

Template Design for Complex Block Copolymer Patterns Using a Machine Learning Method

Zhihan Liu, Yi-Xin Liu,* Yuliang Yang, and Jianfeng Li*

Cite This: *ACS Appl. Mater. Interfaces* 2023, 15, 31049–31056

Read Online

ACCESS |



Metrics & More



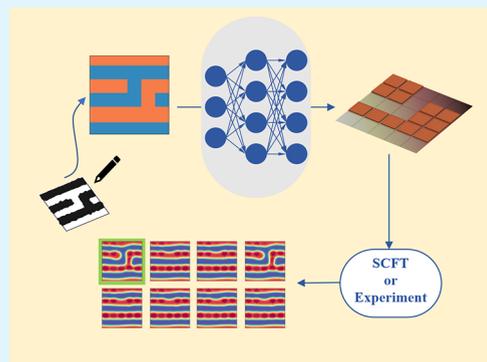
Article Recommendations



Supporting Information

ABSTRACT: This study represents the first attempt to address the inverse design problem of the guiding template for directed self-assembly (DSA) patterns using solely machine learning methods. By formulating the problem as a multi-label classification task, the study shows that it is possible to predict templates without requiring any forward simulations. A series of neural network (NN) models, ranging from the basic two-layer convolutional neural network (CNN) to the large NN models (32-layer CNN with 8 residual blocks), have been trained using simulated pattern samples generated by thousands of self-consistent field theory (SCFT) calculations; a number of augmentation techniques, especially suitable for predicting morphologies, have been also proposed to enhance the performance of the NN model. The exact match accuracy of the model in predicting the template of simulated patterns was significantly improved from 59.8% for the baseline model to 97.1% for the best model of this study. The best model also demonstrates an excellent generalization ability in predicting the template for human-designed DSA patterns, while the simplest baseline model is ineffective in this task.

KEYWORDS: block copolymer, directed self-assembly, machine learning, inverse design, lithography



INTRODUCTION

The self-assembly of block copolymers in the bulk phase can form various intricate and ordered morphologies,^{1–4} which are governed by various factors, including the Flory–Huggins interaction parameter, degree of polymerization (N), and volume fractions of blocks (f). These diverse and beautiful nanostructures or patterns offer various candidate substrate geometries for manufacturing nanodevices,^{5,6} lithography,^{7–13} biomimetics, etc. However, a drawback of block copolymer self-assembly in the bulk phase is its lack of designability, which researchers can overcome by directing the self-assembly (DSA) of block copolymers (BCPs). Compared to top-down lithography methods, DSA employs lithographically defined chemical or topological templates to guide the BCPs to self-assembly into semiconductor-relevant features, such as line/space or contact holes, at a relatively low cost.^{14–16} Currently, DSA of BCPs has found applications in various semiconductor device manufacturing processes, including fin field-effect transistors (FinFETs), memory devices, and photonic devices.^{17–19} To promote the application of DSA in high-volume manufacturing, extensive research has been focused on improving the quality of DSA patterns. This includes investigating the mechanisms of defect formation and accelerating their annihilation through experimental and simulation approaches.^{20–27}

Despite the usefulness of the DSA, its design efficiency through trial and error methods alone is limited. To accelerate

the design process, researchers often employ computer simulations, such as self-consistent field theory (SCFT),^{28–33} and machine learning techniques,^{34–39} such as random forest method³⁷ and convolutional neural network (CNN),^{40,41} to search for the desired self-assembly structures of block copolymers and to predict DSA patterns based on a given template. Nevertheless, this forward design approach is still relatively slow, particularly for complex patterns that may require thousands of attempts to achieve the desired outcome.

The inverse design method,⁴² which involves inferring the guiding template based on a given aiming pattern, offers a more efficient solution. This is also a crucial step in actual design procedures for various microelectronic and nano-electronic devices. Several forward simulation methods have been developed for the inverse design task. For instance, Khaira et al.⁴³ used a simulation–evolution method to design templates for regular striped or lamellar DSA patterns, while Hannon et al.⁴⁴ tackled more complex patterns using Monte Carlo optimization assisted by SCFT simulations, showing good agreement with the experimental results.

Received: April 7, 2023

Accepted: June 2, 2023

Published: June 19, 2023



To further promote the design efficiency, this work solely employs machine learning to perform the inverse design task directly from a complex input pattern without any auxiliary forward simulations. The designed chemical template features a 5×5 adsorption square post array that guides the self-assembly of A/B diblock copolymers. The template's square posts are either adsorptive to the A blocks (ON) or neutral to both A and B blocks (OFF). From the perspective of supervised model training, the status of the 25 template posts can be described as 25 ON/OFF labels, and the inverse design task can be modeled as a multi-label binary classification problem, as shown in Figure 1. This problem is very

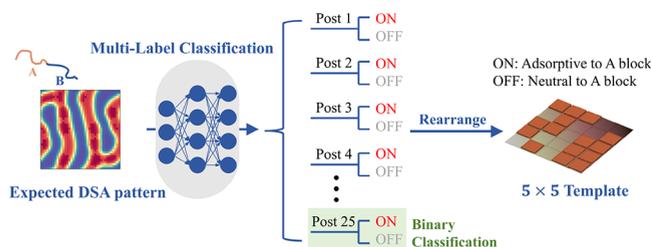


Figure 1. Inverse design for DSA patterns can be typically modeled as a multi-label classification problem. This work aims at designing a DSA pattern guided by a 5×5 post template, which corresponds to the 25-label binary classification problem.

challenging for two reasons. First, the exact-match accuracy⁴⁵ of this multi-label classification problem with random guesses is only about $1/2^{25}$, which is significantly lower than that of a one-label classification problem, such as 10-class handwritten digit recognition with the random-guess accuracy $\sim 1/10$. Second, as the mapping between the DSA pattern and the template is normally many-to-one rather than one-to-one, it may still take several attempts to obtain the desired DSA pattern based on the predicted template. As a result, there has been very little work done on performing inverse design solely through machine learning. Only a few studies have used machine learning methods to address DSA-related problems, such as evaluating the quality of templates^{46–48} or analyzing DSA pattern defects.²⁷

Therefore, this work aims to develop an efficient and robust deep-learning-based method for directly predicting templates that can guide the self-assembly of BCPs into complex target patterns (as shown in Figure 2). The data set used for model training is generated through forward simulations. First, random template post arrays are generated, with each square post either ON or OFF, which serve as the sample labels. Then, SCFT simulations are used to simulate the self-assembly patterns formed on these templates, which serve as the sample features. A number of deep-learning techniques, originally developed for image recognition, will be specifically modified to improve the accuracy of the inverse prediction of the template. These techniques have been shown to significantly enhance the accuracy of inverse prediction for simulated directed self-assembly (DSA) pattern inputs. Even for human-designed pattern inputs, containing essential lithography structures such as dense or isolated lines and spots, angles, jags, and T-junctions, the success rate is still acceptable with the improved NN model, in contrast to the original NN model's extremely low success rate on this task.

