# A Surrogate-Assisted Extended Generative Adversarial Network for Parameter Optimization in Free-Form Metasurface Design

Manna Dai[a], Yang Jiang[b], Feng Yang[a], Joyjit Chattoraj[a], Yingzhi Xia[a], Xinxing Xu[a], Weijiang Zhao[c], My Ha Dao[d], Yong Liu[a,*]

[a] *Computing and Intelligence Department, Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), 138632, Singapore*
[b] *Shenzhen University, 518060, PR China*
[c] *Electronics and Photonics Department, Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), 138632, Singapore*
[d] *Fluid Dynamics Department, Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), 138632, Singapore*

**Abstract**

Metasurfaces have widespread applications in fifth-generation (5G) microwave communication. Among the metasurface family, free-form metasurfaces excel in achieving intricate spectral responses compared to regular-shape counterparts. However, conventional numerical methods for free-form metasurfaces are time-consuming and demand specialized expertise. Alternatively, recent studies demonstrate that deep learning has great potential to accelerate and refine metasurface designs. Here, we present XGAN, an extended generative adversarial network (GAN) with a surrogate for high-quality free-form metasurface designs. The proposed surrogate provides a physical constraint to XGAN so that XGAN can accurately generate metasurfaces monolithically from input spectral responses. In comparative experiments involving 20000 free-form metasurface designs, XGAN achieves 0.9734 average accuracy and is 500 times faster than the conventional methodology. This method facilitates the metasurface library building for specific spectral responses and can be extended to various inverse design problems, including optical metamaterials, nanophotonic devices, and drug discovery.

*Keywords:* Fifth-generation (5G), free-form metasurfaces, generative adversarial network (GAN), surrogate, inverse design.

## 1. Introduction

This Metasurfaces, as the two-dimensional (2D) counterparts of metamaterials, enable the manipulation of electromagnetic (EM) behaviors, such as phases, amplitudes, and polarization of reflected/transmitted waves in space [1, 2, 3]. With the advent of fifth-generation (5G) wireless communication, metasurfaces have garnered considerable scientific and engineering interest due to the advantages of planar profile, easy integration, and low loss [4]. The concept of "digital

*Corresponding author
*Email address:* liuyong@ihpc.a-star.edu.sg (Yong Liu)

metamaterials" allowed for the coding of metasurfaces to achieve different EM responses [5, 6]. The conventional metasurface design involves three steps: determining metasurface patterns by a metamaterial specialist, achieving the EM responses of patterns with the aid of full-wave EM simulators, and selecting the metasurface with desired EM responses. Nevertheless, the pattern design requires the expertise of a specialist. Moreover, the metasurface selection relies on trial and error [7], requiring tremendous iterative EM simulations to achieve a metasurface with a satisfactory spectral response. Consequently, there is a pressing need to quickly predict the EM responses of various digital patterns to facilitate rapid metasurfaces design [8, 9].

Deep learning [10] has replaced a lot of manual work in metasurface design, such as metasurface coding [11] and real-time response simulation [12]. The metasurface design involves forward modeling and inverse design [13, 14]. Forward modeling, also known as surrogate modeling, predicts EM response without EM simulation by using structural patterns as input [15, 16]. On the other hand, inverse design quickly produces structural patterns with the input of EM response [17, 18]. This on-demand design method overcomes the limitations of traditional methods, which are generally time-consuming, inefficient, and experience-dependent. Each metasurface design exhibits a different degree of freedom (DOF), comprising input DOF, parameter DOF, and output DOF [19]. Our study focuses on the parameter DOF, as it aims to control the geometric parameters of complex metasurface structures to realize desired EM responses. Liu et al. [20] introduced deep learning as a substitute to solve the inverse design problem in photonics and optics, particularly for structure designs with high parameter DOF that are too intricate for traditional methods to handle. Deep learning models for inverse design can be categorized into discriminative models and generative models. Discriminative models capture the relationship between structural patterns and optical responses, but cannot perfectly map an optical response to a unique set of design parameters due to multiple configurations of patterns corresponding to the same response. Thus, discriminative models are suitable for low DOF structure designs and can be implemented with fundamental network architectures such as fully convolutional networks (FCN) [21, 22] or convolutional neural networks (CNN) [23, 24]. As DOF continues growing to thousands and more, generative models become useful in reducing the design's dimensionality and seeking relationships between structural patterns and optical responses for optimization. Generative adversarial networks (GANs) [25, 26] and variational autoencoders (VAEs) [27] are two commonly used deep generative models that capture the distribution of high-dimensional data and represent it in a reduced-dimensional latent space. The generative models can be jointly leveraged with discriminative models and optimization algorithms to accelerate the design process and locate global optimal solutions.

Some studies have demonstrated the efficacy of using deep learning for synthesizing metasurfaces with extra parameter DOF. Liu et al. [28] employed a GAN to produce patterns with arbitrary topology for desired input responses. They focused on the basic shapes of binary images, including the classes of L-shape, circle, arc, square, rectangle, ellipse, and sector. Ma et al. [29] utilized VAEs to encode metamaterial patterns and optical responses into a shared latent space, allowing for the automatic clustering of similar patterns and responses. Candidate patterns were generated by sampling the latent space given requirements in the decoding process. Naseri and Hum [30] used a VAE to encode multilayer EM metasurfaces into latent vectors, which were searched using particle swarm optimization (PSO) [31] to find an optimum feature that closely matched the desired transmission coefficient. The obtained feature was then decoded by the VAE to obtain the metasurface design. An et al. [32] presented a GAN that generated metasurfaces to meet multifunctional design goals, producing free-form structures in a single design iteration instead of using iterative optimization. Dai et al. [33] combined a GAN with a CNN-based forward model, to design free-form patterns with binary images, which could ensure that the spectral responses of generated patterns resembled the given input ones.

Although these deep learning-assisted metasurface design methods emerge in the past few years, there exist the following limitations to be resolved. Firstly, most networks tackle some simple typical geometries with limited parameter DOF, such as square [28], H-shape [29], and circle [34]. These simple geometries impose constraints on the capacity of metasurfaces to effectively produce desired responses [35]. Therefore, the generation of free-form geometries is necessary for providing high parameter DOF. Secondly, some networks produce metasurface designs that are memorized rather than generating novel solutions by learning the underlying design principles. Inverse design, as a one-to-many problem, can result in multiple patterns corresponding to a specific response. Thirdly, based on research by Yla-Oijala et al. [36], achieving calculation convergence for the vertex-to-vertex structure in binary matrices requires a large condition number. However, existing networks often have insufficient condition parameters, leading to numerical calculation uncertainty.

