Solving Out-of-Distribution Challenges in Optical Foundation Models using Self-Improving Data Augmentation

Mingqian Ma
Department of EECS
University of Michigan, Ann Arbor
mamq@umich.edu

Taigao Ma[*] [†]
Department of Physics, University of Michigan
Ann Arbor, MI
taigaom@umich.edu

L. Jay Guo[*]
Department of EECS
University of Michigan, Ann Arbor
guo@umich.edu

Abstract

Optical multilayer thin film structures are widely used in many photonic applications, including filters, absorbers, photovoltaics, display devices. The important part to enable these applications is the inverse design, which seeks to identify a suitable structure that satisfy desired optical responses. Recently, a Foundation model-based OptoGPT is proposed and has shown great potential to solve a wide range of inverse design problems. However, OptoGPT fails to design certain types of optical responses that are important to practical applications. The major reason is that the training data is randomly sampled and it is highly probable that these design targets are not selected in training, leading to the out-of-distribution issue. In this work, we propose a self-improving data augmentation technique by leveraging neural networks' extrapolation ability. Using this method, we show significant improvement in various application design tasks with minimum fine-tuning. The approach can be potentially generalized to other inverse scientific foundation models.

1 Introduction

Optical multilayer thin film structures (shortened as multilayer structures) are stacked thin film layers (typical thickness range from a few nm to several hundred nanometers) with different materials at each layer. Because of their ease of fabrication, they have been widely used for a variety of industrial applications including absorber [1], optical filters [2], structural color [3, 4], and distributed Bragg reflectors (DBR) [5, 6]. The different optical properties of the material (e.g., refractive index) and different thickness at each layer will combine together to determine the overall optical behavior of the stack, enabling precise control over light transmission, reflection, and absorption. The forward process of obtaining such responses for a specific structure can be easily calculated through electromagnetic simulation tools such as Transfer Matrix Methods (TMM) [7]. However, the inverse process is nontrivial, where one needs to identify a suitable structure to satisfy desired optical responses. For a long time, iterative optimization-based methods have been widely used

[*] Correspondence to: taigaom@umich.edu and guo@umich.edu.

[†] Work done while at University of Michigan, currently at Visa Research, Austin, TX 78759, USA.

Foundation Models for Science Workshop,38th Conference on Neural Information Processing Systems (NeurIPS 2024).

to design these multilayer structures, including Particle Swarm Optimization (PSO) [8, 9], Needle Optimization [10] and Genetic Algorithm [11, 12]. However, these methods are often time-consuming and task-specific, making them unsuitable for complex and practical design scenarios.

In recent years, advances in deep learning techniques have made it possible to inverse design diverse multilayer structures after training a neural network. Especially, with the development of a new foundation model based OptoGPT [13], researchers are able to design any type of spectrum and obtain multiple different structures within 0.1 second, which significantly shortens the design time while still achieving improved design accuracy. In OptoGPT, the authors reformulate the multilayer structure design as a sequence generation task conditioned on an input optical spectrum, where each generated token denotes both material and thickness information at each layer, and use a decoder-only transformer architecture for the structure generation. In order to train OptoGPT, they generate a large training dataset comprising 10M pairs of multilayer structures and simulated optical responses, i.e., the reflection and transmission spectrum. During dataset generation, they first randomly sample 10M structures from a large potential design space with 10^{59} different structures, then use the TMM to simulate and obtain corresponding optical responses.

However, random sampling inside the large space of multilayer structures does not guarantee a similar distribution inside the space of optical responses. This is because the forward mapping from multilayer structures to optical responses is non-uniform (originated from solving the non-linear equations in TMM). Therefore, it is possible that the randomly sampled structures only correspond to a limited region in the space of optical responses, e.g., the green region in Figure 1. In the cases where we want to design optical responses in the red region, the model would find it difficult to deal with because of insufficient training data inside this region. For example, the DBR-type high reflector [5, 6] requires an alternating high/low refractive index profile with thickness at each layer equal to a quarter wavelength. This leads to a special structure that can hardly be generated through random sampling. We refer to this situation as the **out-of-distribution (OOD)** problem, which is a common issue in many deep-learning-based inverse design algorithms because of the random sampling strategy in structure space. Notice we cannot randomly sample in the space of optical responses because that's the target we hope to design.