The contributions of this work are summarized as follows: (i) We present the first work to predict a DSA guiding

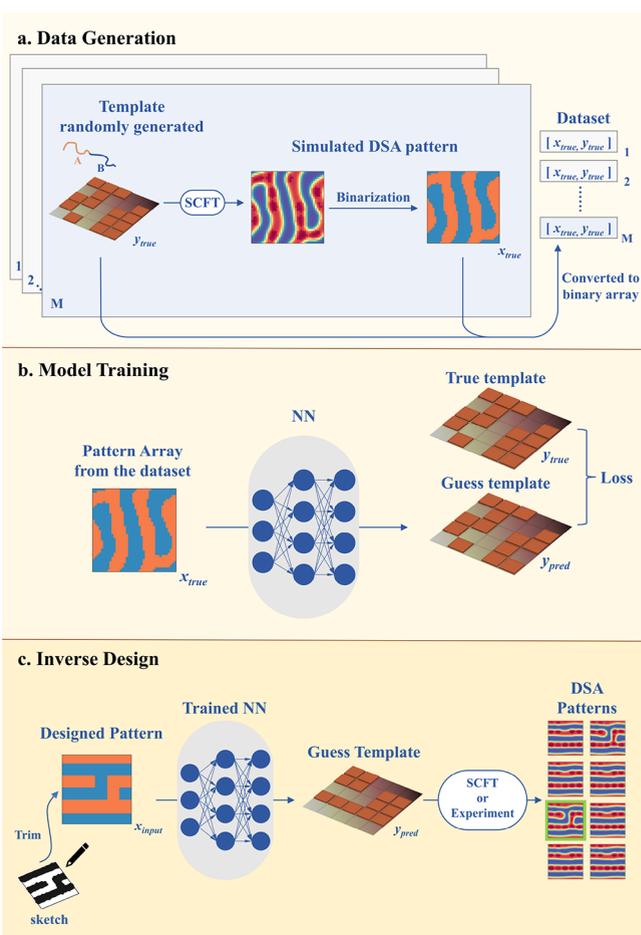


Figure 2. Flowchart for the inverse design of the guiding template for DSA patterns using the single-channel input strategy (SCIS) is shown, while that for the multi-channel input strategy (MCIS) can be found in Figure S1. The process consists of three steps: (a) Data generation: A data set is generated using SCFT simulations, where each sample consists of a template y and the corresponding DSA pattern. The ON/OFF-square-post template is randomly generated, and SCFT is used to simulate the DSA process on the template. The resulting template and DSA pattern are then converted into 5×5 and 40×40 binary arrays, respectively. Around 90,000 samples are generated in this manner. (b) Model training: The data set is used to train an NN model to learn how to accurately predict the correct template for a given input pattern. (c) Inverse Design: The trained NN model is asked to predict a template based on a human-designed pattern. One of the DSA patterns that forms on the predicted template is expected to recover the designed pattern. This can be tested by conducting experiments or using SCFT simulations.

template using only machine learning methods without relying on any forward simulations. It should be noted that SCFT simulations are only used for data generation and accuracy evaluation, as shown in Figure 2(a),(c). (ii) We propose several deep-learning techniques, including the interface noise adding augmentation (INAA) and multi-channel input strategy (MCIS), adapted from the image recognition but are specially suitable for DSA-pattern related problems. In particular, MCIS can effectively address the many-to-one mapping problem mentioned above. These techniques significantly improved the prediction accuracy from $\sim 59.8\%$ to $\sim 97.1\%$, which is a remarkable improvement compared to the random-guess accuracy of $1/2^{25}$.

METHODS

Self-Consistent Field Theory and Data Generation. The neural network (NN) in this work was trained under supervised learning using a preobtained data set generated through forward self-consistent field theory (SCFT)^{2,3} simulations (see Figure 2a). To generate the data set, a chemical template featuring a 5×5 square posts array was designed. Each post in the array was either adsorptive or neutral to one block (A block) of the equal-armed A/B diblock copolymer (BCP). The post size of the template was slightly larger than half of the characteristic period of the BCP to produce a more refined pattern from a coarse template. Approximately 90,000 ON/OFF-post array templates with a size of 5×5 were randomly generated, and the target patterns were produced by 2D SCFT simulations with a periodic boundary condition in the x - y plane. Obviously, the simulation box in this work is symmetric along the z -direction, where the generated DSA patterns will exhibit higher fidelity and better designability compared to those of DSA using the free surface fields.

The SCFT simulations in this work use the same equations and numerical schemes as those presented in refs 2 and 3. The model setup for the SCFT simulations is as follows: the total chain length of the block copolymer is set to 80 with a Kuhn length of $b = 1$, and $\chi N = 20$ with χ being the Flory–Huggins parameter and N being the chain length. The attractive potential of the template on the A block is expressed as $\varepsilon \delta_A(\mathbf{r})$, where $\varepsilon = 1.6k_B T$ and $\delta_A(\mathbf{r}) = 1$ for \mathbf{r} inside the square post adsorptive to A, while $\delta_A(\mathbf{r}) = 0$ otherwise (see Figure S10). The potential exerted on the B block by the template is set to zero for all \mathbf{r} . A forward SCFT calculation begins with a pair of random initialized fields $\omega_A(\mathbf{r})$ and $\omega_B(\mathbf{r})$ and undergoes a sufficient number of iterations until the fields are self-consistent or the free energy of the assembly reaches its local minimum.

Due to the stochastic nature of SCFT, even for a given template, different initial random fields can result in different assembly BCP patterns, which correspond to different metastable patterns observed in experiments. Therefore, in the real forward SCFT simulations, about 20 random initial fields are typically tried, and the patterns with the minimum free energy are added to the data set. It is important to note the following: (i) There is no algorithm to guarantee that the BCP pattern generated by the SCFT is the global minimum in free energy.² (ii) The many-to-one (many patterns to one template) mapping problem can increase the complexity of the prediction work and reduce the prediction accuracy. (iii) The defects^{25–27} arising from the periodic boundary conditions of the simulated box and the incommensurability between the box dimensions and those of the microphase-separated domains would significantly complicate the prediction task (also see Figure S11). Fortunately, this problem can be addressed in the following context through the implementation of a multiple-channel input method.

Both the templates and aiming patterns are transformed into two-dimensional binary arrays, where 0 and 1 represent the neutral (OFF) and adsorptive (ON) square posts for the template, respectively, and for the aiming pattern, they represent A and B microdomains, respectively. The binary pattern array x is of 40×40 in shape, which describes the same sized physical space as the 5×5 template array y .