To address the above limitations, we conceptually propose an extended generative adversarial network guided by a pretrained surrogate, called XGAN. This network leverages a GAN for automatically designing free-form metasurface patterns that possess desired spectral responses. We also design a ternary coding strategy to mitigate the impact of vertex-to-vertex structure on calculation convergence. We treat the surrogate modeling as a 2D image encoding process that produces 1D features. To address this challenge, we draw inspiration from the contrastive language-image pretraining (CLIP) [37], which has demonstrated impressive performance in image-to-text mapping tasks by effectively leveraging large-scale, unstructured data. Although CLIP primarily focuses on aligning visual and textual representations, the ability of its image encoder to capture the semantics of unstructured data can be adapted to improve metasurface design tasks. We adopt a modified ResNet [38] with a multi-head attention model [39] to build the model, which is referred to as forward-ResNet (F-ResNet). We train the F-ResNet alongside a GAN to generate free-form ternary metasurface patterns capable of manipulating gigahertz (GHz) waves, which helps mitigate non-uniqueness issues in metasurface inverse designs. With GAN's rapid sampling capability, our XGAN model is competent to generate a thousand independent patterns in a second, with an accuracy of 0.9734. In this study, we developed evaluation metrics to assess XGAN, namely model fitting metrics. Our comparative analysis shows that XGAN is exceptionally effective for metasurface inverse designs. Generating high-quality patterns that match desired responses is crucial in fields like optics, electronics, and materials science, especially in the context of 5G millimeter-wave chip integration regarding frequency-selective performance. XGAN holds tremendous potential to significantly impact these fields by delivering high-quality, easy-to-manufactured high-DOF metasurface variants.

The contributions of this work are summarized as follows.

1) Pioneer the use of image-to-text encoders to create a surrogate model capable of predicting the 1D spectral responses of 2D metasurface patterns.
2) Innovatively leverage deep learning techniques to design ternary metasurface patterns, effectively mitigating the challenges posed by vertex-to-vertex structures and enhancing calculation convergence.
3) Design the surrogate model F-ResNet, which outperforms other CNN-based methods, offering precise response predictions and maintaining adherence to essential physical constraints during the tuning of XGAN.
4) Propose the XGAN inverse model, which achieves a remarkable 500-fold speed improve-

ment over traditional optimization methods like simulated annealing (SA). XGAN produces high-quality metasurface patterns that precisely match the desired spectral responses, when compared to alternative deep learning-based inverse models.

5) XGAN can create new solutions based on learned design principles, overcoming the drawback of tandem networks that rely on memorized metasurface designs.

We start the rest of this paper by first reviewing the related work in Section 2. Then, the methodology of XGAN is elaborated in Section 3. Comparative experiments and evaluation results are presented in Section 4.

## 2. Related Work

### *2.1. Metasurface Inverse Design*

The inverse problem of deducing the geometric structure of a metasurface from a desired spectral response, is not easy to solve [15]. First, when a designer specifies a particular spectral response, it may not always be feasible to find metasurface patterns capable of achieving this response. Second, there is the one-to-many mapping problem [40, 41], meaning that many highly dissimilar geometric structures can yield remarkably similar spectral responses. Therefore, for a given set of desired spectral responses, there isn't a unique solution, but rather multiple feasible metasurface designs that can achieve the same goal. This challenge arises because metasurfaces with different geometric structures can be designed by various sets of physical parameters. An ideal solution to this inverse problem should address two critical aspects: (i) ascertain the existence of any valid solution and (ii) provide a set of designs that best approximate the specified response. Naseri et al. [42] utilized a response prediction tool to serve as a surrogate models in the analysis and optimization of non-uniform metasurfaces. The integration of machine learning-based surrogate models with inverse models offers a promising solution to the one-to-many inverse design problem [16].

### *2.2. Image-to-Text Mapping: CLIP*

An image-to-text encoder [43] is a type of neural network or model that takes an image as input and converts it into a textual representation or embedding. This technology is often used in the field of computer vision and natural language processing to bridge the gap between visual and textual information. It allows computers to understand and process images in a way that's more compatible with natural language processing. Applications of image-to-text encoders include: 1) Image Captioning: Generating natural language descriptions for images [44]. 2) Image Retrieval: Searching for images using textual queries [45]. 3) Object Recognition: Identifying objects and their attributes in images [46]. 4) Visual Question Answering: Answering questions about images with natural language responses [47]. 5) Scene Understanding: Describing the scene, context, and relationships in an image using text [48]. Prominent models that perform image-to-text encoding include CLIP, where both images and text are processed in a shared model to understand their relationships. These models have wide-ranging applications in fields like image classification, image generation from textual descriptions, recommendation systems, and more.

Although deep learning methods have been adopted in the area of metasurface design, to our knowledge, no researchers have attempted to employ image-to-text transformation in metasurface inverse design. As a spectral response can be regarded as a special language or text to describe a metasurface pattern, we will demonstrate that image-to-text approach can be used to automatically realize the response prediction for a given metasurface pattern.

# 3. Methodology

## 3.1. Overview

There are two components in the metasurface design framework: one is the surrogate model (F-ResNet) that predict spectral responses from metasurface patterns, and the other is the inverse model (XGAN) that automatically generates patterns with desired spectral responses.

## 3.2. Surrogate Model: F-ResNet

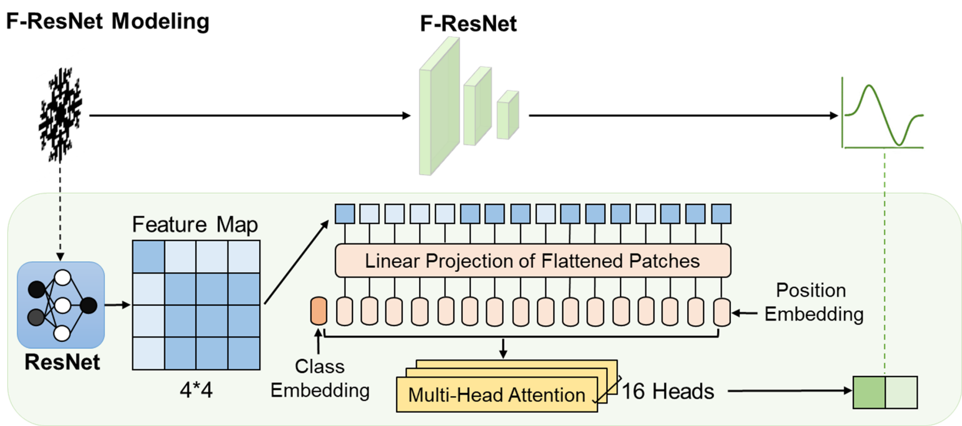


Figure 1: The network of the forward/surrogate model F-ResNet.

The framework of F-ResNet is depicted in Figure 1. We approach the forward processing as a 1D feature extraction task applied to a 2D image. Our forward model utilizes the image encoder from CLIP, which converts images into high-dimensional vector representations. The primary distinction between our work and the problems addressed by CLIP lies in the application domain and the specific task objectives. While CLIP is designed for aligning visual and textual representations in image-to-text tasks, our work focuses on leveraging its image encoding capabilities in the metasurface design task. Rather than directly applying CLIP to solve the same problems it was designed for, we leverage its image encoder as the foundation of our surrogate model. Specifically, we exploit the image encoder's ability to encode the semantics of unstructured data, such as metasurface patterns, and represent them as responses. This specific adaptation and application of its image encoder distinguish our approach from the problems traditionally addressed by CLIP.

