning," *J. Materiomics*, vol. 3, p. 159, 2017.

[24] S. Lu, Q. Zhou, Y. Ouyang, Y. Guo, Q. Li, and L. Wang, "Accelerated discovery of stable lead-free hybrid organic-inorganic perovskites via machine learning," *Nat. Commun.*, vol. 9, p. 3405, 2018.

[25] A. V. Lavrinenko, J. Lægsgaard, N. Gregersen, F. Schmidt, and T. Søndergaard, *Numerical Methods in Photonics*, CRC Press, 2015.

[26] J. Peurifoy, Y. Shen, L. Jing, et al., "Nanophotonic particle simulation and inverse design using artificial neural networks," *Sci. Adv.*, vol. 4, p. eaar4206, 2018.

[27] I. Malkiel, M. Mrejen, A. Nagler, U. Arieli, L. Wolf, and H. Suchowski, "Plasmonic nanostructure design and characterization via deep learning," *Light Sci. Appl.*, vol. 7, p. 60, 2018.

[28] P. Wiecha and O. L. Muskens, "Deep learning meets nanophotonics: a generalized accurate predictor for near fields and far fields of arbitrary 3D nanostructures," *Nano Lett.*, vol. 20, p. 329, 2019.

[29] L. Pilozzi, F. A. Farrelly, G. Marcucci, and C. Conti, "Machine learning inverse problem for topological photonics," *Commun. Phys.*, vol. 1, p. 57, 2018.

[30] B. Wu, K. Ding, C. T. Chan, and Y. Chen, *Machine Prediction of Topological Transitions in Photonic Crystals*, arXiv:1907.07996, 2019.

[31] Y. Long, J. Ren, Y. Li, and H. Chen, "Inverse design of photonic topological state via machine learning," *Appl. Phys. Lett.*, vol. 114, p. 181105, 2019.

[32] S. Inampudi and H. Mosallaei, "Neural network based design of metagratings," *Appl. Phys. Lett.*, vol. 112, p. 241102, 2018.

[33] S. An, C. Fowler, M. Y. Shalaginov, et al., "Modeling of all-dielectric metasurfaces using deep neural networks," in *International Applied Computational Electromagnetics Society Symposium*, 2019, pp. 1–2.

[34] C. C. Nadell, B. Huang, J. M. Malof, and W. J. Padilla, "Deep learning for accelerated all-dielectric metasurface design," *Opt. Express*, vol. 27, p. 27523, 2019.

[35] Z. A. Kudyshev, A. V. Kildishev, and V. M. Shalaev, and A. Boltasseeva, "Machine-learning-assisted metasurface design for high-efficiency thermal emitter optimization," *Appl. Phys. Rev.*, vol. 7, p. 021407, 2020.

[36] Z. Liu, D. Zhu, S. P. Rodrigues, K. Lee, and W. Cai, "A generative model for inverse design of metamaterials," *Nano Lett.*, vol. 10, p. 6570, 2018.

[37] J. Jiang, D. Sell, S. Hoyer, J. Hickey, J. Yang, and J. A. Fan, "Free-form diffractive metagrating design based on generative adversial networks," *ACS Nano*, vol. 13, p. 8872, 2019.

[38] J. Jiang and J. A. Fan, "Simulator-based training of generative neural networks for the inverse design of metasurfaces," *Nanophotonics* [ahead of print], 2019. https://doi.org/10.1515/nanoph-2019-0330.

[39] J. S. Jensen and O. Sigmund, "Topology optimization for nano-photonics," *Laser Photon. Rev.*, vol. 5, p. 308, 2011.

[40] S. Molesky, Z. Lin, A. Y. Piggott, W. Jin, J. Vucković, and A. W. Rodriguez, "Inverse design in nanophotonics," *Nat. Photonics*, vol. 12, p. 659, 2018.

[41] T. Back, U. Hammel, and H. P. Schwefel, "Evolutionary computation: comments on the history and current state," *IEEE Trans. Evol. Comput.*, vol. 1, p. 3, 1997.

[42] E. Kerrinckx, L. Bigot, D. Douay, and Y. Quiquempois, "Photonic crystal fiber design by means of a genetic algorithm," *Opt. Express*, vol. 12, p. 1990, 2004.