In this work, two different input strategies will be employed in the NN model building, and accordingly two different data sets are generated using SCFT calculations. In the single-channel input strategy (SCIS), only one DSA pattern obtained from SCFT is input into the NN model and each sample of the data set for SCIS consists of one randomly generated template and one corresponding DSA pattern (Figure 2); while each sample of the data set for multi-channel input strategy (MCIS) consists of a randomly generated template and five corresponding DSA patterns (Figure S1). Each data set (90,000 samples) will be further divided into three distinct groups: training data set (80,000 samples), validation data set (2,220 samples), and test data set (3,448 samples). These data sets are mutually exclusive, meaning that no template in one data set can be transformed into a template in another data set through data augmentation operations. To ensure the independence between the validation and test data sets

and the training data set, over 4,000 samples from the remaining 10,000 samples have been included.

Machine Learning and Inverse Design. In contrast to forward simulations, the neural network (NN) in the inverse design takes the BCP pattern x as input and the template y as the output. It is worthwhile to note that the output array is converted to one dimension, while the input array is retained in its two-dimensional form to preserve two-dimensional physical information. This makes the task analogous to multi-label classification in image recognition, i.e., to predict 25 labels based on a 40×40 image (see Figure 1).

Also note that in this work, the input can be either single-channel (Figure 2) or multi-channel (Figure S1). In the single-channel NN, the input comprises only one of the BCP patterns formed on the given template. Conversely, in the multichannel NN, we input five patterns formed on the given template into the NN.

During the training process (Figure 2(b)), layer parameters are initially generated using the default initialization method of tensorflow2.0 and progressively optimized during training. The model adjusts all the parameters to minimize the value of a loss function that quantifies the degree of deviation between the predicted and the true values. In this multi-label classification problem, we use binary cross entropy (BCE) as the loss function,⁴⁹ which computes the average cross entropy for each label. The value of BCE is calculated using the following equation:

$$L[f_i(x)] = -\frac{1}{m} \sum_{i=1}^m [y_i \log f_i(x) + (1 - y_i) \log(1 - f_i(x))] \quad (1)$$

where x is the input array, m is the label number ($m = 25$), $i \in [1, m]$ represents the i th label, y_i equals 0 or 1 representing the i th true label, and $f_i(x) \in [0, 1]$ represents the i th predicted label by the NN. The Adam optimization method,⁴⁹ based on the gradient descent algorithm, is used to optimize the BCE loss during the training process. In each epoch, the model processes all the training data set samples, on average. The model is trained on the training data set for a specified number of epochs, where a validation-based early stopping strategy⁴⁹ will be applied to prevent overfitting (see Figure S5).

After the training process, the NN can be applied to the test samples generated by SCFT simulations or to the inverse design of human-designed patterns (Figure 2(c)).

In the application of the trained NN on test samples, the prediction accuracy is evaluated by directly comparing the template predicted by NN and the true template obtained by SCFT simulations (see Figure 2(c) and eqs 2 and 3).

In the inverse design or the application of the trained NN on hand-designed patterns, there is no true template to evaluate the success of the inverse design. Thus, we used the predicted template to perform a series of SCFT simulations or experiments to examine whether they can produce one or more patterns that closely resemble the human-designed pattern. We define the success rate of the inverse design as the percentage of human-designed patterns that are correctly reproduced by the SCFT simulations based on the predicted template given by the NN.

Neural-Network Architectures. In this work, several neural network architectures have been employed. These architectures consist of building blocks of dense layers, CNN layers,⁴⁹ residual blocks (ResNet),⁵⁰ and a self-conscious neural network (SCN) module (Figure S2). CNN layers are commonly used in object recognition and are particularly effective at extracting features from 2D structural data. ResNet was originally developed to address the vanishing gradient problem and has been widely used in various fields. The fundamental concept behind the SCN module is to utilize the NN's inner state to enhance its performance (see section II in SI for more details).

In this study, we investigated a total of 32 neural networks. Four CNN networks were examined, namely CNN₂, CNN₈, CNN₂₄, and CNN₃₂, where CNN _{n} refers to a deep CNN with n layers. Additionally, four residual neural networks were explored, namely, CNN₂₄-Res₁₂, CNN₃₂-Res₈, CNN₃₂-Res₁₂, and CNN₃₂-Res₁₆, where CNN _{n} -Res _{m} denotes a ResNet with n CNN layers and m residual

blocks (refer to Table S2 for more details). Furthermore, we introduced the SCN module into these eight neural networks, forming an additional eight NNs, which we denote as $\text{CNN}_n\text{-Res}_m\text{-SCN}$. We also considered two different types of NN inputs, namely, single-channel and multichannel input. We labeled the multichannel input NN with an added “M-” in front of the name. For example, $\text{M-CNN}_n\text{-Res}_m$ is a multichannel Resnet.

In the following context, we shall use the notation $\text{X-CNN}_n\text{-Res}_m\text{-SCN}_Y$ to refer to the 32 NNs explored in this work, where X can be either M (multi-channel) or S (single-channel), n represents the number of CNN layers employed, m represents the number of residual blocks employed, and Y can be either ON (with SCN module) or OFF (without SCN module). For a more detailed description of these NN structures, please refer to Tables S1 and S2.

Data-Augmentation Techniques. In this work, we employed four data augmentation techniques to enhance the performance of NNs: rotation and reflection (R&R), pattern broadening (PB), random window moving (WM), and interface noise-adding augmentation (INAA).

The R&R technique applies rotation and mirror-reflection operations to the original images to increase the number of samples in the training data set. Specifically, eight additional samples are generated for each original image.

The PB augmentation technique broadens the vision of NNs by increasing the size of the pattern from 40×40 to 48×48 (Figure S3), which injects more information into NNs to obtain better predictions.

The WM augmentation technique is adapted from random-crop augmentation in object recognition. Since the DSA patterns generated in our work have periodic boundary conditions, nine copies of the original pattern can be tiled to form a larger pattern (120×120), and a 40×40 (or 48×48) window is randomly moved with steps compatible with the square-post size of the template to crop the new pattern (see Figure S4).

For the INAA augmentation technique, we designed a strategy that specifically targets the inverse-design problem of DSA patterns and is suitable for other problems that involve predictions related to phase-separated patterns. The core idea is that NNs should ignore slight changes in the interfaces of different phase domains. However, randomly adding noise can create new small phase domains that do not improve prediction accuracy (Figure 3, left). Therefore, instead of adding random noise, we flip the colors only on the boundary between phase domains to maintain the original domain structure (see Figure 3).



Figure 3. DSA patterns in the training data set after traditional noise adding augmentation (left) and interface noise adding augmentation INAA (right).

