

End-to-End Optimization for a Compact Optical Neural Network Based on Nanostructured 2×2 Optical Processors

Caiyue Zhao¹, Jiguang Wang¹, Simei Mao¹, Xuanyi Liu¹, Wai Kin Victor Chan¹, *Senior Member, IEEE*, and H. Y. Fu¹, *Senior Member, IEEE*

Abstract—Recent research in silicon photonic chips has made huge progress in optical computing owing to their high speed, small footprint, and low energy consumption. Here, we employ nanostructured 2×2 optical processors in an optical neural network for implementing a binary classification task efficiently. The proposed optical neural network is composed of five linear layers including ten optical processors in each layer, and nonlinear activation functions. 2×2 optical processors are designed based on digitized meta-structures which have an extremely compact footprint of $1.6 \times 4 \mu\text{m}^2$. A brand-new end-to-end design strategy based on Deep Q-Network is proposed to optimize the optical neural network for classifying a generated ring data set with better generalization, robustness, and operability. A high-efficient transfer matrix multiplication method is applied to simplify the calculation process in traditional optical software. Our numerical results illustrate that the maximum and mean accuracy on the testing data set can reach 90.5% and 87.8%, respectively. The demonstrated optical processors with a significantly compact area, and the efficient optimization method exhibit high potential for large-scale integration of whole-passive optical neural network on a photonic chip.

Index Terms—Digitized meta-structure, optical computing, optical neural networks, reinforcement learning.

I. INTRODUCTION

THE linear optical processor in optical neural networks (ONNs) has attracted tremendous research interest, due to their low energy consumption and high calculation speed [1]. High-dimensional linear optical processors configured to implement matrix multiplication in ONNs can be realized by cascading 2×2 basic building blocks [2]. Traditionally, reconfigurable Mach-Zehnder interferometer (MZI)-based linear optical processors have been demonstrated to perform matrices multiplications in neural networks [3]. However, this architecture suffers phase noise and large footprint. An on-chip

diffractive ONN was proposed with simple structure, and lower power consumption [4], but this architecture suffers large footprint over $1000 \mu\text{m}$. Another integrated ONN based on optical scattering units (OSUs) was proposed [5]. Nevertheless, the OSU composed of a 9×9 multimode interference (MMI) coupler is optimized by the adjoint method with a high degree of freedom, resulting in high designing difficulty and a complicated fabrication process.

The digitized meta-structures have emerged as an ideal solution for constituting ONNs due to their ultra-compact size and controllable minimum feature size. The footprint of a single digitized meta-structure can be reduced to a few microns, making it highly suitable for large-scale integration. A range of functional components based on compact digitized meta-structures has been designed [6], [7], [8], [9]. Digitized meta-structures not only can be designed to implement a specific function but have been proved to serve as an optical processor to achieve the bar state, cross state, and unitary transmission [10]. 2×2 silicon photonic components have been arranged in waveguide circuit architecture to realize diverse computational functionalities [11], demonstrating the potential in computing and communications applications.

Digitized meta-structures could be designed by utilizing numerous algorithms, such as genetic algorithm (GA), and direct binary search (DBS) [12], [13]. However, these algorithms relying on hundreds of iterations of electromagnetic simulations to obtain the optimal results are time-consuming. Especially when multiple digitized meta-structures with different functions are integrated to achieve large-scale matrix multiplications, these methods become extremely laborious. Besides, a deep neural network (DNN) can be applied to the design of nanophotonic devices due to its capability to learn the complex relationship between optical responses and device geometries. Nevertheless, the training of neural networks tends to be accompanied by issues such as taking a long time, demanding a large data set, high training difficulty, and overfitting [10], [14].

Traditional device design methods are not always optimized for system performance. As for the systematic training method of ONNs to perform a specific task, Hughes et al. proposed in situ back propagation and gradient measurement [15]. They used this approach to numerically demonstrate the implementation of a logical XOR gate. However, due to the discrete decision of the optimization process caused by the meta-structures, it is not

Manuscript received 23 April 2023; revised 17 July 2023; accepted 26 August 2023. Date of publication 29 August 2023; date of current version 8 September 2023. This work was supported by Guangdong Basic and Applied Basic Research Foundation under Grant 2021A1515011450. (Corresponding author: H. Y. Fu.)

The authors are with the Tsinghua-Berkeley Shenzhen Institute and Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China (e-mail: zhaocy20@mails.tsinghua.edu.cn; wjg20@mails.tsinghua.edu.cn; maosm19@mails.tsinghua.edu.cn; liuxuanyi20@mails.tsinghua.edu.cn; chanw@sz.tsinghua.edu.cn; hyfu@sz.tsinghua.edu.cn).

Digital Object Identifier 10.1109/JPHOT.2023.3309835

possible to optimize the devices through an explicit functional relation. In traditional optimization schemes, the first step is to obtain the transfer matrix corresponding to the device through weight matrix decomposition [2], and in the second step, the corresponding device can be optimized from the transfer matrix [10]. However, the inverse optimization from the optical response to the physical structure is hard to get the target solution [14], [16], and the error accumulation is not linear. Therefore, the overall error becomes uncontrollable for large-scale integrated digitized meta-structures mesh. To overcome these challenges, we merge this step with the training of ONNs by using an end-to-end bi-level optimization algorithm based on Deep Q-Network (DQN), so that the generated devices can be quickly selected through simulation to determine if they are suitable for the neural network.