The first step in our model is to employ ResNet50, a widely used CNN, to extract a feature map from the input image. This feature map captures high-level visual information, and then we flatten the feature map via linear projection. To generate the class embedding vector, we compute the mean of the flattened feature map. This process contributes to the improvement in model understanding of the image content and category. To provide positional information, we introduce a learnable 1D position embedding vector. This position embedding is concatenated with the class embedding as a new feature vector, enabling the model to capture spatial relationships within the image. Next, we pass the augmented feature vector through a multi-head attention model with 16 heads. This attention mechanism enhances the model's capability to encode complex sequences and capture semantic representations. Each attention head simultaneously attends to every part of the feature map, allowing for comprehensive and fine-grained analysis. Ultimately, the output of the surrogate model is considered as the extracted image feature, which serves as the response for the input image. This surrogate model works as a EM simulator, enabling us to accelerate the prediction of EM responses. Figure 1 provides a visual representation of the architecture of our surrogate model. During training, the loss function of our F-ResNet employs $L_1$ loss:

$$L_{forward}(c, c') = L_1(c, c') \tag{1}$$

where $c$ and $c'$ represent the desired response and the generated response.

### 3.3. Inverse Model: XGAN

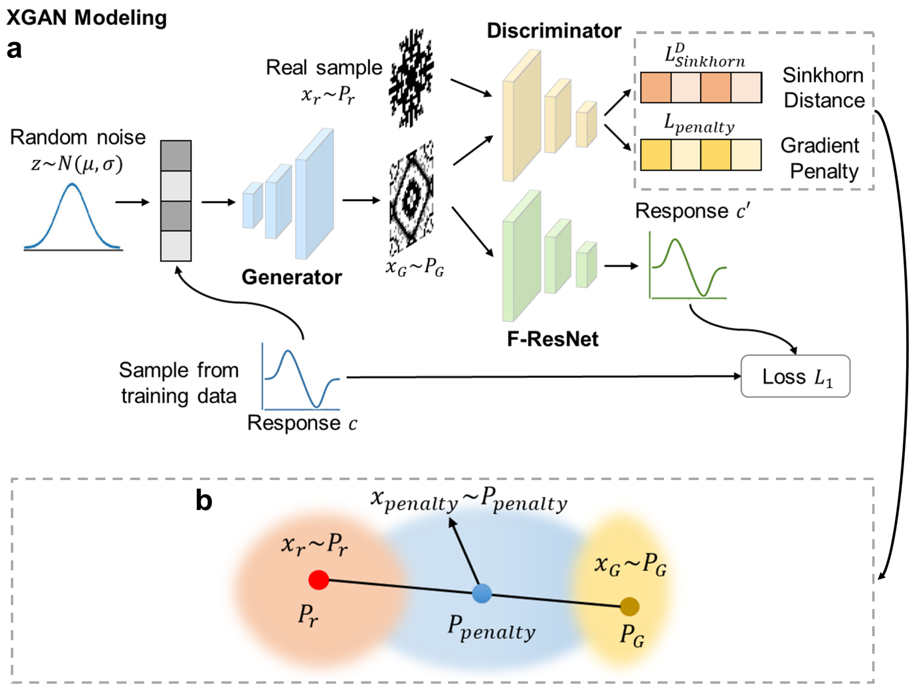


Figure 2: The modeling of XGAN. (a) The training processes of the generator and discriminator of XGAN. (b) The loss functions of XGAN consist of the discriminator loss $L_D$ and the generator loss $L_G$.

From different types of generative models, we choose GANs due to their ability to generate high-quality samples and their fast sampling capabilities. The learning process of a GAN involves an adversarial game between two neural networks, a generator, and a discriminator (Figure 2a). During the training, the response $c$ sampled from the training data is concatenated with a random noise $z$ sampled from the Gaussian distribution $N(\mu, \sigma)$. This concatenated vector is fed into the generator and outputs a pattern $x_G$ which follows the distribution $P_G$. When the generator is trained, $x_G$ is fed into the discriminator and F-ResNet, and their outputs are utilized to calculate the Sinkhorn distance $L^G_{Sinkhorn}$ and response loss $L_1$, respectively. When we train the discriminator, $x_G$ and $x_r$ are fed into the discriminator to compute the Sinkhorn distance $L^D_{Sinkhorn}$.

To deal with the issue of exploding and vanishing gradients, we gain a sample $x_{penalty}$ from the distribution $P_{penalty}$ to calculate the gradient penalty $L_{penalty}$[49]. As depicted in Figure 2b, the sample $x_{penalty}$ is calculated by the linear interpolation of $x_r$ and $x_G$.

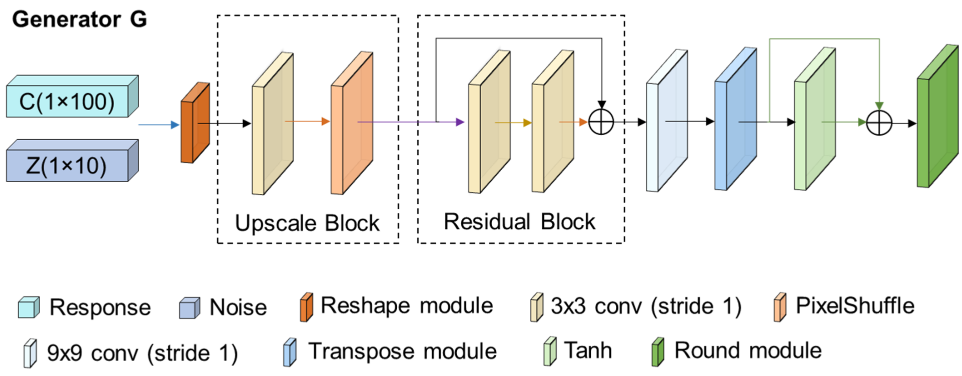


Figure 3: The generator of the proposed XGAN architecture

### *3.3.1. Generator Network G*

The detailed architecture of Generator is depicted in Figure 3. In the construction phase, the generator follows a specific sequence of layers and operations. This sequence involves generating patterns from a desired response vector of size $1 \times 100$ and random Gaussian noise of size $1 \times 10$. The construction process includes a linear layer, a parametric rectified linear unit (PReLU) layer [50] for introducing nonlinear fitting, and a reshape layer to create a 2D square matrix. To enhance the resolution to a 16×16 half-pattern size, an upscale block is employed, which consists of a convolutional layer, a batch-normalization layer, a pixel-shuffle layer [51], and a PReLU. The pixel-shuffle layer is responsible for upsampling the input, allowing for higher resolution and finer details in the synthesized pattern.

To further refine the pattern synthesis, a residual block is incorporated. This block is composed of a sequence of layers: a convolutional layer, a batch-normalization layer, a PReLU layer, another convolutional layer, and another batch-normalization layer. Notably, this residual block incorporates a skip connection of ResNet50, connecting the input and output layers to alleviate the gradient vanishing problem. Following the residual block, the output passes through a convolutional layer and a Tanh activation layer, limiting the output values to the range of $[-1, 1]$. To achieve a symmetric structure, the output of the Tanh layer is added to its transposed matrix. The resulting matrix is then multiplied by 0.5 to control the output matrix values within $[-1, 1]$. Finally, a rounding function is applied to produce a ternary matrix representation of the generated image, with values $\{0, 0.5, 1\}$. By following this specific sequence of layers and operations, the generator can synthesize high-quality patterns that closely align with the desired response vector while incorporating random noise for added variation and diversity.