RESULTS AND DISCUSSION

Evaluation Metrics. To evaluate the performance of our trained NNs, we counted the number of test samples (denoted as $m_t(N_{\text{ON/OFF}})$) for which the number of correctly predicted ON/OFF template blocks is exactly $N_{\text{ON/OFF}}$ (see Figure 4 and Figure S6). For example, when $N_{\text{ON/OFF}} = 25$, this indicates that the predicted template is fully correct. Using $m_t(N_{\text{ON/OFF}})$, we calculate the exact match (EM) accuracy⁴⁵ and overall precision in multi-label classification as follows:

$$A_{\text{EM}} = \frac{m_t(25)}{M} \quad (2)$$

$$P = \frac{\sum_{i=1}^{25} i \cdot m_t(i)}{25M} \quad (3)$$

where $M = 3448$ represents the total number of test samples in this work. The EM accuracy metric considers only fully corrected predicted templates and is considered to be harsh, while the overall precision metric considers partially correct predictions. For instance, the EM accuracy for the baseline model is $2061/3448 \approx 59.8\%$ (see the green bars in Figure S6(a)), while its overall precision is much higher ($(2061 \times 25 + 1007 \times 24 + \dots)/3448/25 \approx 97.9\%$). However, the poor performance of the baseline model indicates that the overall precision of the model can be misleading in this case. Therefore, we primarily rely on EM accuracy to evaluate models in the following context.

Poor Performance of the Baseline CNN Model. As mentioned in the introduction, the task of inverse design for DSA patterns can be reduced to a multi-label classification problem. However, predicting all of the labels (ON/OFF template posts) on the template correctly is extremely challenging, as there are 2^{25} possible template configurations. In fact, with random guesses, the accuracy is only $1/2^{25} \approx 2.3 \times 10^{-8}$, which is much lower than the accuracy of $1/10$ for the 10-class classification problem of handwritten digit recognition.

The poor performance of the baseline CNN model ($\text{S-CNN}_2\text{-Res}_0\text{-SCN}_{\text{OFF}}$) on this problem confirms the above analysis. The baseline model (see column (a) in Table S1) has only one single-channel input layer, two CNN layers, and two dense layers. It achieves an EM accuracy of only about 59.8% on the test data set (see the first row in Table 1.; also see Figure S6(a) for other $N_{\text{ON/OFF}}$ results), which makes it almost useless for the inverse design task of human-designed patterns (Figures 5 and S7).

Impact of Data-Augmentation Techniques on NN Performance. The baseline model trained by the original training data set achieves an EM accuracy of only 0.598 on the test data set. However, the performance can be significantly improved by applying data augmentation techniques, as discussed in the “Data-Augmentation Techniques” section.

By applying rotation (90° , 180° , 270° , and 360°) and mirror-reflection (R&R) operations to the original training data set, a new data set is generated that is 8 times larger. This leads to an improvement in the prediction accuracy to 0.639. The pattern broadening (PB) technique, which utilizes the periodic boundary condition in the SCFT data generation process to add extra boundary information to the input pattern, further improves the prediction accuracy to 0.667. Moreover, by using the window movement method, which exploits the periodic boundary condition to effectively increase the size of the training set by 6-fold, the prediction accuracy is significantly enhanced to 0.722 (Table 1). It is also worth noting that the baseline model with data augmentation techniques converges much faster than the original baseline model, as evidenced by the decrease in the early stopping epoch shown in Table 1.

Impact of NN Architecture on Model Performance. In Table S1, we progressively deepened the baseline CNN model with 2 CNN layers to a 32-CNN-layer model, and introduced interlayer structures, including SCN (Figure S2) and residual structure⁵⁰ (Table S2), to facilitate data integration and cross-layer data transmission in a deep CNN architecture.

The SCN, which contains a self-consistent loop,^{51,52} acts as a fine-tuning mechanism for the variables in the dense layers of

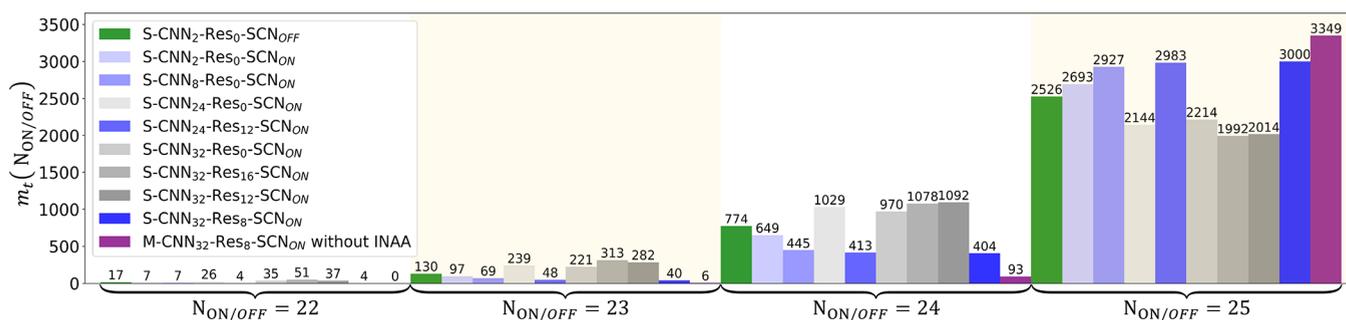


Figure 4. Performance of ten typical NNs on the test data set (3448 samples). The metric used is the number of correctly predicted ON/OFF posts ($N_{ON/OFF}$) of the 25 total 25 ON/OFF posts in the template. $m_t(N_{ON/OFF})$ is the number of test samples with a correct prediction of $N_{ON/OFF}$ posts (eqs 2 and 3). For example, the green bars correspond to the model S-CNN₂-Res₀-SCN_{OFF}, which correctly predicted all 25 ON/OFF blocks for 2526 test samples, while partially correct predictions (24/25) were obtained for 774 test samples. The naming conventions for the NNs are described in the inset of Table 2.

Table 1. Exact Match Accuracy (EM Acc.) and the Overall Precision (Prec.) of the Baseline Model on the Test Dataset without (First Row) or with the Listed Data Augmentation Techniques^a

augmentations	EM Acc.	Prec.	ES Epoch
	59.8%	97.9%	236
R&R	64.0%	98.2%	52
R&R + PB	68.2%	98.4%	46
R&R + PB + WM	73.3%	98.7%	21

^aObtained at the corresponding early-stopping (ES) epoch. The ES epoch is determined by the validation accuracy curves (see Figure S2). R&R: rotation and reflection; PB: pattern broadening; WM: window movement (also see Figures S3 and S4). See Figure S6(b) for other $N_{ON/OFF}$ results.

the CNN, leading to a significant improvement in prediction accuracy to approximately 0.781 (as shown in Table 2).

Furthermore, the inclusion of the residual structure in the model helped to alleviate the vanishing gradient problem that may arise when the model is deepened beyond 24 CNN layers. Consequently, the 32-CNN-layer model achieved a maximum accuracy of 0.870, beyond which there is no significant improvement in prediction accuracy (as demonstrated in Table 2).