Compact 2×2 digitized meta-structures are cascaded for constructing a high-dimensional optical processor to perform the linear operations in the neural network. In this training of an ONN, the transfer matrices, including transmission coefficients of the meta-structures in each layer, are optimized so that the light can propagate to the desired output ports finally. Once this training assisted by a computer is accomplished, the passive layers can be fabricated physically and combined with activation functions together to form an optical network with low power consumption. Besides, we employ a high-efficient matrix multiplication method to simplify the calculation process in traditional optical software INTERCONNECT from Lumerical, Inc. [17]. In particular, we divide the designed algorithm into two levels: the upper level generates meta-structure candidates, while the lower level selects optimal meta-structures and determines their positions in the ONN corresponding to the order of matrix multiplication. The numerical results illustrate that the classification accuracy on the training data set is 90.0%. The mean and standard deviation of test data set accuracy are 87.8% and 0.0141, respectively. The maximum accuracy on the testing data set is 90.5%. We believe our proposed ultra-compact optical processors via digitized meta-structures, and the corresponding optimization algorithm demonstrate a promising candidate for future large-scale on-chip ONNs.

II. DEMONSTRATION OF THE PROPOSED ONN

A. Structure of 2×2 Optical Processors

As shown in Fig. 1(a), nanostructured 2×2 optical processors can process the optical signal from input ports to output ports as the transfer matrix form

$$T = \begin{bmatrix} a_{13} + jb_{13} & a_{23} + jb_{23} \\ a_{14} + jb_{14} & a_{24} + jb_{24} \end{bmatrix}, \quad (1)$$

where $a_{ij}, b_{ij} (i, j \in \{1, 4\})$ are the real parts and imaginary parts, respectively, representing transmission coefficients from port i to port j . Assuming that the input signals are E_{I1} and E_{I2} , and the output signals are E_{O1} and E_{O2} , the output signal can be obtained by the interaction between the input signal and the

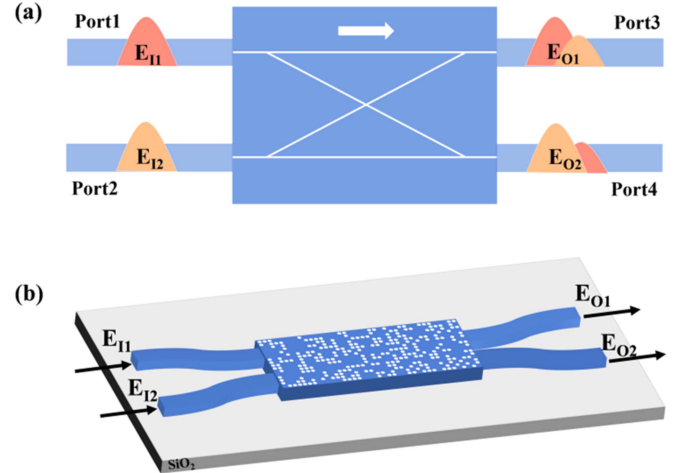


Fig. 1. (a) Schematic diagram of the proposed optical processor, (b) the architecture of the nanostructured 2×2 optical processor.

transfer matrix as

$$\begin{bmatrix} E_{O1} \\ E_{O2} \end{bmatrix} = T \cdot \begin{bmatrix} E_{I1} \\ E_{I2} \end{bmatrix} = \begin{bmatrix} a_{13} + jb_{13} & a_{23} + jb_{23} \\ a_{14} + jb_{14} & a_{24} + jb_{24} \end{bmatrix} \cdot \begin{bmatrix} E_{I1} \\ E_{I2} \end{bmatrix}. \quad (2)$$

The accurate transfer matrix is calculated by S-parameter sweep in 3D Finite-Difference Time-Domain (FDTD) software from Lumerical Solutions, Inc. As shown in Fig. 1(b), we propose a compact optical processor on a standard silicon-on-insulator (SOI) platform with a 220-nm-thick silicon and a silica cladding layer for protection [10]. The proposed optical processor is composed of two input ports, two output ports, and a middle digitized meta-structure formed by a set of nanoholes with a specific distribution. Each port consists of an s-bend waveguide with a width of 400 nm. The gap between the two waveguides at the input port is set to be $1.6 \mu\text{m}$ to avoid light coupling. The middle region with a $1.6 \times 4 \mu\text{m}^2$ area is discretized into 16×40 pixels of circular holes with a diameter of 60 nm. The material property for each circular hole can be either silicon or silica. These etched holes with a large refractive index contrast diffract the input light on subwavelength scale. The distance between the adjacent holes is chosen to be 100 nm. And the nanohole distribution can be determined by the optimization algorithm for a specific application.

B. Architecture of the ONN

In most cases, an artificial neural network (ANN) maps an input vector to an output vector via an alternate sequence of linear operations and nonlinear functions. In our scheme, Fig. 2(a) exhibits the framework of our proposed ONN, which is composed of linear operations and nonlinear functions. The ONN consists of an input layer, five hidden layers, and an output layer. A single hidden layer implementing a 5×5 weight matrix is marked in blue. Specifically, there are five linear layers in the ONN, including ten meta-structures in each linear layer. We use the Clement structure to build linear units of the ONN composed of meta-structures to realize the weight matrix due

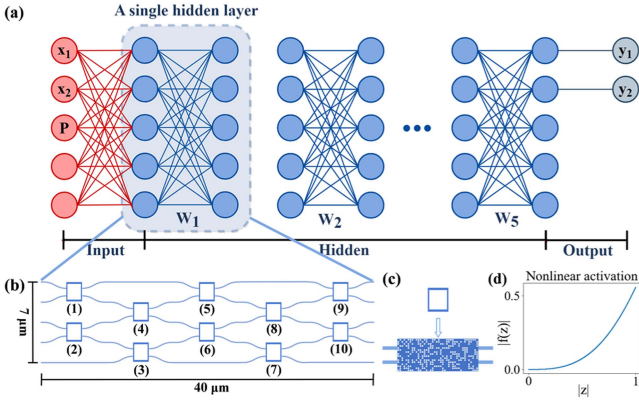


Fig. 2. (a) The framework of neural network. (b) A 5×5 optical processor including ten meta-structures for implementing the weight matrix. (c) Optical processors composed of meta-structures. (d) Nonlinear activation function.

to multiport interferometers based on the clement structure possessing several advantages in terms of better stability, shallower depth, and more compact area [18]. The footprint of one layer is $40 \times 7 \mu\text{m}^2$.