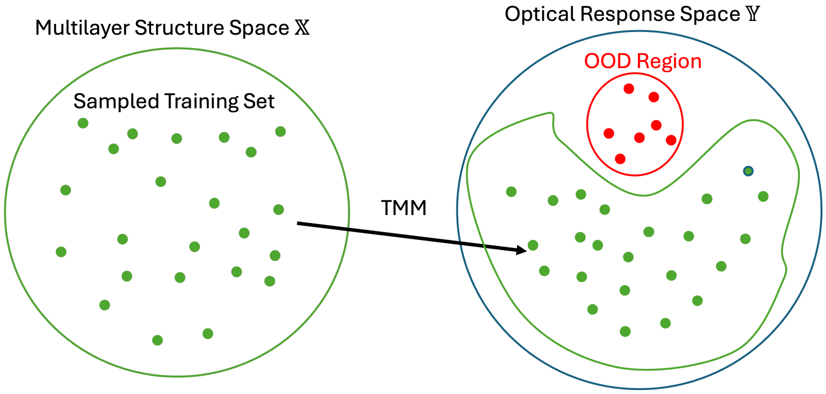
Multilayer Structure Space $\mathbb{X}$
Sampled Training Set
TMM
Optical Response Space $\mathbb{Y}$
OOD Region

Figure 1: Out-of-distribution (OOD) Problem: a random sampling strategy inside the structure space may correspond to a subspace inside the responses space. When designing for a new spectrum out of this region, e.g., the red region, the model will experience OOD problem. TMM: Transfer Matrix Methods.

In this work, we propose a self-improving data augmentation method to solve the OOD problem in inverse design by continuously querying the OptoGPT model and gradually obtaining improved design performance towards the OOD region through a series of perturbation techniques. Using a minimal amount of augmented data, we finetune our model and demonstrate improved performance on a wide range of OOD applications. Our major contributions can be summarized as follows:

- We propose a workflow of self-improving data augmentation to iteratively finetune foundation models to explore the out-of-distribution problem. The data augmentation involves obtaining initial data by leveraging the extrapolation ability of neural networks, then combining genetic algorithm and particle swarm optimization method to actively perturb the generated data towards the OOD region.

- We define an evaluation metric to balance the design accuracy as well as the distance away from the original dataset region in the corresponding spectrum space. Using this metric, we are able to visualize the learning effect of our model at different augmentation steps.
- We demonstrate the proposed methods in the OptoGPT model and achieve much better performance in many difficult but practical photonic applications.

2 Related Work

2.1 Multilayer Thin Film Inverse Design

Traditional inverse design methods for multilayer structures are usually based on heuristic optimizations. Recently, researchers have started to use deep learning methods as an alternation, including tandem network [14, 15], Variational Auto-Encoders (VAE) [16], Generative Adversarial Network [17] and Generative Pretrained Transformers (GPT) [13]. After training on a large dataset, these models can learn the inverse mapping from optical responses to structure space and finish designs within one second, which is much faster than optimization methods. Notably, the complexity of these deep learning models requires large amounts of training data, which are usually synthetic data prepared through simulation methods like the Transfer Matrix Method [7]. Most of these methods adopt a similar random sampling policy and suffer from OOD issues. However, how to deal with the OOD problem has not been extensively explored in this domain.

2.2 Out-of-Distribution Problem

Out-of-Distribution (OOD) problem in many traditional deep learning tasks refers to the phenomenon that the training set has a significant distribution shift with the test data, leading to poor generalization performance [18, 19]. A promising solution method to the problem is Transfer Learning, which has been investigated in both CV and NLP fields [20, 21, 22, 23]. The most related work in the field of nanophotonics problems is Ref. [24, 25], which solves the OOD problems in forward simulation by generating new training data iteratively [24] to enable the model to predict two-dimension photonic attributes (material stiffness and strength level). However, our problem considered here is much more complicated and difficult since we are considering the inverse problem with high dimensional out-of-distribution data.

3 Methods

3.1 Problem Definition

In the scientific domain, considering the causal factors $x \in \mathbb{X}$ and the observed physical outcome $y \in \mathbb{Y}$, there are two major problems to be solved: the forward simulation of $\mathbb{X} \to \mathbb{Y}$, and the inverse design of $\mathbb{Y} \to \mathbb{X}$. The forward simulation is definite and can be obtained by solving the mathematical equations involved inside the physical system. However, the inverse design is nontrivial because there is no such equation that describes the inverse relationship. In addition, the one-to-many mapping issue, meaning multiple different x can lead to similar y, also complicates this problem.