Table 2. Exact Match Accuracy (EM Acc.) and the Overall Precision (Prec.) of Six Typical NNs on the Test Dataset at Their Corresponding Early-Stopping (ES) Epoch^a

NN model	EM Acc.	Prec.	ES Epoch
S-CNN ₂ -RES ₀ -SCN _{ON}	78.1%	99.0%	20
S-CNN ₈ -RES ₀ -SCN _{ON}	84.9%	99.3%	19
S-CNN ₂₄ -RES ₁₂ -SCN _{ON}	86.5%	99.4%	17
S-CNN ₃₂ -RES ₈ -SCN _{ON}	87.0%	99.4%	16
M-CNN ₃₂ -RES ₈ -SCN _{ON}	97.1%	99.9%	8
M-CNN ₃₂ -RES ₈ -SCN _{ON} ^b	95.5%	99.8%	13

^aDetermined by the validation accuracy curves shown in Figure S2. The models are named according to the following rules: X-CNN_n-Res_m-SCN_Y (see the “Neural-Network Architectures” section and Table S2), where X denotes M (multi-channel) or S (single-channel), n denotes the number of CNN layers used, m denotes the number of residual blocks used, and Y denotes ON (with SCN module) or OFF (without SCN module). ^bThe model in the sixth row includes INAA. See Figure S6(c),(d) for other $N_{ON/OFF}$ results.

The Role of Multichannel Input Strategy. Using the multichannel input strategy (Figure S1), the best performing model (M-CNN₃₂-RES₈-SCN_{ON}) in this work achieves an EM accuracy of 0.971 and an overall precision of 0.999 (as shown in Table 2). This result indicates that five channels are

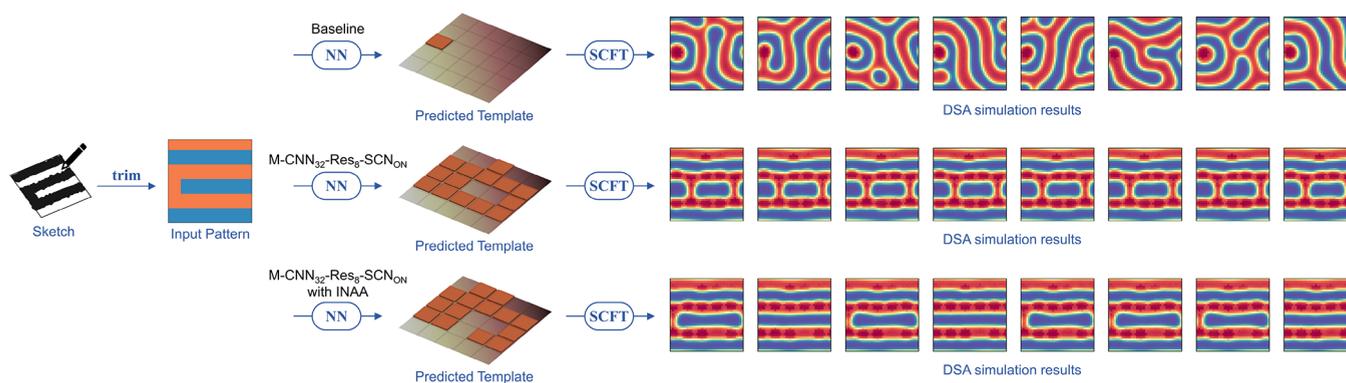


Figure 5. Illustrative example of human-designed pattern recognition. A pattern is hand-drawn and converted to a 40×40 binary array, which is then input to the trained NN. The NN predicts a 5×5 possible template. DSA simulation is then conducted by SCFT under different initial conditions. The multi-channel-input best model with INAA (third row) successfully guides the self-assembly of the BCP to form DSA patterns (highlighted by green boxes) similar to the design sketch, while the best model without INAA fails in this task. The baseline model (S-CNN₂-Res₀-SCN_{OFF}, first row) performs the worst, with simulated patterns that are completely different from the desired pattern. NN model names are described in the footnote of Table 2.

adequate for capturing most of the possible patterns that can be generated from a single template.

However, it is noteworthy that the EM accuracy drops to approximately 0.769 (also observed in the models with star symbols in Figure S6(d)) when all five input channels contain duplicates of a single pattern.

Human-Designed Pattern Recognition and Inverse Design. Although our best model achieves high EM accuracy (97.1%) for predicting the guiding templates based on the test DSA patterns obtained by SCFT simulations, in real-world inverse design, the input DSA patterns are normally designed by engineers rather than by SCFT. The phase interface of these human-designed (HD) patterns often differs from that of DSA patterns obtained by SCFT, which may pose challenges for the NN model trained solely on SCFT data. Therefore, a useful NN model for inverse design in lithography should first be evaluated by using HD patterns.

As shown in Figure 5, we begin by designing a DSA pattern either manually or using computer drawing tools, provided that the HD pattern has a higher resolution than the template (Figure S9). Next, this HD pattern is input to the NN model to obtain the predicted template (for the multi-channel NN model, five copies of the same HD pattern are used as input). Unlike when using SCFT-simulated patterns where the “true” templates are provided to evaluate the model, there is no “true” template in this task. Hence, experiments or SCFT simulations are carried out based on the predicted template to determine whether the microphase separation of block copolymers guided by the predicted template will indeed yield the desired DSA pattern. In this work, we employ SCFT simulations to generate several DSA patterns under different initial fields, guided by the predicted template; if one of these patterns is similar to the desired pattern, the inverse design task of this HD pattern is deemed successful.

Our findings indicate that our best model M-CNN₃₂-RES₈-SCN_{ON} with INAA (Bestinaa model) performs best in this task, successfully reproducing approximately half of the HD patterns, whereas only a few HD patterns could be reproduced by our best model without INAA, and the baseline model is the least effective, generating templates that differ significantly from those of the Bestinaa model (see Figures S7 and S8).

However, although the Bestinaa model can accomplish half of the inverse design tasks of HD patterns, it is still not entirely satisfactory. The unsatisfactory results can be attributed to two main reasons. First, the periodic boundary conditions used for generating the DSA patterns during the model training and SCFT simulations, while convenient, can limit the inverse design of HD patterns. Second, some HD patterns may be unsolvable for inverse design, implying that there may be no 5×5 template capable of inducing the desired pattern. We are currently uncertain whether the DSA patterns in Figure S8 failed due to this unsolvability.

Analysis of INAA's Performance. The improved robustness of the model after applying the noise operation INAA to the training data set is illustrated in Figure 6. In this example, the pattern of the sample gradually changes with a small number of pixels from the left to right. The model without noise operation produces different templates, whereas the model trained with INAA consistently outputs the correct template. This indicates that INAA can effectively enhance the model's ability to generalize to variations in the input data.