Randomly generated meta-structures are simulated by FDTD to calculate their S-matrices which are imported into INTERCONNECT software for the construction of the linear layers of the proposed ONN. Fig. 2(b) depicts a 5×5 processor containing the meta-structures labeled (1) to (10) to construct the linear transformation matrix $[W]_{5 \times 5}$. Fig. 2(c) exhibits the 2×2 optical processor composed of a meta-structure. The designed structure implements a linear transformation between the input matrix $[I]_{1 \times 5}$ and the output matrix $[O]_{1 \times 5}$ based on optical wave interactions in physical devices [19]. The 5×5 optical processor serving as the weight matrix can be calculated by the product of the transfer matrices of its constituent 2×2 optical processors as

$$[W]_{5 \times 5} = [T^{(n)}]_{5 \times 5} \cdot [T^{(n-1)}]_{5 \times 5} \cdot [T^{(n-2)}]_{5 \times 5} \cdots [T^{(2)}]_{5 \times 5} \cdot [T^{(1)}]_{5 \times 5}, \quad (3)$$

where n is the number of meta-structures in one linear layer. Q -mode multiport (shown here for $Q = 5$) can be implemented using a mesh of $Q(Q-1)/2$ 2×2 optical processors [18]. Here, n is determined to be 10.

Apart from the linear operation in ONNs, activation functions improve the performance of ONNs significantly by enabling them to learn a more complicated mapping from input to output. The $f(z)$ function plays an important role in the nonlinear activation of neurons. Researchers propose nonlinear functions which can be realized by using nonlinear effects in optics, such as saturation absorption of monolayer graphene, the nonlinear realization of electro-optical hybrid elements, and optical bistable and two-photon absorption characteristics [20], [21], [22], [23]. Here, we introduce an electro-optical activation function, which has been demonstrated experimentally using a SiN waveguide technology [24], [25]. The nonlinear response can be achieved by converting a fraction of the optical input into the electrical signal, and then modulating the intensity of the

remaining portion in the original optical signal. The activation function is defined as

$$f(z) = j\sqrt{1-\alpha} \cdot \exp\left(-j\left[\frac{g|z|^2}{2} + \frac{\phi}{2}\right]\right) \cdot \cos\left(\frac{g|z|^2}{2} + \frac{\phi}{2}\right)z, \quad (4)$$

where z is the amplitude of the input signal. g is the phase gain parameter, determined to be 0.4π [24]. α can control the portion of the input converted to the electrical signal. Here, α is assumed to be 0.1. ϕ is one of the key parameters in the activation function. Here, the parameter ϕ is equal to π . In this case, the activation function output amplitude as a function of input signal amplitude exhibits the ReLU-like response, as shown in Fig. 2(d). In addition, the striking advantage is that the activation function can be programmed to produce different types of nonlinear responses by tuning the electrical bias. A photoelectric hybrid neural network is built by using Python script to assist INTERCONNECT simulation software. The linear matrix operation is implemented in the optical domain by meta-structures, and the nonlinear activation function is implemented in the electrical domain.

III. END-TO-END OPTIMIZATION USING REINFORCEMENT LEARNING

A. Algorithm Design

We explore the matrix multiplication method to substitute for INTERCONNECT, which accelerates the optimization process extremely. The feasibility of this replacement can be demonstrated through the Pearson experiment analysis in Appendix A. The proposed optimization algorithm is based on a bi-level framework, as illustrated in Fig. 3(a). The proposed optimization algorithm for the five-layer ONN is based on the meta-structures which can implement a unitary weight matrix. The upper-level diagram depicts the process of meta-structure generation, while the lower-level diagram shows the steps involved in selecting and positioning the meta-structures for the ONN. The upper level provides meta-structure candidates pool to the lower level, and the lower level provides performance feedback to the upper level by calculating matrix multiplication. In addition, transfer matrices of the meta-structures for matrix calculation are generated in FDTD. The performance of the ONN for classification tends to be poor initially before properly adjusting the distribution of nanoholes in the meta-structures. These two levels work together to achieve the final optimization task through continuous iteration learning. More specifically, the input of the algorithm is N meta-structures, and the output of the algorithm is the optimized meta-structures and their positions in the linear layer of the ONN. The positions of the meta-structures are crucial as they correspond to the order of matrix multiplication, and different positions result in different transfer matrices for each layer.

DQN uses a deep neural network to approximate state behavior value function Q [26]. Q , denoted as $Q(s, a)$ represents the expectation of the reward that acting a in the state s can get. In