For multilayer structures, the causal factors x refer to the multilayer structure and include the material composition $m = [m_1, m_2, \ldots, m_N]$ and layer thicknesses $t = [t_1, t_2, \ldots, t_N]$, where N denotes the number of layers. Usually, material m_i is a discrete variable, and t_i is a continuous variable. On the other hand, y refers to the optical responses that we hope to design, e.g., spectrum or colors, and can be represented as a vector. TMM [7] can finish the forward prediction and calculate the spectrum y for a given x by solving equations relating to the wave fields at the boundaries of each layer. However, TMM does not inherently support the reverse computation of deriving x from a given y. In OptoGPT, the model treats the design parameters of $x = \{[m_1, m_2, \ldots, m_N], [t_1, t_2, \ldots, t_N]\}$ as a sequence of $x = \{[m_1, t_1], [m_2, t_2], \ldots, [m_N, t_N]\}$ and uses sequential generation to design multilayer structures layer-by-layer when given input y as a condition. In their consideration, at each layer, there are 18 different types of materials and 50 different thicknesses discretized by 10nm ranging from 10nm to 500nm. The maximum number of layers is 20. Therefore, the size of the multilayer structures design space extends to $(18 * 50)^{20} \sim 10^{59}$. For optical responses of y, they are considering reflection and transmission spectrum among 400-1100nm with 10nm gap.

The dataset generation in OptoGPT is a two-step random sampling process. They first randomly generate 10M different structures by uniformly sampling the material and thickness at each layer, then use TMM to simulate and obtain their spectrum, making the dataset consist of (x, y) pairs of structures and spectrum. This synthetic dataset is then used to train OptoGPT to learn the inverse mapping from desired spectral outcomes back to structural designs. As mentioned above, this would lead to the issue of OOD and causes problems when designing for some practical applications. This issue is non-trivial as we cannot add infinite data in the sampling process due to computational limit, and many traditional methods in CV and NLP such as in-context learning [26] cannot be applied as we have no plausible assumptions on the distribution of our sampled region with the actual distribution of the space.

3.2 Self-Improving Data Augmentation

Neural networks have shown some limited extrapolation ability [27]. This, however, can be utilized to provide a starting point when augmenting training data. Here is the intuition: we can gradually explore and extend the data space out of the original scope through perturbations guided by physical simulators and finally reach the OOD region. We illustrate this idea in Figure 2a. The green region refers to the space of y corresponding to the random sampling in the space of x. The red region is the OOD region that is not sampled or under-sampled in the space of y. Notice that each point denotes a different optical response, namely, a different inverse design task. After training on the green region, we expect that the neural networks have some extrapolation ability to expand the identifiable region to the blue region, which is the region that is slightly outside of the training dataset distribution.

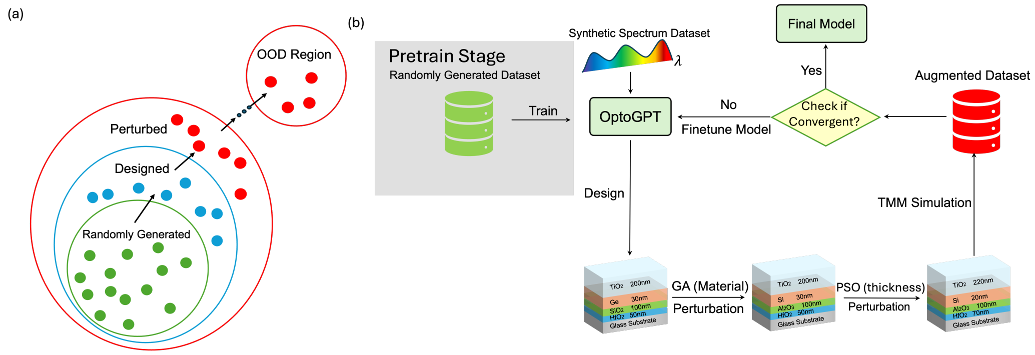

Figure 2: Illustration of the self-improving data augmentation. (a) Visualization of the self-improving in the space of optical responses y. (b) Diagram of our workflow: We input a synthetic OOD dataset and ask the model to generate the structure design. Then using perturbation method, we further explore beyond the scope of the original model, and construct a small augmented dataset with better data representation of the OOD region to finetune the model.

To reach the blue region, we query our model with some practical and difficult design targets that are outside of the training dataset and ask our model to give designs, which will be used as a starting point to approach the red region. Next, we will perturb inside the structure space of x. To guarantee that we are approaching the red region in the response space of y, we use GA to perturb the materials and PSO to perturb the thickness with the goal of minimizing the difference in the optical responses. However, instead of running the complete optimization which requires thousands of iteration steps, we only run limited optimization steps (e.g., 5 iterations) to maintain small perturbation and fast speed. After that, we use TMM to obtain the simulated optical responses for these perturbated structures and calculate their spectrum MSE compared to the target spectrum. We only keep these structures with a lower spectrum MSE than the previous model and collect them to create a new dataset for finetuning. We can iterate this process several times to gradually push the model to approach the OOD region and achieve better performance. Figure 2b shows the complete pipeline of self-improving data augmentation. We also provide a pseudo algorithm in Algorithm 1.