Here, we highlight the differential performance of INAA when predicting templates for SCFT-calculated patterns versus

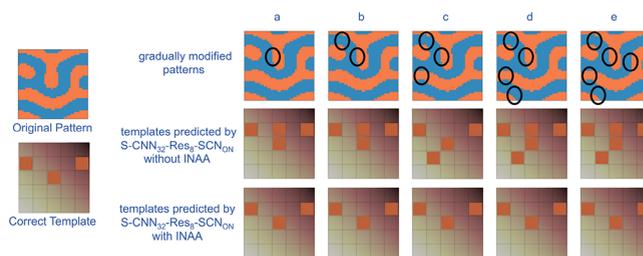


Figure 6. Effect of INAA on pattern recognition robustness. The right panel shows a sequence of gradually modified patterns (highlighted by black circles on the phase interface) and the corresponding templates predicted by the model S-CNN₃₂-RES₈-SCN_{ON} trained with and without INAA. The model trained with INAA (bottom row) consistently produces the correct template, while the model without INAA (center row) produces different templates as the pattern is modified. The original pattern and template from the data set are displayed on the left.

human-designed (HD) patterns. The “roughness” of the interfaces of the DSA pattern is inherently determined given a specific model setting. Consequently, it is not necessary to use INAA to promote the model's performance in predicting the template for the SCFT-obtained patterns. Interestingly, results in Table 2 indicate that the utilization of INAA may even slightly impair the model's performance in this case. In contrast, for human-designed patterns, the interface is ordinarily smoother than that of SCFT-obtained patterns. We should preclude the model from exploiting “roughness” information in its predictions in this case, which is exactly what INAA was designed to address.

It is worth noting that INAA is closely related to one of the problems in the real DSA, the line edge roughness (LER),^{24,53–56} since both involve changing the roughness of the interface. We can utilize the machine-learning methods developed in this work to extract the relationship between various factors, such as the pattern dimensions, the Flory–Huggins parameter, or the polydispersity index of BCP and LER, thereby identifying a suitable approach to minimize LER. Alternatively, reinforcement learning (RL) could also be harnessed to lessen LER.

CONCLUSIONS

This work undertakes the first effort to tackle the inverse design challenge of the guiding template for directed self-assembly (DSA) patterns using machine learning methods. By presenting the problem as a multi-label classification task, the study proves that templates can be predicted without the need for any forward simulations. The proposed techniques draw inspiration from image recognition but are tailored to predict phase-morphology-related problems, incorporating several data augmentation methods and model construction rules. A notable addition is the interface noise adding augmentation (INAA) method, which is extremely effective in pattern and template morphology prediction. Using these techniques and rules, the initially poor-performing baseline CNN model is gradually refined, resulting in the best model that incorporates the residual network and introspection mechanisms. The best model achieves an exact match accuracy improvement of nearly 40% up to 97.6%. Notably, even in the challenging and practical task of human-designed pattern recognition, the best model combined with INAA achieves a success rate of

approximately 50%, while the baseline model proves to be almost useless for this task.

However, there are still some limitations to this model. First, the periodic boundary conditions used in pattern generation restrict its applicability and extension. Second, the DSA pattern size used in this work is relatively small, and there are still too many defects in the calculated pattern (Figure S11). In our future work, we aim to explore alternative boundary conditions, increase the pattern size, focus on machine learning aided defect annihilation, and examine nonlamellar patterns.

■ ASSOCIATED CONTENT

SI Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acsami.3c05018>.

Flowchart of the multi-channel input strategy (MCIS), descriptions of NN structures for SCN and residual structure, descriptions of data augmentation methods for R&R, PB, WM, and INAA, schematic diagram of SCN, tables of NN architectures, schematic diagrams of PB and WM, accuracy curves of model performance on the validation data set, bar graphs of model performance on the test data set, exemplary and poor results of the inverse design of HD patterns, and schematic diagram of the necessity of inverse design (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Jianfeng Li – The State Key Laboratory of Molecular Engineering of Polymers, Department of Macromolecular Science, Fudan University, Shanghai 200433, China; orcid.org/0000-0002-8811-0030; Email: lijf@fudan.edu.cn

Yi-Xin Liu – The State Key Laboratory of Molecular Engineering of Polymers, Department of Macromolecular Science, Fudan University, Shanghai 200433, China; orcid.org/0000-0001-9374-5981; Email: lyx@fudan.edu.cn

Authors

Zhihan Liu – The State Key Laboratory of Molecular Engineering of Polymers, Department of Macromolecular Science, Fudan University, Shanghai 200433, China; orcid.org/0009-0001-5631-838X

Yuliang Yang – The State Key Laboratory of Molecular Engineering of Polymers, Department of Macromolecular Science, Fudan University, Shanghai 200433, China

Complete contact information is available at: <https://pubs.acs.org/10.1021/acsami.3c05018>

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant Nos. 21973018 and 21873021).