[43] J. D. Joannopoulos, S. G. Johnson, J. N. Winn, and R. D. Meade, *Photonic Crystals: Molding the Flow of Light*, 2nd ed. Princeton University Press, 2008.

[44] K. Sakoda, *Optical Properties of Photonic Crystals*, 2nd ed., Springer, 2004.

[45] F. Wang, J. S. Jensen, and O. Sigmund, "Robust topology optimization of photonic crystal waveguides with tailored dispersion properties," *J. Opt. Soc. Am. B*, vol. 28, p. 387, 2011.

[46] F. Wang, B. S. Lazarov, and O. Sigmund, "On projection methods, convergence and robust formulations in topology optimization," *Struct. Multidisc. Optim.*, vol. 43, p. 767, 2011.

[47] We note, however, that such additional constraints and regularization techniques could be leveraged to further guarantee and improve the fabricability of generative ML designs (with opportunities both in pre-selection of training data and in post-processing or "normalization" of generated designs).

[48] S. G. Johnson and J. D. Joannopoulos, "Block-iterative frequency-domain methods for Maxwell's equations in a planewave basis," *Opt. Express*, vol. 8, p. 173, 2001.

[49] A. Paszke, S. Gross, F. Massa, et al., "PyTorch: an imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, vol. 32, pp. 8024–8035, 2019.

[50] T. Tieleman and G. Hinton, *Lecture 6.5–rmsprop*, 2012.

[51] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37, pp. 448–456, 2015.

[52] https://github.com/clott3/PhC-2D-sq.

[53] N. Claussen, B.A. Bernevig, and N. Regnault, *Detection of Topological Materials with Machine Learning*, arXiv:1910.10161, 2019.

[54] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., “Generative adversarial nets,” in *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

[55] M. Zhang, Y. Zhang, L. Zhang, C. Liu, and S. Khurshid, “DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems,” in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 132–142, 2018.

[56] A. Brock, J. Donahue, and K. Simonyan, *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, arXiv:1809.11096, 2018.

[57] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, *Efficient GAN-based Anomaly Detection*, arXiv:1802.06222, 2018.

[58] We also tested Wasserstein GAN (WGAN) [75] but did not achieve good results.

[59] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, “Least squares generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2794–2802, 2017.

[60] N. Kodali, J. Abernethy, J. Hays, and Z. Kira, *On Convergence and Stability of GANs*, arXiv:1705.07215 (2017).

[61] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, “Are GANs created equal? a large-scale study,” *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 698–707, 2018.

[62] H. Kang, https://github.com/znxlwm/pytorch-generative-model-collections, 2017.

[63] M. Arjovsky and L. Bottou, “Towards principled methods for training generative adversarial networks,” in *5th International Conference on Learning Representations*, 2017.

[64] C. Hesse, https://affinelayer.com/pixsrv/, 2017.

[65] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5967–5976, 2017.

[66] M. Mirza and S. Osindero, *Conditional Generative Adversarial Nets*, arXiv:1411.1784, 2014.

[67] J.-Y. Zhu, T. Park, and T. Wang, “Image-to-image translation in PyTorch,” https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix, 2017.

[68] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017.

[69] D. P. Kingma and M. Welling, *Auto-encoding Variational Bayes*, arXiv:1312.6114, 2013.

[70] Z. Hu, Z. Yang, R. Salakhutdinov, and E. Xing, *On unifying deep generative models*, arXiv:1706.00550, 2018.

[71] W. Ma, F. Cheng, Y. Xu, Q. Wen, and Y. Liu, “Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy,” *Adv. Mater.*, vol. 31, p. 1901111, 2019.

[72] Z. Liu, L. Raju, D. Zhu, and W. Cai, “A hybrid strategy for the discovery and design of photonic nanostructures,” *IEEE J. Emerg. Sel. Top. C [early access]*, 2019.

[73] W. Ma, F. Cheng, and Y. Liu, “Deep-learning-enabled on-demand design of chiral metamaterials,” *ACS Nano*, vol. 12, p. 6326, 2018.

[74] D. Liu, Y. Tan, E. Khoram, and Z. Yu, “Training deep neural networks for the inverse design of nanophotonic structures,” *ACS Photonics*, vol. 5, p. 1365, 2018.

[75] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein Generative Adversarial Networks,” in *Proceedings of Machine Learning Research*, vol. 70, pp. 214–223, 2017.