Algorithm 1 Self-Improving Data Augmentation

```
1: Construct a dataset $S$ with optical responses in the OOD region.
2: Query the OptoGPT model with $S$ and generate a set of structures $X$ using auto-regressive
   decoding method $D$, e.g., greedy decoding, Top-KP sampling [28, 29], or beam search.
3: Apply perturbation method GA_PSO on $X$ to create multiple new samples $\hat{X}$.
4: Use TMM to evaluate the performance of $\hat{X}$ and select those with improved performance to form
   a new augmented dataset $X'$.
5: Finetune OptoGPT on this augmented dataset $X'$.
6: Repeat step 2-5 until no significant improvement on model performance.
```

3.3 Perturbation Method

The core of our method is the perturbation, which enables our model to effectively explore beyond the current scope and move to the desired OOD region. Therefore, the generated structures of X should be as diverse as possible, giving a wide range of distinct starting point so that we can better explore beyond the original scope. Fortunately, for GPT-type models, this can be easily achieved using probabilistic decoding methods such as top-k [28] or top-p sampling [29], where different sampling seeds would lead to different outputs.

Once we obtain initial structures from OptoGPT, the next step is to perturb these structures towards the OOD region. Multilayer structures have two different sets of parameters: the material and thickness. Therefore, we propose a two-step perturbation approach called **GA_PSO** to first perturb the material using GA, then perturb thickness using PSO. During each perturbation, the structures are optimized and modified with the goal of decreasing the difference between simulated optical responses and target responses. However, different from general optimization process which takes thousands or more iterations, we terminate the optimization process within few iterations of the workflow. This is because we only want to introduce a small perturbation to the training dataset for a better learning. In addition, running more iterations on each individual sample from this generated structures can take a long time. Early termination in the optimization process can provide a good balance between performance and efficiency. We also provide a pseudo algorithm for **GA_PSO** in Algorithm 2.

Algorithm 2 GA_PSO Perturbation Method

```
For each individual structure in the generated structure $X$, fix its thickness and apply the Genetic
Algorithm (GA) to perturb the materials with the following strategies:
    • Mutation: Divide materials into five different types based on the refractive index: high,
      medium, and low refractive index dielectric material, metal, and semiconductor materials.
      For each layer, perform material mutations with probabilities: 70% within the same group,
      20% switch to a different group, and 10% remain unchanged.
    • Crossover: Select two structures, and create a new structure by concatenating the first half
      from the first structure and the second half from the second structure.
For each individual structure in the generated structure $X$ after doing material perturbation, fix its
material and apply Particle Swarm Optimization (PSO) on the thickness. Since we only conduct
several iterations for PSO, we save all these intermediate structures among the optimization
trajectory for next step.
```

Notice here we perturb the material and thickness separately using different optimization methods. There are several considerations: (1) The space of the combination of material and thickness is huge, and the perturbation method needs to achieve progress in a controllable time. Since genetic algorithm is time-consuming, separating the thickness optimization for PSO will yield better efficiency. (2) Conducting material permutation requires integrating prior knowledge in material design, in which here we separate the materials in groups. This is only achievable by Genetic Algorithm in mutation phase since it can deal with discrete variables.

4 Experiment

In this section, we report experiments with both synthetic evaluation dataset and real-world design cases to offer insights into the effectiveness of our workflow. We highlight the following results:

- Our workflow significantly improves the model's design ability in multiple different tasks which experience out-of-distribution issues.
- We observe that our model is continuously learning and performs better with more self-augmenting iterations. Although the added dataset is much smaller compared to the training dataset, after finetuning, our model exhibits improved performance in each iteration measured by two metrics: the distance toward the original training dataset and the mean squared error with the target design dataset.
- In multiple practical photonic design tasks, our workflow surpasses original OptoGPT model and design structures with close-to-target spectrum.

4.1 Framework Performance

4.1.1 Synthetic OOD Dataset S

We construct our synthetic OOD dataset S using 1,000 spectrum from the following four types: (1) Gaussian spectrum; (2) Double Gaussian spectrum; (3) Sigmoid-shape Spectrum (4) High reflection spectrum from distributed Bragg reflector (DBR) structures [5]. We select these four kinds of spectrum for the following reasons: (1) The original OptoGPT model has very poor performance on these kinds of spectrum, i.e., they are in the out-of-distribution region. (2) Improving the performance on these synthetic spectrum designs benefits a series of real-word applications, including Fabry–Perot-cavity-based filters [30], DBR structures [5], band-pass filter [31], etc. This is because most of these applications are related to design spectrum with shapes similar to our synthetic dataset S.