■ REFERENCES

- (1) Bates, F. S.; Fredrickson, G. H. Block Copolymers-Designer Soft Materials. *Phys. Today* **1999**, *52*, 32.
- (2) Matsen, M. W.; Schick, M. Microphase Separation in Starblock Copolymer Melts. *Macromolecules* **1994**, *27*, 6761–6767.
- (3) Tang, P.; Qiu, F.; Zhang, H. D.; Yang, Y. L. Phase Separation Patterns for Diblock Copolymers on Spherical Surfaces: A Finite Volume Method. *Phys. Rev. E* **2005**, *72*, 016710.
- (4) Xie, N.; Liu, M.; Deng, H.; Li, W.; Qiu, F.; Shi, A. Macromolecular Metallurgy of Binary Mesocrystals via Designed Multiblock Terpolymers. *J. Am. Chem. Soc.* **2014**, *136*, 2974.
- (5) Huang, H. J.; Liu, R. Z.; Ross, C. A.; Alexander-Katz, A. Self-Directed Self-Assembly of 3D Tailored Block Copolymer Nanostructures. *ACS Nano* **2020**, *14*, 15182–15192.
- (6) Hu, X.-H.; Xiong, S. Fabrication of Nanodevices Through Block Copolymer Self-Assembly. *Front. Nanotechnol.* **2022**, *4*, 762996.
- (7) Doerk, G. S.; Liu, C. C.; Cheng, J. Y.; Rettner, C. T.; Pitera, J. W.; Krupp, L. E.; Topuria, T.; Arellano, N.; Sanders, D. P. Pattern Placement Accuracy in Block Copolymer Directed Self-Assembly Based on Chemical Epitaxy. *ACS Nano* **2013**, *7*, 276–285.
- (8) Seino, Y.; Yonemitsu, H.; Sato, H.; Kanno, M.; Kato, H.; Kobayashi, K.; Kawanishi, A.; Azuma, T.; Muramatsu, M.; Nagahara, S. et al. Contact Hole Shrink Process Using Directed Self-Assembly. *Proc. Altern. Lithogr. Technol. IV* **2012**, 83230Y.
- (9) Yi, H.; Bao, X. Y.; Zhang, J.; Bencher, C.; Chang, L. W.; Chen, X. Y.; Tiberio, R.; Conway, J.; Dai, H. X.; Chen, Y. M.; Mitra, S.; Wong, H. S. P. Flexible Control of Block Copolymer Directed Self-Assembly Using Small, Topographical Templates: Potential Lithography Solution for Integrated Circuit Contact Hole Patterning. *Adv. Mater.* **2012**, *24*, 3107–3114.
- (10) Ruiz, R.; Dobisz, E.; Albrecht, T. R. Rectangular Patterns Using Block Copolymer Directed Assembly for High Bit Aspect Ratio Patterned Media. *ACS Nano* **2011**, *5*, 79–84.
- (11) Hinsberg, W.; Cheng, J.; Kim, C.; Sanders, D. P. Self-Assembling Materials for Lithographic Patterning: Overview, Status, and Moving Forward. *Proc. Altern. Lithogr. Technol. II* **2010**, 76370G.
- (12) Detcheverry, F. A.; Nealey, P. F.; de Pablo, J. D. Directed Assembly of a Cylinder-Forming Diblock Copolymer: Topographic and Chemical Patterns. *Macromolecules* **2010**, *43*, 6495–6504.
- (13) Ruiz, R.; Kang, H. M.; Detcheverry, F. A.; Dobisz, E.; Kercher, D. S.; Albrecht, T. R.; de Pablo, J. J.; Nealey, P. F. Density Multiplication and Improved Lithography by Directed Block Copolymer Assembly. *Science* **2008**, *321*, 936–939.
- (14) Darling, S. B. Directing the Self-Assembly of Block Copolymers. *Prog. Polym. Sci.* **2007**, *32*, 1152–1204.
- (15) Jeong, S. J.; Kim, J. Y.; Kim, B. H.; Moon, H. S.; Kim, S. O. Directed Self-Assembly of Block Copolymers for Next Generation Nanolithography. *Mater. Today* **2013**, *16*, 468–476.
- (16) Ouk Kim, S.; Solak, H. H.; Stoykovich, M. P.; Ferrier, N. J.; de Pablo, J. J.; Nealey, P. F. Epitaxial Self-Assembly of Block Copolymers on Lithographically Defined Nanopatterned Substrates. *Nature* **2003**, *424*, 411–414.
- (17) Tsai, H.; Pitera, J. W.; Miyazoe, H.; Bangsaruntip, S.; Engelmann, S. U.; Liu, C. C.; Cheng, J. Y.; Bucchignano, J. J.; Klaus, D. P.; Joseph, E. A.; Sanders, D. P.; Colburn, M. E.; Guillorn, M. A. Two-Dimensional Pattern Formation Using Graphoepitaxy of PS-*b*-PMMA Block Copolymers for Advanced FinFET Device and Circuit Fabrication. *ACS Nano* **2014**, *8*, 5227–32.
- (18) Chen, H. Y.; Brivio, S.; Chang, C. C.; Frascaroli, J.; Hou, T. H.; Hudec, B.; Liu, M.; Lv, H.; Molas, G.; Sohn, J.; Spiga, S.; Teja, V. M.; Vianello, E.; Wong, H. S. P. Resistive Random Access Memory (RRAM) Technology: From Material, Device, Selector, 3D Integration to Bottom-Up Fabrication. *J. Electroceram.* **2017**, *39*, 21–38.
- (19) Rasappa, S.; Schulte, L.; Ndoni, S.; Niemi, T. Directed Self-Assembly of a High-Chi Block Copolymer for the Fabrication of Optical Nanoresonators. *Nanoscale* **2018**, *10*, 18306–18314.
- (20) Zhang, L. S.; Liu, L. L.; Lin, J. P. Well-Ordered Self-Assembled Nanostructures of Block Copolymer Films via Synergistic Integration of Chemoepitaxy and Zone Annealing. *Phys. Chem. Chem. Phys.* **2018**, *20*, 498–508.
- (21) Paik, M. Y.; Bosworth, J. K.; Smilges, D. M.; Schwartz, E. L.; Andre, X.; Ober, C. K. Reversible Morphology Control in Block