### *3.3.2. Discriminator Network D*

In the discriminator architecture, the ternary pattern is provided as input. The input pattern is then fed into 2 blocks followed by a convolutional layer, a flatten layer, and an output layer. Each block comprises two convolution layers with interleaved Leaky Rectified Linear Unit (Leaky

ReLU) [52] activation layers. These convolution layers serve a dual purpose by extracting pattern features and achieving dimensionality reduction. Meanwhile, the Leaky ReLU activation layers introduce nonlinear fitting ability and facilitate the retention of relevant features. The flatten layer is utilized to reshape the input into the vector format representing the distributions of the input patterns.

During XGAN training, the generator produces patterns with symmetric shapes, while the discriminator computes the distribution distances between real patterns and synthetic patterns. The pre-trained F-ResNet approximates the spectral response of the generated pattern. The generator and the discriminator are trained sequentially: once the discriminator completes training on six batches, the parameter settings are updated and then fixed. Subsequently, the generator initiates training on a single batch. This training is guided by the updated discriminator as well as the pre-trained F-ResNet. Ultimately, the loss function of the generator is defined as

$$\begin{aligned} L_G &= L_{Sinkhorn}^{G} + \lambda * L_1 \\ &= L_{Sinkhorn}(D(x_G), 1) + \lambda * L_1(c, S(x_G)), \end{aligned} \tag{2}$$

where $G$, $D$, and $S$ are the generator, the discriminator, and the pre-trained surrogate model F-ResNet, respectively. $x_G = G(c, z)$ represents the generated pattern $x_G$ by the generator with inputs of the desired response $c$ and the random Gaussian noise $z$. $L_{Sinkhorn}$ denotes the Sinkhorn distance [53], and $\lambda$ is utilized to adjust the weights of losses.

The loss function of the discriminator is computed by

$$\begin{aligned} L_D &= L_{Sinkhorn}^{D} + \gamma * L_{penalty} \\ &= L_{Sinkhorn}(D(x_r), -1) + L_{Sinkhorn}(D(x_G), 1) \\ &+ \gamma * L_{penalty}. \end{aligned} \tag{3}$$

Here, we utilize the gradient penalty $L_{penalty}$ [49] to constrain the Sinkhorn distance, and $\gamma$ is utilized to adjust the weights of losses. The gradient penalty $L_{penalty}$ is defined as follows.

$$L_{penalty} = \mathop{\mathbb{E}}_{x_{penalty} \sim \mathbb{P}_{penalty}} [(\|\nabla_{x_{penalty}} D(x_{penalty})\|_2 - 1)^2], \tag{4}$$

where $x_{penalty}$ obeys the distribution $\mathbb{P}_{penalty}$ and is sampled via the following equation.

$$x_{penalty} = \epsilon * x_r + (1 - \epsilon) * x_G, \tag{5}$$

where, $\epsilon$ is utilized to adjust the weights of samples.

## 4. Experiment

### *4.1. Data Collection*

Figure 4a illustrates the discretization and coding of metasurface patterns. The physical dimensions of a metasurface on the 2D plane are both 5.4 mm. These metasurfaces are uniformly discretized as 32×32 matrices, which are 90° rotation symmetric and exhibit free-form shapes. In order to alleviate the impact of vertex-to-vertex structure, the coding sequence of the metasurface pattern equals to a ternary matrix, where ‘1’ means air, ‘0’ means square metal, and an additional

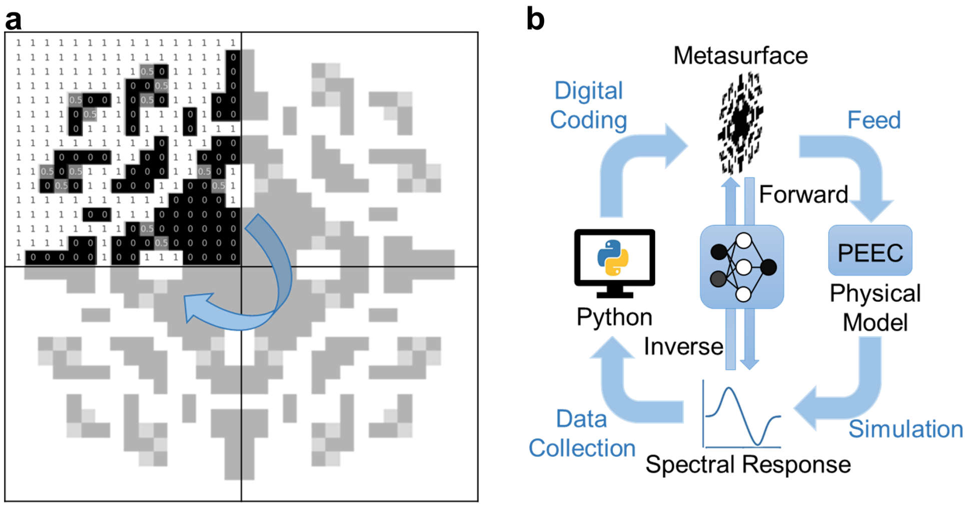


Figure 4: Data collection process. (a) Discretization and coding of the metasurface pattern. (b) The process of collecting data based on Python-PEEC co-simulation. The collected data are utilized for training neural networks that replace the manual work in forward and inverse processes.

'0.5' represents triangular metal. We use Python-PEEC co-simulation to prepare the dataset, as illustrated in Figure 4b. Python first produces a random symmetric matrix, and the partial element equivalent circuit (PEEC) method [54] then calculates the magnitude of the reflection coefficient of this matrix. We add the matrix and its corresponding response to our dataset. The magnitude of the reflection coefficient of the meta-surfaces will be sampled uniformly at 100 points between 20 and 35 GHz, whose values range from -1 to 1. Our dataset consists of 200,000 samples, of which 180,000 are used for training and 20,000 for testing.

### *4.2. Metasurface Design*

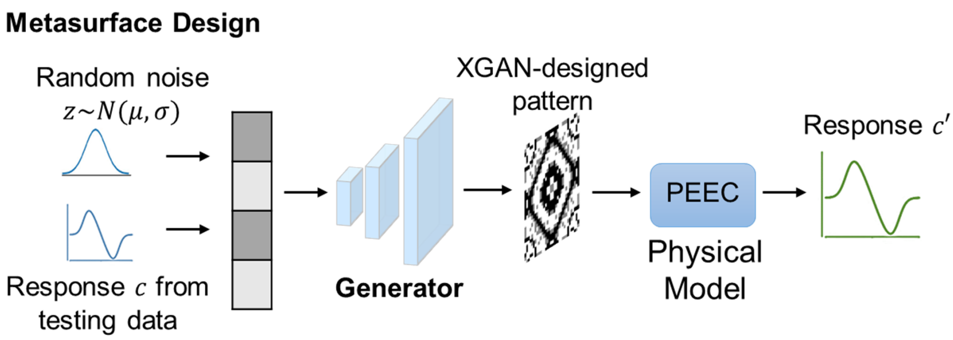


Figure 5: A sketch of metasurface design via XGAN.