Figure 3 provides several examples for these spectrum. In the dataset S, each spectrum differ from each other in terms of the peak position, center spectrum and standard deviation. The reflectance and transmittance spectrum are either set to be complementary (added to 100%), or 0 for one of them.

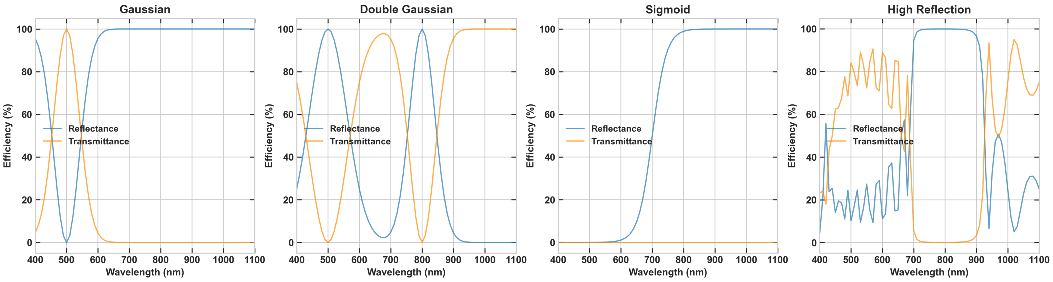

Figure 3: Examples of four different types of synthetic spectrum

4.1.2 Evaluation Metrics

There are two metrics we care about: the model's design accuracy, and the distance between augmented dataset and the training dataset. To evaluate the design accuracy, we first input the target spectrum y to our model and obtain designed structures $\hat{x}$, then we use TMM to calculate the simulated spectrum $\hat{y} = \text{TMM}(\hat{x})$. The design accuracy in a dataset is defined as the Mean Square Error (MSE) between $\hat{y}$ and y:

$$\text{MSE} = \frac{1}{N}\frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{M}||\hat{y}_{ij} - y_{ij}||^2$$

where N is the number of design targets, and M is dimension of spectrum.

To evaluate how far the augmented dataset is away from the original training dataset, for each augmented spectrum y_{aug_i}, we first compare the Euclidean distance between y_{aug_i} and all spectrum

y_{train} in the training dataset, and average the distance on five nearest spectrum. The overall distance of the augmented dataset is then averaged on all y_{aug}:

$$\text{Distance} = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{5}\sum_{\text{Five Nearest}} \text{Euclidean}(y_{\text{aug}_i}, y_{\text{train}})) \quad (1)$$

where N is the number of design targets. Considering the training dataset is large, identifying the five nearest spectrum inside the augmented dataset will take a long time. Therefore, we use FAISS[32] to help to speed up searching the nearest spectrum.

4.1.3 Model Performance

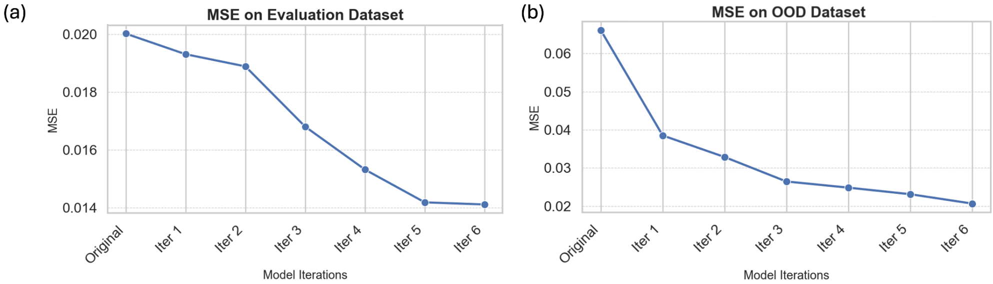

Figure 4: (a) MSE on the original evaluation dataset. (b) MSE on the Synthetic Dataset.

We first evaluate the model's design accuracy at each iteration step of data augmentation. We provide the details of OptoGPT model pretraining and our finetuning method in Appendix A. In total, we run for six iterations as we don't see a significant improvement after that. At each iteration, we evaluate the MSE on the original evaluation dataset and the synthetic OOD dataset, and visualize results in Figure 4. We can see that while the initial MSE on synthetic dataset is very high, the MSE drops quickly and continuously for several iterations using our data augmentation. This indicates that our method does significantly improve the design performance on OOD problems. In addition, the coherent decrease of MSE in the original evaluation dataset also suggests that our model does not sacrifice performance on the original dataset while improving performance on the OOD region.