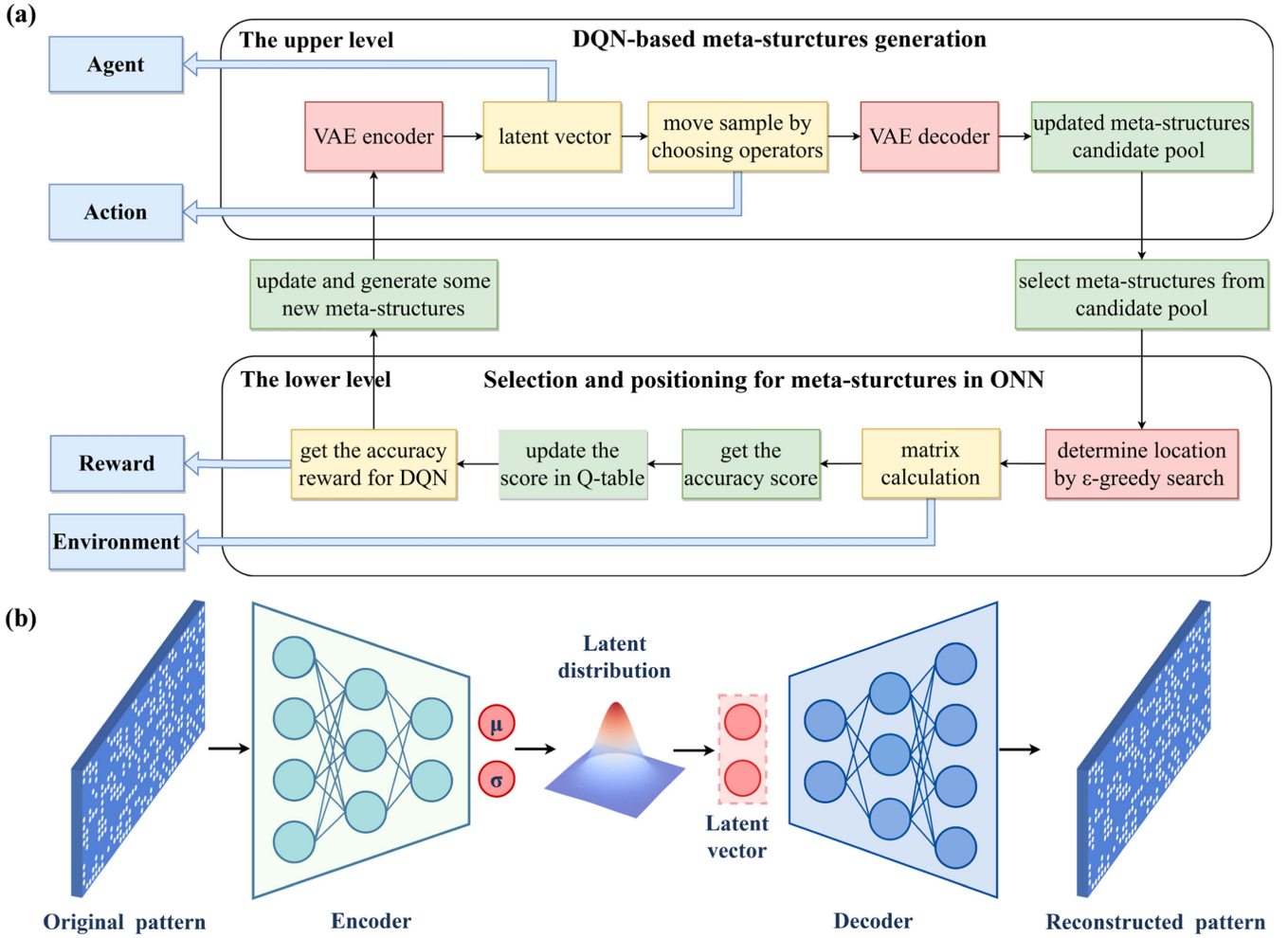


Fig. 3. (a) The framework diagram of the proposed bi-level deep Q-network. (b) Illustration of the variational autoencoder.

addition, the environment will feedback corresponding reward R according to the agent's action, so the action with the greatest reward can be determined based on the constantly updated Q value. The reward can be calculated by

$$R = 100 \times \left(\frac{\text{True predicted samples}}{\text{Total samples}} - 0.5 \right). \quad (5)$$

Here, the encoded meta-structure serves as the agent, and the state is defined as the value of an encoded meta-structure, which represents the location of this sample in the low dimensional space. The action is defined as the unidirectional movement in the low dimensional space for the encoded sample, which is specifically manifested as the increase or decrease in a dimension of the low dimensional vector. After adjustment, the meta-structure is restored through the decoder. The reward is defined as the best accuracy results of the meta-structures in the lower-level matrix multiplication calculation. The matrix calculation in the lower level serves as the environment of DQN.

Due to the complex structure of meta-structures, variational autoencoder (VAE) is introduced for dimensionality reduction of meta-structures to define the state of reinforcement learning, and the framework of VAE as shown in Fig. 3(b). VAE uses an encoder to transform the high dimensional input data into the

latent vector and a decoder to reconstruct the original input data from the latent low dimensional vector. The high dimensional data represented by meta-structures is mapped into a Gaussian distribution over the latent space, described by a mean vector and a standard deviation vector. The low-dimensional latent vector can be sampled from the latent space for the decoder. VAE is trained by two loss functions: one is the reconstruction loss that forces the decoded sample to match the initial input. The other is regularization loss, described by Kullback-Leibler (KL) divergence, which helps to learn latent space with good structure and diminish overfitting in training data [27]. It is worth noting that VAE in our proposed algorithm needs to be trained using a heap of randomly generated meta-structures with a normal distribution in advance.

In the upper level, the value of the encoded latent vector can be defined as the state of DQN. The sample is moved in a specific direction by selecting operators using DQN. After the operators' work, the moved latent vector can be decoded into the reconstructed meta-structures by the decoder. These meta-structures constitute the candidate pool for the lower level. The lower level, as the simulator of the upper level, employs an ϵ -greedy algorithm based on the Q-table to select meta-structures from the candidates' pool and change their positions, which

TABLE I
THE ALGORITHM HYPER-PARAMETERS

Hyperparameters	N	M	I	ϵ	β	γ
Value	50	20	5000	0.5	0.5	0.9

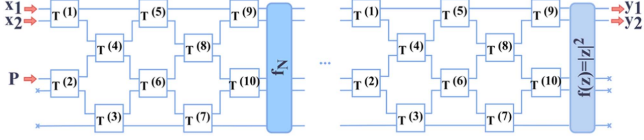


Fig. 4. Numerical demonstration of ONN to classify a ring.

means that randomly arranging new meta-structures for each position when the probability is larger than ϵ , otherwise selecting the meta-structures with the best performance at that location previously. And Q -table is used to record the best performance of each location for each meta-structure. With the increase in iterations, the number of generated new meta-structures ($N-M$) in the upper-level decreases, gradually screening the meta-structures with better prediction results. When iteration I in the lower level reaches 5000, the overall iteration will stop. The pseudocode for our proposed reinforcement learning-based bi-level adaptively optimization algorithm is shown in Appendix B algorithm 1.