Figure 5 depicts the metasurface design created with XGAN. During inference, we utilize only the generator of XGAN and remove its discriminator. To generate an initial metasurface pattern, we feed a random noise following a Gaussian distribution and a target response into the generator. PEEC is adopted to calculate the response of this pattern. If the error between the calculated response and the target response is below a predetermined threshold, the initial design pattern becomes the final design. In this paper, the random noise distribution $N(\mu, \sigma)$ is set to $N(0, 1)$.

### *4.3. Implementation Details*

The code was executed on the Ubuntu 20.04.3 LTS operating system. The deep learning algorithm was implemented using the Anaconda platform with Python 3.7.11 and PyTorch 1.7.1. These models were trained on a consumer-grade desktop that was equipped with an NVIDIA GeForce RTX 3090 GPU.

### *4.4. Training Parameters*

In this paper, we perform two group comparisons involving surrogate models and inverse models. For the surrogate models, we use a CNN as described in [33] to compare with our F-ResNet. The network settings, including kernel size, output channels, and stride, are detailed in [33]. For the inverse model comparisons, we evaluate WGAN-GP [49], InfoGAN [55], and SA [56] against our proposed XGAN.

Table 1: Execution details used in the training of surrogate and inverse models.

| | CNN [33] | F-ResNet | WGAN-GP [49] | InfoGAN [55] | SA [56] | XGAN |
|---|---|---|---|---|---|---|
| Software | Pytorch | Pytorch | Pytorch | Pytorch | Pytorch | Pytorch |
| Hardware | GPU | GPU | GPU | GPU | GPU | GPU |
| Training set size | 180000 | 180000 | 180000 | 180000 | NA | 180000 |
| Test set size | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 |
| Optimizer | Adam | Adam | Adam | Adam | NA | Adam |
| learning rate | 2e-4 | 2e-4 | 2e-4 | 2e-4 | NA | 2e-4 |
| $\beta 1$ | 0.5 | 0.5 | 0.5 | 0.5 | NA | 0.5 |
| $\beta 2$ | 0.999 | 0.999 | 0.999 | 0.999 | NA | 0.999 |
| Batch size | 256 | 256 | 256 | 256 | NA | 256 |
| Epochs | 1000 | 1000 | 10000 | 10000 | 1000 | 10000 |
| Surrogate for training | NA | NA | NA | NA | F-ResNet | F-ResNet |
| Simulator for evaluation | NA | NA | PEEC [54] | PEEC [54] | PEEC [54] | PEEC [54] |
| Evaluation Metric | Model Fitting Metrics | Model Fitting Metrics | Model Fitting Metrics | Model Fitting Metrics | Model Fitting Metrics | Model Fitting Metrics |

The F-ResNet and XGAN models are optimized using Adam optimizers [57] with specific hyperparameters. The $\beta_1$ and $\beta_2$ values are set to 0.5 and 0.999, respectively, and a fixed learning rate of 0.0002 is used. Each network is equipped with its dedicated optimizer, and training is performed with a batch size of 256. The F-ResNet model is trained for a total of 1000 epochs, while the XGAN model is subjected to an extensive training regimen spanning 10000 epochs. During XGAN training, one epoch of generator training is interleaved with six epochs of discriminator training. Execution details used in the all comparative models are listed in Table 1. The training losses of F-ResNet and XGAN are depicted in Figure 6. As the generator loss and discriminator loss tend to be stable after 10000 epochs, this demonstrates that the adversarial losses of XGAN converge well.

### *4.5. Model Fitting Metrics*

The fitting accuracy of our proposed XGAN is evaluated by four quantitative accuracy metrics $MAE_{ave}$, $ACC_{ave}$, $ACC_{min}$ and $R^2_{ave}$. Specially, the calculation of $ACC_{ave}$ and $ACC_{min}$ follow the equations in the reference [33].

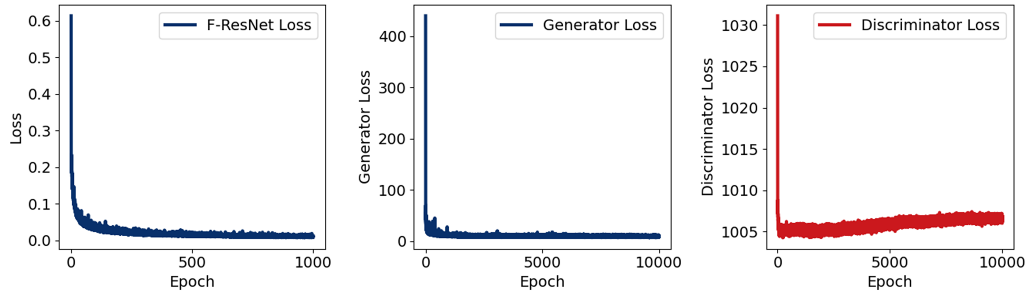


Figure 6: Training losses of the proposed models, including (a) F-ResNet loss in Eq. 1, (b) XGAN generator loss $L_G$ in Eq. 2, and (c) XGAN discriminator loss $L_D$ in Eq. 3.

1) Assume that the mean absolute error of each design is defined as $MAE$, the average $MAE$ of total $m$ designs is computed by:

$$MAE_{ave} = \frac{1}{m}\sum_{i=1}^{m} MAE(m). \tag{6}$$

2) The average accuracy of total designs is:

$$ACC_{ave} = 1 - \frac{1}{2}MAE_{ave}. \tag{7}$$

3) The minimum accuracy of total designs is:

$$ACC_{min} = 1 - \frac{1}{2}\max_{1\leq i\leq m} MAE(i). \tag{8}$$

4) Average $R^2$ is defined as follows:

$$R^2_{ave} = 1 - \sum_{i=1}^{m}\frac{MSE(i)}{Var(i)}, \tag{9}$$

where the $MSE$ and variance of each design are represented as $MSE$ and $Var$.

### 4.6. *Experimental Results*

#### 4.6.1. *Prediction Performance for Surrogate Model F-ResNet*

The comparisons are made to evaluate the effectiveness of the surrogate models in accurately predicting the response values of the metasurface designs. In this section, we use the CNN-based surrogate model from our previous work [33] as a contrastive benchmark. Density heatmaps (with 20 bins) in Figure 7 are used to compare the performance of surrogate models (CNN, F-ResNet) in terms of MAE values. The comparisons are based on 20,000 testing datasets. Specifically, F-ResNet achieves MAE values generally below 0.05, whereas the CNN model only achieves this level of accuracy in half of the cases (see Figure 7). These results demonstrate the effectiveness of our proposed F-ResNet model for the given task. Regarding simulation times, Table 2 reveals that both CNN and F-ResNet, as deep learning methods, have relatively quick simulation times for a given case. In contrast, PEEC, a numerical EM simulator, can only