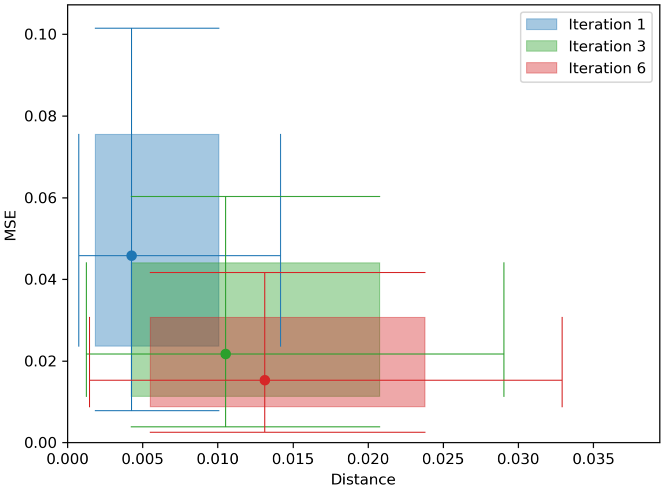

Figure 5: Visualization of the MSE (Design accuracy) v.s. the Distance of augmented dataset and training dataset at iteration 1, iteration 3 and iteration 6.

To have a better understanding of our augmented dataset, we further use a 2-d box plot to represent the distribution of augmented dataset on the two metrics of MSE and distance for iteration 1, iteration

3, and iteration 6 of the whole finetune framework. The results can be found in Figure 5. From the graph we can see the overall trend: with more iterations of the data augmentation, the model has lower MSE in spectrum design and the augmented dataset shows larger distance w.r.t. the original training dataset, i.e., the boxplot moves to the lower-right corner. This trend is coherent with the improvement of model's performance on the synthetic dataset.

4.2 Practical Photonic Applications

In addition to testing the performance of our methods on the synthetic dataset, we also dive into some practical inverse design tasks. As an example, we compare the design on four different spectrum: 1) High Reflection in Near-Infrared Region (NIR), 2) High Reflection in 600-900nm wavelength, 3) Band-notch filter in 550nm, 4) Dichroic filter which only reflects at 700nm but absorbs all other lights. These target spectrum have been widely used in many laser systems[33], spectrometer [34], imaging and sensing instruments [35], etc.

We compare the design performance of the original OptoGPT and our model with self-augmented dataset. In the generation stage, we use Top-KP sampling for both models with $k = 10, p = 0.8$. For both models, we sample for 100 times and report the structure that has the lowest MSE w.r.t. the target spectrum. We use TMM to obtain their simulated spectrum and visualize them in Figure 6. From the results, we can see that designed spectrum from our model are closer to the target and have less side reflections, demonstrating the effectiveness of our methods.