B. Algorithm Parameters

The training of VAE involves two neural networks, the encoder and the decoder. Specifically, the encoder consists of an input layer with 640 units, four fully connected hidden layers with 1024, 128, 64, and 32 units, and an output layer with 4 units. 640 units in the input layer correspond to discretized 16×40 pixels of circular holes, while 4 units in the output layer represent a specific coordinate in four-dimensional solution space. The activation functions of every middle layer are ReLU functions, while the final layer is followed by a Sigmoid activation function. In addition, the decoder network has the exact opposite structure. DQN contains an input layer with 4 units, three entirely hidden connected layers with 48, 24 and 12 units in each layer, and an output layer with 8 units, which means expected values for 8 different moving operators in 4 bidirectional dimensions. These hidden layers are followed by the ReLU functions. The key hyper-parameters in the algorithm are shown in Table I.

IV. VERIFICATION OF THE DESIGNED ONN

A. Training Demonstration

As shown in Fig. 4, the ONN involves five linear layers constructed by meta-structures, followed by the nonlinear activation function except the last linear layer. The nonlinear function f_N is exhibited in (4). A final $|z|^2$ function is applied at the end to measure output power. The size of the network (number of waveguides) is set to five. We generate four hundred samples for training and testing. These samples can be represented by mapping from input to output ($X_0 \rightarrow O$). Here, $X_0 = [x_1, x_2, P, 0, 0]^T$ where x_1 and x_2 are input powers independently, set

TABLE II
CLASSIFICATION ACCURACIES ON DIFFERENT TESTING DATA SETS

	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
Accuracy	87.0%	88.0%	89.0%	89.5%	88.0%
	Data set 6	Data set 7	Data set 8	Data set 9	Data set 10
Accuracy	87.0%	87.5%	86.0%	90.5%	85.5%

to be real values for simplicity. While P is related to x_1 and x_2 , calculated by $P = \sqrt{p_0 - x_1^2 - x_2^2}$. The function of P is to normalize input data to have the same total power input by injecting extra power into the third input port. Specifically, p_0 is the total power injected with each data input. Here, p_0 is chosen to be 40. Each training sample has its corresponding label, y_{pre} , which is encoded into the output O , as $[1, 0, 0, 0, 0]^T$ and $[0, 1, 0, 0, 0]^T$ for $y_{pre} = 0$ and $y_{pre} = 1$, respectively.

The following formula involving noise rate n and random variable v is utilized to label each point in a generated data set. The noise rate is set to be 0.05, added to disturb data distribution and increase the robustness of the model. Variables v are randomly generated following a standard normal distribution. We set label $y = 0$ if

$$0.4 \leq \sqrt{x_1^2 + x_2^2} + n \times v \leq 0.8. \quad (6)$$

Otherwise, $y = 1$. The underlying distribution for the data set resembles a ring centered at (2,2).

B. Simulation Results and Discussion

We randomly generated 10 ring data sets and selected 200 sample points in each data set for testing. Table II exhibits that the classification accuracy of the proposed ONN on these data sets. The mean and standard deviation of test data set accuracy can be calculated as 87.8% and 0.0141, respectively. The maximum accuracy on the testing data set is 90.5%. The confusion matrices for the selected 200 samples on the training and testing datasets are depicted in Fig. 5(a) and (b), respectively. Fig. 5(c) exhibits the accuracy curve as a function of training iterations. The individual accuracy for predicting 0 and 1 on the training data set is 89% and 90.5%, respectively, demonstrating the equivalent prediction capacity for both labels. We define a generation of fifty meta-structures in the upper level as one iteration for convenience. By conducting 3D FDTD simulation, the transfer matrix of one meta-structure can be obtained in 115 s. After 71 iterations, the classification accuracy achieves 90%. The classification effect can be demonstrated in Fig. 6(a) and (b). Here we use the sample points of testing data set 9 to demonstrate the classification results. Circles represent the correct prediction described as $y_{pre} = y$, while crosses correspond to the false prediction. In addition, the label y corresponds to the color of the dots. For example, red crosses mean that $y = 0$, $y_{pre} = 1$, causing the false prediction. The background color exhibits the generated ring data set without added noise. It can be noticed that most of the samples in the data set can be classified correctly,

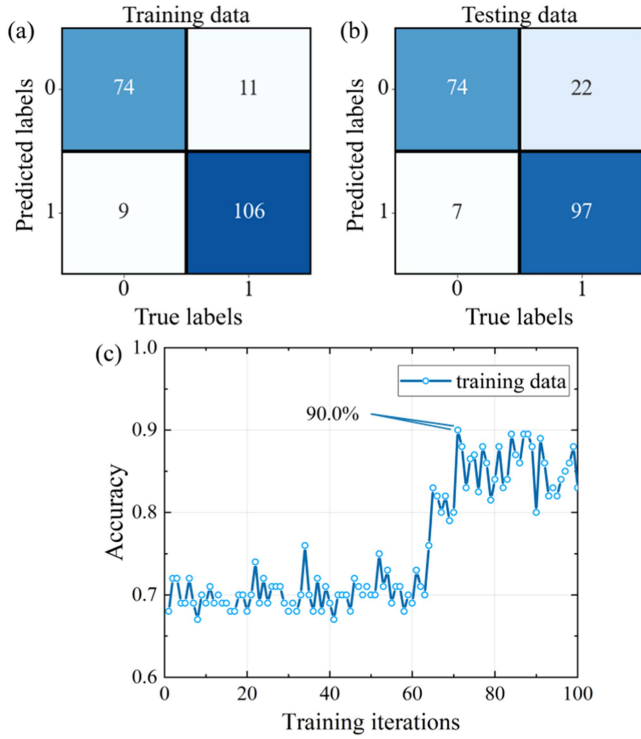


Fig. 5. Confusion matrices for the 200 samples from the generated ring data set (a) on the training data set, and (b) in the testing data set. (c) The training process. The highest accuracy obtained by the optimum parameters is 90.0% on the training data set.