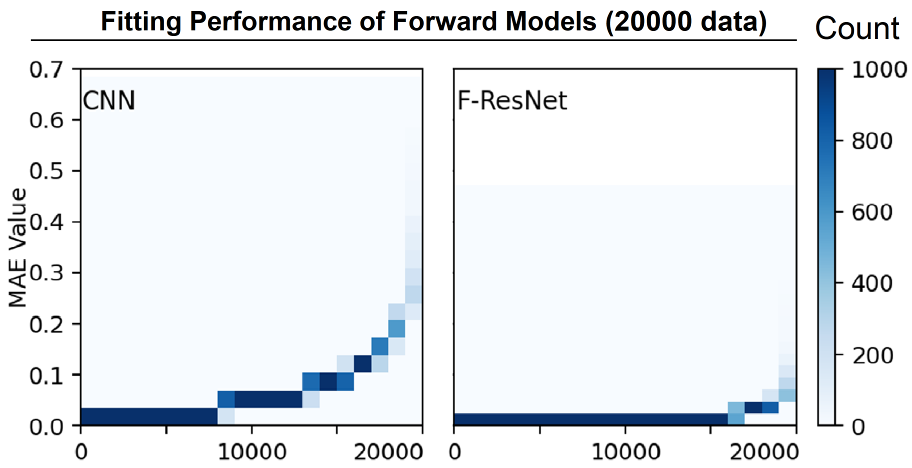


Figure 7: Density heatmaps (bins=20) of forward models (CNN, F-ResNet) comparisons regarding MAE.

simulate 1 to 3 patterns per minute. In terms of *model fitting metrics*, F-ResNet surpasses CNN in response prediction, as evidenced by superior performance in $ACC_{ave}$, $ACC_{min}$, $MAE_{ave}$, and $R^2$.

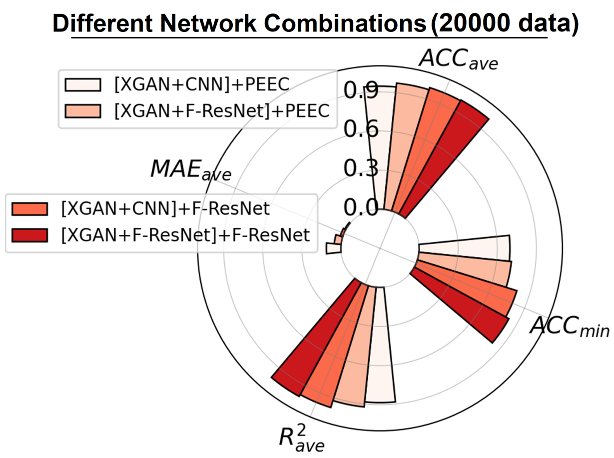


Figure 8: The polar plots of response accuracies for different network combinations (Legend meaning: [Inverse Model+Surrogate Model]+Evaluation Simulator).

We integrate F-ResNet into various network configurations to assess its efficacy in predicting responses during XGAN training, and its performance as a simulation tool for XGAN evaluation. Our results are evaluated by using the *model fitting metrics*. As presented in Figure 8, incorporating F-ResNet leads to improved XGAN performance, as opposed to using CNN. This finding is validated through evaluation, where PEEC or F-ResNet is used as a simulation tool.

Table 2: Accuracy and time cost for different forward and inverse models

| Model Type | Model Name | $ACC_{ave}$ | $ACC_{min}$ | $MAE_{ave}$ | $R^2$ | Time/Second |
|---|---|---|---|---|---|---|
| Forward/Surrogate Model | CNN [33] | 0.9644 | 0.6576 | 0.0712 | 0.8675 | 0.001 |
| | **F-ResNet** | **0.9923** | **0.7636** | **0.0154** | **0.9869** | **0.001** |
| | PEEC [54] | NA | NA | NA | NA | 20-60 |
| Inverse Model | WGAN-GP [49] | 0.8221 | 0.5338 | 0.3558 | 0.7251 | 0.0015 |
| | InfoGAN [55] | 0.8321 | 0.6096 | 0.3358 | 0.6401 | 0.0015 |
| | SA [56] | 0.8104 | 0.5335 | 0.3791 | 0.7638 | 0.75 |
| | **XGAN** | **0.9734** | **0.7132** | **0.0533** | **0.9228** | **0.0015** |

### *4.6.2. XGAN Outperforms Existing Inverse Design Methods*

The experimental approaches for evaluating inverse design in this study include WGAN-GP [49], InfoGAN [55], the simulated annealing (SA) [56], and XGAN.

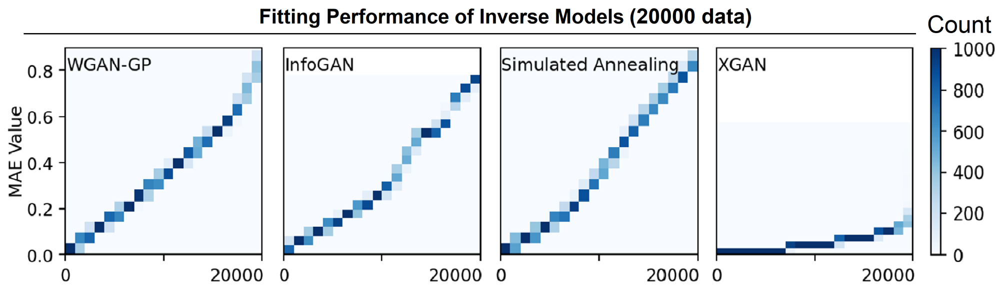


Figure 9: Density heatmaps (bins=20) of response MAE for inverse models (WGAN-GP, InfoGAN, Simulated Annealing, and XGAN).

**Quantitative Comparison:** Figure 9 highlights the superior performance of XGAN compared to other contrastive methods regarding MAE. Specifically, the MAE values achieved by XGAN are generally below 0.1, whereas WGAN-GP and the SA have the highest MAE value of 1, and InfoGAN has the highest value of 0.8. Table 2 provides the running time of each inverse model. During inference, the proposed XGAN model takes 0.0015 seconds to generate a solution for a given response.

**Qualitative Comparison:** Figure 10a provides some metasurface designs encoded by WGAN-GP, InfoGAN, SA, and XGAN. These models are fed with the target responses, and the corresponding designed patterns are simulated by PEEC to obtain the actual responses. Our aim is to determine which model could generate the highest-quality patterns that accurately matched the target responses. Specifically, XGAN generates patterns with response MAEs below 0.1, whereas the other models fail to achieve the same level of accuracy (see Figure 10b). This finding indicates the proficiency of XGAN in generating patterns with desired responses. This superior performance can be attributed to the use of interpretable latent vectors from the XGAN and the response feedback mechanism from F-ResNet, which allow for condition control and precise refinement of the generated patterns. To sum up, XGAN showcases the distinctive ability to generate appropriate patterns based on input responses, setting it apart from other models that lack this unique capability.

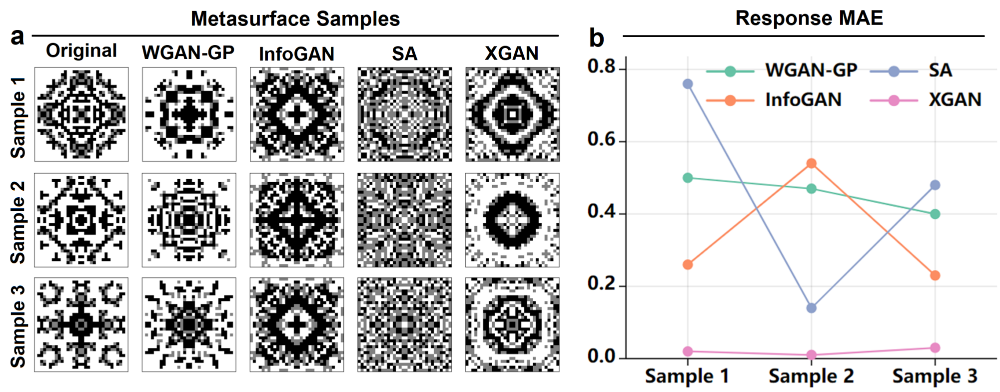


Figure 10: Free-form metasurface design by various inverse models, including WGAN-GP, InfoGAN, Simulated Annealing (SA), and XGAN. (a) Model sample and original data visualization, when Gaussian noise is $N(\mu = 0, \sigma = 1)$ and input responses are variable. (b) The MAE between the input response and the PEEC-simulated response of each sample.