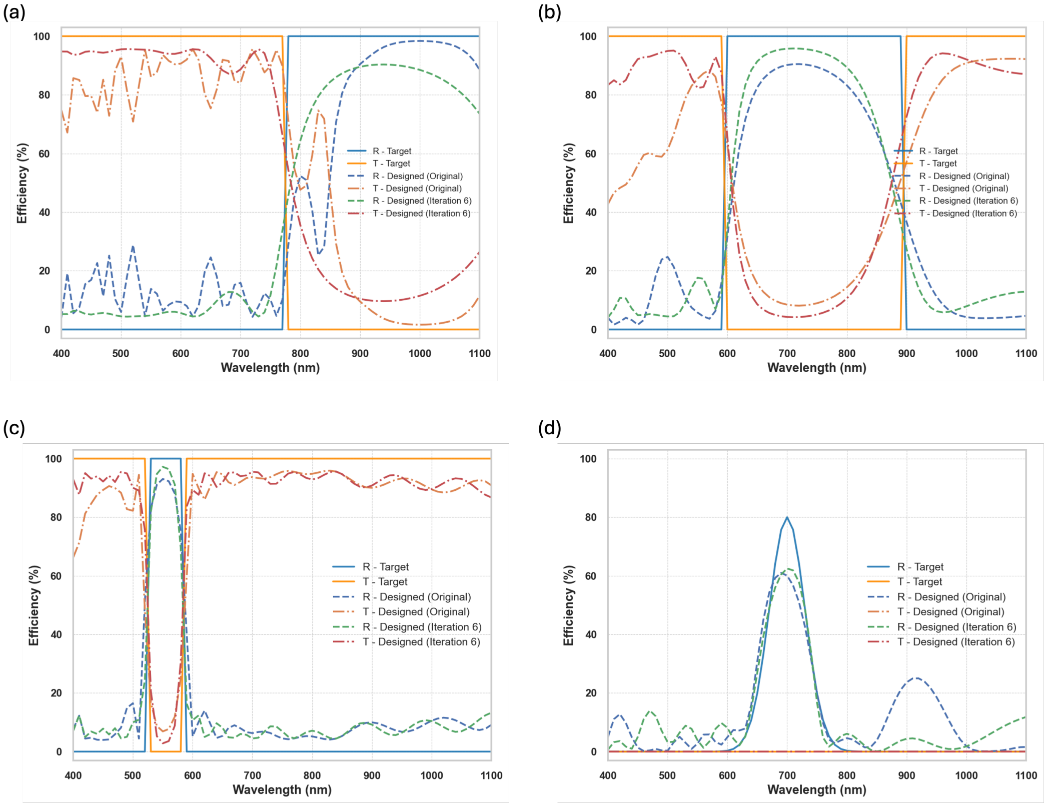

Figure 6: Examples for designing practical spectrum. Solid lines represent the design target and dash lines represent model designs. (a) Design of High Reflection spectrum in NIR region. (b) Design of High Reflection in 600-900nm wavelength. (c) Design of Band-notch filter in 550nm. (d) Dichroic filter which only reflects at 700nm but absorbs all other lights. In all these examples, we can see that our model performs better than the original model.

5 Conclusion

In this study, we introduce a novel self-improving data augmentation methods to effectively address the out-of-distribution challenges in multilayer thin film structures inverse design. By combining genetic algorithms and particle swarm optimization for structure perturbation as well as forward simulation to add augmented dataset for finetune, our method actively explores and expands the design space beyond the initial training set, thereby enhancing the model's performance and adaptability to out-of-distribution region and real-world scenarios. The success of this methodology underscores the potential of tackling out-of-distribution problem for general scientific inverse design problem, including designing metasurface and waveguide in photonic applications, and possibly in other fields like molecules or drug design.

References

[1] Weiwei Li et al. “Metamaterial absorbers: from tunable surface to structural transformation”. In: *Advanced Materials* 34.38 (2022), p. 2202509.

[2] Chengang Ji et al. “Decorative near-infrared transmission filters featuring high-efficiency and angular-insensitivity employing 1D photonic crystals”. In: *Nano Research* 12.3 (2019), pp. 543–548.

[3] Zhengmei Yang et al. “Enhancing the Purity of Reflective Structural Colors with Ultrathin Bilayer Media as Effective Ideal Absorbers”. In: *Advanced Optical Materials* 7.21 (2019), p. 1900739.

[4] Danyan Wang et al. “Structural color generation: from layered thin films to optical metasurfaces”. In: *Nanophotonics* 12.6 (2023), pp. 1019–1081.

[5] Jiaqi Hu et al. “Polariton laser in the Bardeen-Cooper-Schrieffer regime”. In: *Physical Review X* 11.1 (2021), p. 011018.

[6] Yoel Fink et al. “A dielectric omnidirectional reflector”. In: *Science* 282.5394 (1998), pp. 1679–1682.

[7] Steven J Byrnes. “Multilayer optical calculations”. In: *arXiv preprint* (2016).

[8] Rabi I. Rabady and Almahdi Ababneh. “Global optimal design of optical multilayer thin-film filters using particle swarm optimization”. In: *Optik* 125.1 (2014), pp. 548–553.

[9] Inho Lee et al. “Implementation of particle swarm optimization for complete inverse design of multilayered optical filters”. In: *Appl. Opt.* 62.34 (2023), pp. 8994–9001.

[10] Alexander V. Tikhonravov, Michael K. Trubetskov, and Gary W. DeBell. “Application of the needle optimization technique to the design of optical coatings”. In: *Appl. Opt.* 35.28 (1996), pp. 5493–5508.

[11] Martin F. Schubert et al. “Design of multilayer antireflection coatings made from co-sputtered and low-refractive-index materials by genetic algorithm”. In: *Opt. Express* 16.8 (2008), pp. 5290–5298.

[12] Yu Shi et al. “Optimization of Multilayer Optical Films with a memetic algorithm and mixed integer programming”. In: *ACS Photonics* 5.3 (2017), 684–691.

[13] Taigao Ma, Haozhu Wang, and L. Jay Guo. “OptoGPT: A foundation model for inverse design in optical multilayer thin film structures”. In: *Opto-Electron Adv* 7 (2024), p. 240062.