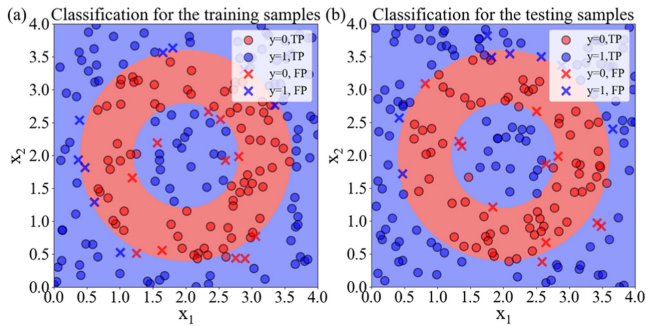


Fig. 6. (a) The classification effect for the training samples. Red and blue dots correspond to labels $y = 0$ and $y = 1$ on a given x_1 and x_2 input. Circles indicate true prediction while crosses represent false prediction, (b) the classification effect for the testing samples.

except for a few at the boundary, demonstrating successful prediction without overfitting.

Fig. 7(a) illustrates the effect of nanohole diameter on the classification accuracy of the generated ring data set, where the optimal diameters for achieving the highest accuracy are 60 nm and 58 nm. Besides, our proposed algorithm to optimize the ONN is insensitive to the diameters. Accuracy can still be maintained larger than 80% even under the diameter variation of 10 nm. Fig. 7(b) exhibits the accuracy of the proposed ONN- m architecture on the generated ring data set, where ONN- m refers to the ONN architecture including m linear layers. In the training process, increasing linear layers composed of digitized meta-structures results in a more complicated ONN with more

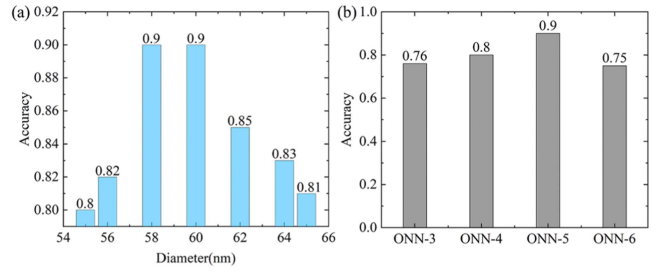


Fig. 7. Variation in the accuracy of the ONN with respect to (a) the diameter of nanoholes, and (b) the different linear layers. ONN- m means that the ONN architecture contains m linear layers, here $m = 3, 4, 5, 6$, respectively.

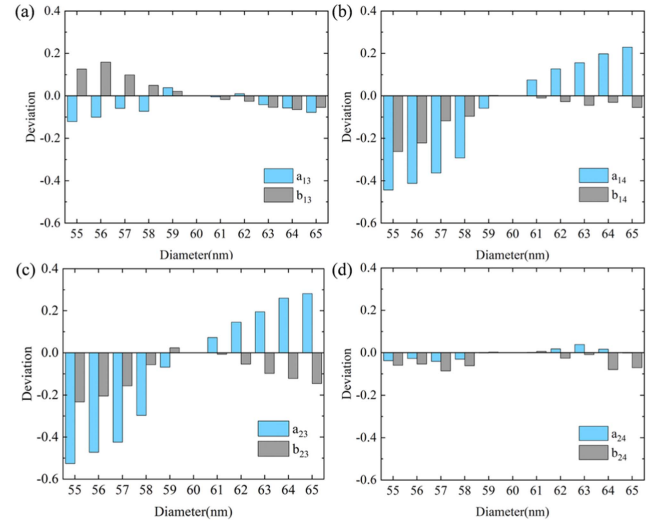


Fig. 8. Deviation of transmission coefficients (a) a_{13} and b_{13} , (b) a_{14} and b_{14} , (c) a_{23} and b_{23} , (d) a_{24} and b_{24} , in transfer matrix with respect to the diameter of nanoholes.

parameters to optimize, leading to better fitting results. However, increasing layers of the neural network continuously bring some negative problems, overfitting and degradation [28]. Moreover, there is a tradeoff between the performance and abundant time cost led by sophisticated ONNs. As depicted in Fig. 7(b), the prediction performance of the ONN with five layers exceeds other architectures. Fig. 8 exhibits the transmission coefficients deviation for single meta-structure with varying diameters of nanoholes, where the deviation is defined as the transmission coefficient difference between other diameters and the central diameter. The variation of diameters has a larger influence on the transmission coefficients relevant to cross ports (e.g., port 1 to port 4, and port 2 to port 3).