## 5. Conclusion

In this paper, we present XGAN, a novel approach for the rapid and accurate realization of free-form metasurfaces. We combine ResNet50 with a multi-head attention model, named F-ResNet, to accurately predict the responses of metasurface patterns. F-ResNet exhibits impressive performance, achieving a prediction rate of 1000 patterns per second, with $ACC_{ave}$ of 0.99 and $R^2$ of 0.98. This is 20000-60000 times faster than the conventional simulator PEEC, effectively replacing it as a surrogate simulation tool. XGAN includes a GAN for inverse design. To address the limitation of traditional GAN discriminators in capturing response relevance between original and generated patterns, we integrate F-ResNet into the framework. F-ResNet acts as a constraint on XGAN training by connecting it with the generator. During the training, XGAN generates an initial pattern using input vectors, and iteratively refines it based on feedback from F-ResNet, ultimately achieving the desired response. To assess the efficiency of XGAN, we design *model fitting metrics*. XGAN achieves $ACC_{ave}$ of 0.97 for 20000 designs in a single run, and produces a metasurface pattern in 0.0015 seconds, which is 500 times faster than the conventional optimization method SA. Overall, XGAN provides a fast and accurate approach for designing free-form metasurfaces to meet the response requirement. This makes XGAN suitable for various applications, including light manipulation and spectral selection.

## CRediT authorship contribution statement

**Manna Dai:** Conceptualization, Visualization, Methodology, Formal analysis, Writing - reviewing & editing. **Yang Jiang:** Data Curation, Resources, Software, Validation. **Feng Yang:** Conceptualization, Methodology, Writing - reviewing & editing. **Joyjit Chattoraj:** Conceptualization, Methodology, Writing - reviewing & editing. **Yingzhi Xia:** Conceptualization, Methodology, Writing - reviewing & editing. **Xinxing Xu:** Conceptualization, Writing - reviewing & editing, Funding acquisition. **Weijiang Zhao:** Data Curation, Software, Writing - reviewing & editing, Funding acquisition. **My Ha Dao:** Writing - reviewing & editing, Supervision, Project administration, Funding acquisition. **Yong Liu:** Writing - reviewing & editing, Supervision, Project administration, Funding acquisition.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.


## Acknowledgements

This work was financially supported by the A*STAR AME Programmatic Fund (Grant No. Grant A20H5b0142).


## References


[1] Y. Yang, Y. Ge, R. Li, X. Lin, D. Jia, Y.-j. Guan, S.-q. Yuan, H.-x. Sun, Y. Chong, B. Zhang, Demonstration of negative refraction induced by synthetic gauge fields, Science Advances 7 (2021) eabj2062.

[2] H.-Y. Cheng, M.-J. Ye, W.-R. Chen, C.-Y. Yang, S.-W. Chu, K.-P. Chen, K.-H. Lin, Large optical modulation of dielectric huygens' metasurface absorber, Advanced Optical Materials (2023) 2300102.

[3] A. M. Shaltout, V. M. Shalaev, M. L. Brongersma, Spatiotemporal light control with active metasurfaces, Science 364 (2019) eaat3100.

[4] Z. N. Chen, T. Li, W. E. Liu, Microwave metasurface-based lens antennas for 5g and beyond, in: 2020 14th European Conference on Antennas and Propagation (EuCAP), IEEE, 2020, pp. 1–4.

[5] T. J. Cui, M. Q. Qi, X. Wan, J. Zhao, Q. Cheng, Coding metamaterials, digital metamaterials and programmable metamaterials, Light: science & applications 3 (2014) e218–e218.

[6] X. Wan, M. Q. Qi, T. Y. Chen, T. J. Cui, Field-programmable beam reconfiguring based on digitally-controlled coding metasurface, Scientific reports 6 (2016) 20663.

[7] T. Qiu, X. Shi, J. Wang, Y. Li, S. Qu, Q. Cheng, T. Cui, S. Sui, Deep learning: a rapid and efficient route to automatic metasurface design, Advanced Science 6 (2019) 1900128.

[8] R. Zhu, T. Qiu, J. Wang, S. Sui, C. Hao, T. Liu, Y. Li, M. Feng, A. Zhang, C.-W. Qiu, et al., Phase-to-pattern inverse design paradigm for fast realization of functional metasurfaces via transfer learning, Nature communications 12 (2021) 2974.

[9] L. Li, T. Jun Cui, W. Ji, S. Liu, J. Ding, X. Wan, Y. Bo Li, M. Jiang, C.-W. Qiu, S. Zhang, Electromagnetic reprogrammable coding-metasurface holograms, Nature communications 8 (2017) 197.

[10] Y. Jia, C. Qian, Z. Fan, T. Cai, E.-P. Li, H. Chen, A knowledge-inherited learning for intelligent metasurface design and assembly, Light: Science & Applications 12 (2023) 82.

[11] C. Liu, Q. Ma, Z. J. Luo, Q. R. Hong, Q. Xiao, H. C. Zhang, L. Miao, W. M. Yu, Q. Cheng, L. Li, et al., A programmable diffractive deep neural network based on a digital-coding metasurface array, Nature Electronics 5 (2022) 113–122.

[12] Y. Zhu, X. Zang, H. Chi, Y. Zhou, Y. Zhu, S. Zhuang, Metasurfaces designed by a bidirectional deep neural network and iterative algorithm for generating quantitative field distributions, Light: Advanced Manufacturing 4 (2023) 1–11.

[13] J. Zhang, C. Qian, Z. Fan, J. Chen, E. Li, J. Jin, H. Chen, Heterogeneous transfer-learning-enabled diverse metasurface design, Advanced Optical Materials 10 (2022) 2200748.

[14] Y. Jia, C. Qian, Z. Fan, Y. Ding, Z. Wang, D. Wang, E.-P. Li, B. Zheng, T. Cai, H. Chen, In situ customized illusion enabled by global metasurface reconstruction, Advanced Functional Materials 32 (2022) 2109331.

[15] C. C. Nadell, B. Huang, J. M. Malof, W. J. Padilla, Deep learning for accelerated all-dielectric metasurface design, Optics express 27 (2019) 27523–27535.

[16] Z. Li, R. Pestourie, Z. Lin, S. G. Johnson, F. Capasso, Empowering metasurfaces with inverse design: principles and applications, ACS Photonics 9 (2022) 2178–2192.