- Copolymer Films via Solvent Vapor Processing: An in Situ GISAXS Study. *Macromolecules* **2010**, *43*, 4253–4260.
- (22) Raybin, J.; Ren, J.; Chen, X.; Gronheid, R.; Nealey, P. F.; Sibener, S. J. Real-Time Atomic Force Microscopy Imaging of Block Copolymer Directed Self Assembly. *Nano Lett.* **2017**, *17*, 7717–7723.
- (23) Li, W. H.; Nealey, P. F.; de Pablo, J. J.; Müller, M. Defect Removal in the Course of Directed Self-Assembly is Facilitated in the Vicinity of the Order-Disorder Transition. *Phys. Rev. Lett.* **2014**, *113*, 168301.
- (24) Kim, B.; Laachi, N.; Delaney, K. T.; Fredrickson, G. H. Directed Self-Assembly of Diblock Copolymers in Laterally Confining Channels: Line-Edge-Roughness and Defectivity. *Proc. Altern. Lithogr. Technol. VI* **2014**, 262–270.
- (25) Izumi, K.; Kim, B.; Laachi, N.; Delaney, K. T.; Carilli, M.; Fredrickson, G. H. Barriers to Defect Melting in Chemo-Epitaxial Directed Self-Assembly of Lamellar-Forming Diblock Copolymer/Homopolymer Blends. *Proc. Altern. Lithogr. Technol. VII* **2015**, 547–556.
- (26) Carpenter, C. L.; Delaney, K. T.; Fredrickson, G. H. Directed Self-Assembly of Diblock Copolymers in Multi-VIA Configurations: Effect of Chemopatterned Substrates on Defectivity. *Proc. Adv. Patterning Mater. Processes XXXIII* **2016**, 181–191.
- (27) Ginige, G.; Song, Y. D.; Olsen, B. C.; Luber, E. J.; Yavuz, C. T.; Buriak, J. M. Solvent Vapor Annealing, Defect Analysis, and Optimization of Self-Assembly of Block Copolymers Using Machine Learning Approaches. *ACS Appl. Mater. Interfaces* **2021**, *13*, 28639–28649.
- (28) Ginzburg, V. V.; Weinhold, J. D.; Trefonas, P. Computational Modeling of Block-Copolymer Directed Self-Assembly. *J. Polym. Sci. Pt. B-Polym. Phys.* **2015**, *53*, 90–95.
- (29) Yi, H.; Latypova, A.; Wong, H. S. P. Computational Simulation of Block Copolymer Directed Self-Assembly in Small Topographical Guiding Templates. *Proc. Altern. Lithogr. Technol. V* **2013**, 86801L.
- (30) Yi, H.; Bao, X. Y.; Tiberio, R.; Wong, H. S. P. Design Strategy of Small Topographical Guiding Templates for Sub-15 nm Integrated Circuits Contact Hole Patterns Using Block Copolymer Directed Self-Assembly. *Proc. Altern. Lithogr. Technol. V* **2013**, 868010.
- (31) Reilly, M.; Ginzburg, V.; Smith, M. D. Combining Physical Resist Modeling and Self-Consistent Field Theory for Pattern Simulation in Directed Self-Assembly. *Proc. Adv. Resist Mater. Processing Technol. XXX* **2013**, 86820G.
- (32) Mishra, V.; Fredrickson, G. H.; Kramer, E. J. Effect of Film Thickness and Domain Spacing on Defect Densities in Directed Self-Assembly of Cylindrical Morphology Block Copolymers. *ACS Nano* **2012**, *6*, 2629–2641.
- (33) Kriksin, Y. A.; Neratova, I. V.; Khalatur, P. G.; Khokhlov, A. R. Pattern Multiplication by Template-Guided Self-Assembly of Cylinder-Forming Copolymers: Field-Theoretic and Particle-Based Simulations. *Chem. Phys. Lett.* **2010**, *492*, 103–108.
- (34) Tu, K. H.; Huang, H.; Lee, S.; Lee, W.; Sun, Z. H.; Alexander-Katz, A.; Ross, C. A. Machine Learning Predictions of Block Copolymer Self-Assembly. *Adv. Mater.* **2020**, *32*, 2005713.
- (35) Tsai, C. L.; Delaney, K. T.; Fredrickson, G. H. Genetic Algorithm for Discovery of Globally Stable Phases in Block Copolymers. *Macromolecules* **2016**, *49*, 6558–6567.
- (36) Miskin, M. Z.; Khaira, G.; de Pablo, J. J.; Jaeger, H. M. Turning Statistical Physics Models into Materials Design Engines. *Proc. Natl. Acad. Sci. U. S. A.* **2016**, *113*, 34–39.
- (37) Arora, A.; Lin, T. S.; Rebello, N. J.; Av-Ron, S. H. M.; Mochigase, H.; Olsen, B. D. Random Forest Predictor for Diblock Copolymer Phase Behavior. *ACS Macro Lett.* **2021**, *10*, 1339–1345.
- (38) Vargo, E.; Dahl, J. C.; Evans, K. M.; Khan, T.; Alivisatos, P.; Xu, T. Using Machine Learning to Predict and Understand Complex Self-Assembly Behaviors of a Multicomponent Nanocomposite. *Adv. Mater.* **2022**, *34*, 2203168.
- (39) Bhattacharya, D.; Kleeblatt, D. C.; Statt, A.; Reinhart, W. F. Predicting Aggregate Morphology of Sequence-Defined Macromolecules with Recurrent Neural Networks. *Soft Matter* **2022**, *18*, 5037–5051.
- (40) Zhao, S. C.; Cai, T. Y.; Zhang, L. S.; Li, W. H.; Lin, J. P. Autonomous Construction of Phase Diagrams of Block Copolymers by Theory-Assisted Active Machine Learning. *ACS Macro Lett.* **2021**, *10*, 598–602.
- (41) Dong, Q. S.; Gong, X. R.; Yuan, K. R.; Jiang, Y.; Zhang, L. S.; Li, W. H. Inverse Design of Complex Block Copolymers for Exotic Self-Assembled Structures Based on Bayesian Optimization. *ACS Macro Lett.* **2023**, *12*, 401–407.
- (42) Sherman, Z. M.; Howard, M. P.; Lindquist, B. A.; Jadrlich, R. B.; Truskett, T. M. Inverse Methods for Design of Soft Materials. *J. Chem. Phys.* **2020**, *152*, 140902.
- (43) Khaira, G. S.; Qin, J.; Garner, G. P.; Xiong, S. S.; Wan, L.; Ruiz, R.; Jaeger, H. M.; Nealey, P. F.; de Pablo, J. J. Evolutionary Optimization of Directed Self-Assembly of Triblock Copolymers on Chemically Patterned Substrates. *ACS Macro Lett.* **2014**, *3*, 747–752.
- (44) Hannon, A. F.; Ding, Y.; Bai, W. B.; Ross, C. A.; Alexander-Katz, A. Optimizing Topographical Templates for Directed Self-Assembly of Block Copolymers via Inverse Design Simulations. *Nano Lett.* **2014**, *14*, 318–325.
- (45) Sorower, M. S. *A Literature Survey on Algorithms for Multi-Label Learning*; Oregon State University, Corvallis, OR, 2010.
- (46) Shim, S.; Cai, S.; Yang, J.; Yang, S.; Choi, B.; Shin, Y. Verification of Directed Self-Assembly (DSA) Guide Patterns Through Machine Learning. *Proc. Altern. Lithogr. Technol. VII* **2015**, 94231E.
- (47) Xiao, Z. G.; Du, Y. L.; Tian, H. T.; Wong, W. D. F.; Yi, H.; Wong, H. S. P.; Zhang, H. B. Directed Self-Assembly (DSA) Template Pattern Verification. *Proc. 2014 51st ACM/EDAC/IEEE Des. Autom. Conf.* **2014**, 1–6.
- (48) Xiao, Z. G.; Guo, D. F.; Wong, M. D. F.; Yi, H.; Tung, M. C.; Wong, H. S. P. Layout Optimization and Template Pattern Verification for Directed Self-Assembly (DSA). *Proc. 2015 52nd ACM/EDAC/IEEE Des. Autom. Conf.* **2016**, 1–6.
- (49) Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press, 2016.
- (50) He, K. M.; Zhang, X. Y.; Ren, S. Q.; Sun, J. Deep Residual Learning for Image Recognition. *2016 IEEE Conf. Comput. Vision Pattern Recognit.* **2016**, 770–778.
- (51) Li, J. F.; Zhang, H. D.; Chen, J. Z. Y. Structural Prediction and Inverse Design by a Strongly Correlated Neural Network. *Phys. Rev. Lett.* **2019**, *123*, 108002.
- (52) Wang, T. Y.; Li, J. F.; Zhang, H. D.; Chen, J. Z. Y. Designs to Improve Capability of Neural Networks to Make Structural Predictions. *Chin. J. Polym. Sci.* **2023**, DOI: 10.1007/s10118-023-2910-x.
- (53) Peters, A. J.; Lawson, R. A.; Ludovice, P. J. Effects of Block Copolymer Polydispersity and xN on Pattern Line Edge Roughness and Line Width Roughness from Directed Self-Assembly of Diblock Copolymers. *Proc. Altern. Lithogr. Technol. V* **2013**, 408–415.
- (54) Peters, A. J.; Lawson, R. A.; Nation, B. D.; Ludovice, P. J.; Henderson, C. L. Simulation Study of the Effect of Molar Mass Dispersion on Domain Interfacial Roughness in Lamellae Forming Block Copolymers for Directed Self-Assembly. *Nanotechnology* **2015**, *26*, 385301.
- (55) Suh, H. S.; Dudash, V.; Lorusso, G.; Mack, C. Roughness Study on Line and Space Patterning with Chemo-Epitaxy Directed Self-Assembly. *Proc. Adv. Patterning Mater. Processes XXXVII* **2020**, 146–154.
- (56) Lai, H.; Huang, G.; Tian, X.; Liu, Y.; Ji, S. Engineering the Domain Roughness of Block Copolymer in Directed Self-Assembly. *Polymer* **2022**, *249*, 124853.