V. CONCLUSION

In summary, we propose a whole-passive ONN architecture based on the SOI platform. We utilize digitized meta-structures with high compactness acting as 2×2 optical processors to construct a five-layer ONN for a ring classification task. A brand-new end-to-end design strategy compatible with digitized meta-structures is proposed to directly optimize the ONN with high efficiency and accuracy. This method exhibits superior advantages over traditional complex optimization for the digitized

TABLE III
COMPARISON OF TIME CONSUMPTION FOR THE SIMULATION AND MATRIX
CALCULATION

	INTERCONNECT	Matrix calculation
Time(s)	150.2	0.0011

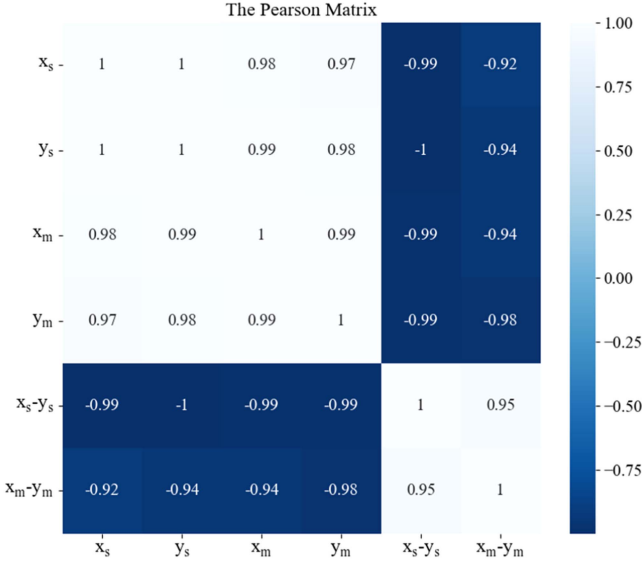


Fig. 9. The heatmap about the correlation coefficients between the simulation and matrix multiplication results.

meta-structure itself. The numerical results illustrate that prediction accuracy in a generated ring data set can reach 90.0% on the training data sets, while the maximum accuracy on the testing data set can reach 90.5%. The integrated 2×2 nanostructured optical processor with a compact footprint can be embedded within a computer architecture as an accelerator to implement matrix multiplications. We believe these findings may unleash the potential of high-dimensional passive optical processors in on-chip ONNs.

APPENDIX A PEARSON CORRELATION ANALYSIS

In fact, INTERCONNECT is a transfer matrix solver fundamentally. However, a large set of extra features make the simulation time-consuming, which is not beneficial to optimizing the performance of the whole ONN. Optical transfer matrix multiplication provides a competitive alternative approach for obtaining results, and more importantly, it is numerically more efficient than simulation. The time of one sample consumed from input to output by simulation and matrix calculation is compared in Table III.

The calculation and simulation are implemented using Python and INTERCONNECT in 2022 R1.1 version on a 2.20 GHz Intel Xeon E5-2650 v4 PC with 32 GB RAM. The feasibility of this alternative can be verified through the Pearson correlation analysis, which produces a score that can vary from -1 to $+1$ [29]. The Pearson score is $+1$ signifying a perfect positive relationship. Two uncorrelated objects would produce a Pearson

Algorithm 1: Reinforcement Learning-Based Bi-Level Adaptively Optimization.

Input: a randomly generated meta-structures set including N structures $Q = \{q_i\}, \forall i \in N$

Output: Optimized meta-structures and final locations in ONN

Initialization parameter

Randomly generated location sequence \vec{s}

Get the accuracy reward R for each meta-structure using matrix calculation (Q, \vec{s})

Update the score r_{is} for each meta-structure in this location using r

While upper-level stopping condition is met:

Update each meta-structure highest reward

$r_i \leftarrow \max(r_{is})$

Update M and the DQN learning rate

Retain M structures that have top rewards as Q'

For meta-structure q_i **in** meta-structures set Q :

If meta-structure q_i **not in** Q'

$\vec{\omega}' =$ Feature dimension reduction using VAE encoder (meta-structure)

$\vec{\omega}' =$ changing the encoded meta-structure $\vec{\omega}$ using Deep Q-network ($\vec{\omega}$)

$q'_i =$ Decode to get a new meta-structure using VAE decoder ($\vec{\omega}'$)

Update the structures set Q , remove q_i from Q , add $q'_i \rightarrow Q$

End if

End for

While lower-level stopping condition is met:

If random number $u < \epsilon$:

update sequence \vec{s}' as randomly generated

Else

Choose the meta-structures with the highest score at this location

End if

Get the accuracy reward r for each

meta-structure using matrix calculation (Q, \vec{s})

Update the score r_{is} for each meta-structure in this location using r

$r_{is} \leftarrow \alpha \cdot r_{is} + (1 - \alpha) \cdot r$

If $r > R$:

Update the best structures set Q and the best sequence \vec{s}

Update the best accuracy reward $R \leftarrow r$

End if

End while

End while

Return Optimized meta-structures and final location sequence in ONN

score near zero. The Pearson correlation analysis is performed to verify the linear relationship between the matrix multiplications and simulations. The heatmap of the Pearson correlation coefficients between the simulation and matrix multiplication results are exhibited in Fig. 9. x_s and y_s represent the power at the first

output port and the second output port in the simulation, respectively. Meanwhile, x_m and y_m indicate the output powers at the first two ports calculated by matrix multiplications, respectively. It can be discovered the correlation coefficient between the x_s - y_s and x_m - y_m is 0.95, revealing the significantly high linear correlation. Consequently, we can employ the method of matrix multiplication to replace the simulation in INTERCONNECT, accelerating the process of solving the optimal results.

APPENDIX B THE ALGORITHM DESIGN

See Algorithm 1.

ACKNOWLEDGMENT

The authors would like to express sincere thanks to Zeyad A. H. Qasem for helpful discussions.