[14] Xiaopeng Xu et al. “An improved tandem neural network for the inverse design of nanophotonics devices”. In: *Optics Communications* 481 (2021), p. 126513.

[15] Xiaogen Yuan et al. “Multi-headed tandem neural network approach for non-uniqueness in inverse design of layered photonic structures”. In: *Optics & Laser Technology* 176 (2024), p. 110997.

[16] Mohammadreza Zandehshahvar et al. “Inverse design of photonic nanostructures using dimensionality reduction: reducing the computational complexity”. In: *Opt. Lett.* 46.11 (2021), pp. 2634–2637.

[17] Sunae So and Junsuk Rho. *Designing nanophotonic structures using conditional-deep convolutional generative adversarial networks*. 2019. arXiv: 1903.08432 [physics.optics].

[18] Jiashuo Liu et al. *Towards Out-Of-Distribution Generalization: A Survey*. 2023. arXiv: 2108.13624.

[19] Ali Geisa et al. *Towards a theory of out-of-distribution learning*. 2022. arXiv: 2109.14501.

[20] Pengzhen Ren et al. *A Survey of Deep Active Learning*. 2021. arXiv: 2009.00236.

[21] Zhihao Peng et al. “Active Transfer Learning”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 30.4 (2020), pp. 1022–1036.

[22] Yang Liu et al. “Imbalanced data classification: Using transfer learning and active sampling”. In: *Engineering Applications of Artificial Intelligence* 117 (2023), p. 105621.

[23] Xing Wu et al. *Conditional BERT Contextual Augmentation*. 2018. arXiv: 1812.06705.

[24] Charles Yang Kundo Park Grace X. Gu Seunghwa Ryu Yongtae Kim Youngsoo Kim. “Deep learning framework for material design space exploration using active transfer learning and data augmentation”. In: *npj Computational Materials* 7.1 (2021), p. 140.

[25] Chun-Teh Chen and Grace X. Gu. “Generative deep neural networks for inverse materials design using backpropagation and active learning”. In: *Advanced Science* 7.5 (2020).

[26] Qingxiu Dong et al. “A survey on in-context learning”. In: *arXiv preprint arXiv:2301.00234* (2022).

[27] Keyulu Xu et al. “How neural networks extrapolate: From feedforward to graph neural networks”. In: *arXiv preprint* (2020).

[28] Angela Fan, Mike Lewis, and Yann N. Dauphin. “Hierarchical Neural Story Generation”. In: *arXiv preprint* (2018). URL: `http://arxiv.org/abs/1805.04833`.

[29] Ari Holtzman et al. *The Curious Case of Neural Text Degeneration*. 2020. URL: `https://arxiv.org/abs/1904.09751`.

[30] Peng Dai et al. “Accurate inverse design of Fabry&#x2013;Perot-cavity-based color filters far beyond sRGB via a bidirectional artificial neural network”. In: *Photon. Res.* 9.5 (2021), B236–B246.

[31] Haozhu Wang et al. “Automated multi-layer optical design via deep reinforcement learning”. In: *Machine Learning: Science and Technology* 2.2 (2021), p. 025013.

[32] Jeff Johnson, Matthijs Douze, and Hervé Jégou. “Billion-scale similarity search with GPUs”. In: *IEEE Transactions on Big Data* 7.3 (2019), pp. 535–547.

[33] Cheng Zhang, Rami ElAfandy, and Jung Han. “Distributed Bragg Reflectors for GaN-Based Vertical-Cavity Surface-Emitting Lasers”. In: *Applied Sciences* 9.8 (2019).

[34] Chunlei Sun et al. “Broadband and high-resolution integrated spectrometer based on a tunable FSR-free optical filter array”. In: *ACS Photonics* 9.9 (2022), pp. 2973–2980.

[35] Chunqi Jin and Yuanmu Yang. “Transmissive nonlocal multilayer thin film optical filter for image differentiation”. In: *Nanophotonics* 10.13 (2021), pp. 3519–3525.

A OptoGPT Architecture

	OptoGPT Pretrain	**Finetune on Augmented Dataset (per round)**
Parameter (M)	58	
Number of Layers	6	
Hidden Dim	1024	
Input Dim	142	
Dropout	0.1	
Optimizer	Adam	
Training Size	10M	200k added
Batch Size	1000	1000
Learning Rate	1e-4	5e-5
Epoch	200	10
LR Decay	Cosine	None

Table 1: Training Details

OptoGPT consists of Transformers Decoder Layers with cross attention mechanism with the input spectrum. In each round of data augmentation, we add 2% of the original dataset (about 200k data) into finetuning stage for 10 epochs with a fixed learning rate of 5e-5.