[17] I. Tanriover, D. Lee, W. Chen, K. Aydin, Deep generative modeling and inverse design of manufacturable free-form dielectric metasurfaces, ACS Photonics (2022).

[18] J. Wang, R. Xi, T. Cai, H. Lu, R. Zhu, B. Zheng, H. Chen, Deep neural network with data cropping algorithm for absorptive frequency-selective transmission metasurface, Advanced Optical Materials 10 (2022) 2200178.
[19] B. Xiong, L. Deng, R. Peng, Y. Liu, Controlling the degrees of freedom in metasurface designs for multi-functional optical devices, Nanoscale Advances 1 (2019) 3786–3806.
[20] Z. Liu, D. Zhu, L. Raju, W. Cai, Tackling photonic inverse design with machine learning, Advanced Science 8 (2021) 2002923.
[21] T. Zhang, C. Y. Kee, Y. S. Ang, L. Ang, Deep learning-based design of broadband ghz complex and random metasurfaces, APL Photonics 6 (2021) 106101.
[22] J. Yun, S. Kim, S. So, M. Kim, J. Rho, Deep learning for topological photonics, Advances in Physics: X 7 (2022) 2046156.
[23] A. Mall, A. Patil, A. Sethi, A. Kumar, A cyclical deep learning based framework for simultaneous inverse and forward design of nanophotonic metasurfaces, Scientific reports 10 (2020) 1–12.
[24] J. A. Hodge, K. V. Mishra, A. I. Zaghloul, Multi-discriminator distributed generative model for multi-layer rf metasurface discovery, in: 2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP), IEEE, 2019, pp. 1–5.
[25] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, A. A. Bharath, Generative adversarial networks: An overview, IEEE signal processing magazine 35 (2018) 53–65.
[26] Z. Wang, Q. She, T. E. Ward, Generative adversarial networks in computer vision: A survey and taxonomy, ACM Computing Surveys (CSUR) 54 (2021) 1–38.
[27] D. P. Kingma, M. Welling, Stochastic gradient vb and the variational auto-encoder, in: Second international conference on learning representations, ICLR, volume 19, 2014, p. 121.
[28] Z. Liu, D. Zhu, S. P. Rodrigues, K.-T. Lee, W. Cai, Generative model for the inverse design of metasurfaces, Nano letters 18 (2018) 6570–6576.
[29] W. Ma, F. Cheng, Y. Xu, Q. Wen, Y. Liu, Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy, Advanced Materials 31 (2019) 1901111.
[30] P. Naseri, S. V. Hum, A machine learning-based approach to synthesize multilayer metasurfaces, in: 2020 IEEE International Symposium on Antennas and Propagation and North American Radio Science Meeting, IEEE, 2020, pp. 933–934.
[31] M. Clerc, Particle swarm optimization, volume 93, John Wiley & Sons, 2010.
[32] S. An, B. Zheng, H. Tang, M. Y. Shalaginov, L. Zhou, H. Li, M. Kang, K. A. Richardson, T. Gu, J. Hu, et al., Multifunctional metasurface design with a generative adversarial network, Advanced Optical Materials 9 (2021) 2001433.
[33] M. Dai, Y. Jiang, F. Yang, X. Xu, W. Zhao, M. H. Dao, Y. Liu, Slmgan: Single-layer metasurface design with symmetrical free-form patterns using generative adversarial networks, Applied Soft Computing 130 (2022) 109646.
[34] S. An, B. Zheng, M. Y. Shalaginov, H. Tang, H. Li, L. Zhou, J. Ding, A. M. Agarwal, C. Rivero-Baleine, M. Kang, et al., Deep learning modeling approach for metasurfaces with high degrees of freedom, Optics Express 28 (2020) 31932–31942.
[35] J. Yang, K. Cui, X. Cai, J. Xiong, H. Zhu, S. Rao, S. Xu, Y. Huang, F. Liu, X. Feng, et al., Ultraspectral imaging based on metasurfaces with freeform shaped meta-atoms, Laser & Photonics Reviews 16 (2022) 2100663.
[36] P. Yla-Oijala, M. Taskinen, Calculation of cfie impedance matrix elements with rwg and n/spl times/rwg functions, IEEE Transactions on Antennas and Propagation 51 (2003) 1837–1846.
[37] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International conference on machine learning, PMLR, 2021, pp. 8748–8763.
[38] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, arXiv preprint arXiv:1512.03385 (2015).
[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).
[40] G. Jing, P. Wang, H. Wu, J. Ren, Z. Xie, J. Liu, H. Ye, Y. Li, D. Fan, S. Chen, Neural network-based surrogate model for inverse design of metasurfaces, Photonics Research 10 (2022) 1462–1471.
[41] I. Tanriover, W. Hadibrata, K. Aydin, Physics-based approach for a neural networks enabled design of all-dielectric metasurfaces, ACS Photonics 7 (2020) 1957–1964.
[42] P. Naseri, S. V. Hum, A generative machine learning-based approach for inverse design of multilayer metasurfaces, IEEE Transactions on Antennas and Propagation 69 (2021) 5725–5739.
[43] M. Żelaszczyk, J. Mańdziuk, Cross-modal text and visual generation: A systematic review. part 1—image to text, Information Fusion (2023).
[44] R. Ramos, B. Martins, D. Elliott, Y. Kementchedjhieva, Smallcap: lightweight image captioning prompted with retrieval augmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 2840–2849.

[45] G. Couairon, M. Douze, M. Cord, H. Schwenk, Embedding arithmetic of multimodal queries for image retrieval, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4950–4958.
[46] L. Yao, J. Han, X. Liang, D. Xu, W. Zhang, Z. Li, H. Xu, Detclipv2: Scalable open-vocabulary object detection pre-training via word-region alignment, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 23497–23506.
[47] F. Gao, Q. Ping, G. Thattai, A. Reganti, Y. N. Wu, P. Natarajan, Transform-retrieve-generate: Natural language-centric outside-knowledge visual question answering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 5067–5077.
[48] J. Kil, S. Changpinyo, X. Chen, H. Hu, S. Goodman, W.-L. Chao, R. Soricut, Prestu: Pre-training for scene-text understanding, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 15270–15280.
[49] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A. C. Courville, Improved training of wasserstein gans, Advances in neural information processing systems 30 (2017).
[50] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.
[51] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1874–1883.
[52] A. L. Maas, A. Y. Hannun, A. Y. Ng, et al., Rectifier nonlinearities improve neural network acoustic models, in: Proc. icml, volume 30, Atlanta, GA, 2013, p. 3.
[53] M. Cuturi, Sinkhorn distances: Lightspeed computation of optimal transport, Advances in neural information processing systems 26 (2013).
[54] Y. Jiang, W.-J. Zhao, R. X.-K. Gao, E.-X. Liu, C. E. Png, A full-wave generalized peec model for periodic metallic structure with arbitrary shape, IEEE Transactions on Microwave Theory and Techniques 70 (2022) 4110–4119.
[55] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, P. Abbeel, Infogan: Interpretable representation learning by information maximizing generative adversarial nets, Advances in neural information processing systems 29 (2016).
[56] J. Lee, D. Perkins, A simulated annealing algorithm with a dual perturbation method for clustering, Pattern Recognition 112 (2021) 107713.
[57] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).