REFERENCES

- [1] H. Zhang et al., "Efficient on-chip training of optical neural networks using genetic algorithm," *ACS Photon.*, vol. 8, no. 6, pp. 1662–1672, 2021.
- [2] F. Shokraneh, M. S. Nezami, and O. Liboiron-Ladouceur, "Theoretical and experimental analysis of a 4×4 reconfigurable MZI-based linear optical processor," *J. Lightw. Technol.*, vol. 38, no. 6, pp. 1258–1267, Mar. 2020.
- [3] Y. Shen et al., "Deep learning with coherent nanophotonic circuits," *Nature Photon.*, vol. 11, no. 7, pp. 441–446, Jun. 2017.
- [4] T. Fu et al., "On-chip photonic diffractive optical neural network based on a spatial domain electromagnetic propagation model," *Opt. Exp.*, vol. 29, no. 20, pp. 31924–31940, Sep. 2021.
- [5] Y. Qu et al., "Inverse design of an integrated-nanophotonics optical neural network," *Sci. Bull.*, vol. 65, no. 14, pp. 1177–1183, 2020.
- [6] Y. Liu et al., "Four-mode waveguide crossing via digitized meta-structure," in *Proc. IEEE Opt. Fiber Commun. Conf. Exhib.*, 2021, pp. 1–3.
- [7] H. Xie et al., "An ultra-compact 3-dB power splitter for three modes based on pixelated meta-structure," *IEEE Photon. Technol. Lett.*, vol. 32, no. 6, pp. 341–344, Mar. 2020.
- [8] A. Majumder, B. Shen, R. Polson, and R. Menon, "Ultra-compact polarization rotation in integrated silicon photonics using digital metamaterials," *Opt. Exp.*, vol. 25, no. 17, pp. 19721–19731, Aug. 2017.
- [9] B. Shen, P. Wang, R. Polson, and R. Menon, "An integrated-nanophotonics polarization beamsplitter with $2.4 \times 2.4 \mu\text{m}^2$ footprint," *Nature Photon.*, vol. 9, no. 6, pp. 378–382, May 2015.
- [10] S. Mao, L. Cheng, F. N. Khan, Z. Geng, Q. Li, and H. Y. Fu, "Inverse design of high-dimensional nanostructured 2×2 optical processors based on deep convolutional neural networks," *J. Lightw. Technol.*, vol. 40, no. 9, pp. 2926–2932, May 2022.
- [11] D. Gostimirovic and R. Soref, "An integrated optical circuit architecture for inverse-designed silicon photonic components," *Sensors*, vol. 23, no. 2, 2023, Art. no. 626.
- [12] S. Mao, L. Cheng, C. Zhao, F. N. Khan, Q. Li, and H. Y. Fu, "Inverse design for silicon photonics: From iterative optimization algorithms to deep neural networks," *Appl. Sci.*, vol. 11, no. 9, 2021, Art. no. 3822.
- [13] S. Molesky, Z. Lin, A. Y. Piggott, W. Jin, J. Vucković, and A. W. Rodriguez, "Inverse design in nanophotonics," *Nature Photon.*, vol. 12, no. 11, pp. 659–670, Oct. 2018.
- [14] R. Singh, A. Agarwal, and B. W. Anthony, "Mapping the design space of photonic topological states via deep learning," *Opt. Exp.*, vol. 28, no. 19, pp. 27893–27902, Sep. 2020.
- [15] T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, "Training of photonic neural networks through in situ backpropagation and gradient measurement," *Optica*, vol. 5, no. 7, pp. 864–871, Jul. 2018.
- [16] Y. Long, J. Ren, Y. Li, and H. Chen, "Inverse design of photonic topological state via machine learning," *Appl. Phys. Lett.*, vol. 114, no. 18, 2019, Art. no. 181105.
- [17] "ANSYS photonics interconnect." [Online]. Available: <https://www.ansys.com/products/photonics/interconnect>
- [18] W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walsmley, "Optimal design for universal multiport interferometers," *Optica*, vol. 3, no. 12, pp. 1460–1465, Dec. 2016.
- [19] F. Shokraneh, S. Geoffroy-Gagnon, M. S. Nezami, and O. Liboiron-Ladouceur, "A single layer neural network implemented by a 4×4 MZI-based optical processor," *IEEE Photon. J.*, vol. 11, no. 6, Dec. 2019, Art. no. 4501612.
- [20] Q. L. Bao et al., "Monolayer graphene as a saturable absorber in a mode-locked laser," *Nano Res.*, vol. 4, no. 3, pp. 297–307, 2011.
- [21] M. Babaieian et al., "Nonlinear optical components for all-optical probabilistic graphical model," *Nature Commun.*, vol. 9, no. 1, pp. 1–8, May 2018.
- [22] M. Soljacic, M. Ibanescu, S. G. Johnson, Y. Fink, and J. D. Joannopoulos, "Optimal bistable switching in nonlinear photonic crystals," *Phys. Rev. E*, vol. 66, no. 5, Nov. 2002, Art. no. 055601.
- [23] C. Rios et al., "Integrated all-photonic non-volatile multi-level memory," *Nature Photon.*, vol. 9, no. 11, pp. 725–732, Sep. 2015.
- [24] I. A. D. Williamson, T. W. Hughes, M. Minkov, B. Bartlett, S. Pai, and S. H. Fan, "Reprogrammable electro-optic nonlinear activation functions for optical neural networks," *IEEE J. Sel. Topics Quantum Electron.*, vol. 26, no. 1, Jan./Feb. 2020, Art. no. 7700412.
- [25] M. M. P. Fard et al., "Experimental realization of arbitrary activation functions for optical neural networks," *Opt. Exp.*, vol. 28, no. 8, pp. 12138–12148, Apr. 2020.
- [26] L. Zhang et al., "DQN-based mobile edge computing for smart internet of vehicle," *EURASIP J. Adv. Signal Process.*, vol. 2022, no. 1, pp. 1–16, 2022.
- [27] Z. Liu, L. Raju, D. Zhu, and W. Cai, "A hybrid strategy for the discovery and design of photonic structures," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 10, no. 1, pp. 126–135, Mar. 2020.
- [28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [29] J. J. Berman, "Indispensable tips for fast and simple Big Data analysis," in *Principles and Practice of Big Data (Second Edition)*. Academic, 2018, pp. 231–257, doi: [10.1016/B978-0-12-815609-4.00011-X](https://doi.org/10.1016/B978-0-12-815609-4.00011-X).