OPTICA

Distillation-guided optical neural networks with reinforcement learning-assisted calibration

KANGJIAN DI,[1,2,†] FUHAO YU,[2,3,†] SILIN CHEN,[1,2,†] JIASHU LI,[1] ANDY LIU,[1] SEN SHAO,[1] ZHIYUAN SHI,[2,3] XUPING ZHANG,[2,3] YIXIN ZHANG,[2,3] WEI JIANG,[2,3] AND NINGMU ZOU[1,2,4,*]

[1]School of Integrated Circuits, Nanjing University, Suzhou 215163, China
[2]Ministry of Education Key Laboratory of Intelligent Optical Sensing and Manipulation, Nanjing University, Nanjing 210023, China
[3]College of Engineering and Applied Sciences, Nanjing University, Nanjing 210023, China
[4]Interdisciplinary Research Center for Future Intelligent Chips (Chip-X), Nanjing University, Suzhou 215163, China
[†]These authors contributed equally to this work.
*nzou@nju.edu.cn

Received 21 January 2026; revised 16 May 2026; accepted 2 July 2026; published 22 July 2026

Optical neural networks offer a promising route to overcome the inference-speed and energy-efficiency bottlenecks of conventional electronic neural networks, owing to their high parallelism and low-latency optical computation. However, practical on-chip training of optical neural networks remains challenging because gradient estimation, hardware nonidealities, and device-compatible parameter updates are difficult to handle directly in optical systems. Here, we propose an on-chip training distillation-guided optical neural network (DGONN) and introduce a forward distillation algorithm to address these challenges. The forward distillation algorithm avoids chain-rule-based optical backpropagation and enables hardware-aware gradient estimation through measured forward differences, thereby mitigating the instability of long-range gradient propagation in photonic chips. The block-MZI optical parameterization reduces the trainable parameter count compared with dense electronic neural networks, while layer-wise distillation recovers the accuracy loss caused by this constrained optical parameterization and improves training stability. Numerical and experimental results demonstrate the high classification accuracy of our DGONN system across various datasets [CIFAR-10, Fashion-MNIST, Handwriting-MNIST, free spoken digit dataset (FSDD), and distributed acousto-optic sensing (DAS)]. Additionally, deep reinforcement learning further enables the controllability and stability of the on-chip optical system. Our work could pave the way for efficient ONNs, offering a promising approach to overcoming the training challenges of optical neural networks and enabling the potential development of energy-efficient, low-power AI systems using photonic technology.

https://doi.org/10.1364/OPTICA.591264

1. INTRODUCTION

With the widespread application of artificial intelligence (AI) through large-scale computing, particularly with the rise of large language models, electronic neural networks with hundreds of millions or even billions of weight parameters are facing increasingly severe bottlenecks in inference speed and energy efficiency [1]. Despite their remarkable successes in image recognition and natural language processing, the soaring computational and energy demands of deep learning models are pushing conventional hardware to its limits. These bottlenecks have been further intensified by the slowdown of Moore's Law, as the semiconductor industry confronts the fundamental physical limits of transistor scaling [2].

Optical neural networks (ONNs) offer inherent advantages—such as high parallelism, low latency, and low power consumption—making them attractive for accelerating AI workloads [3–5]. Leveraging optical interference and diffraction, recent photonic accelerators have demonstrated computational efficiencies exceeding 100 tera operations per second per watt (TOPS/W) [6]. Chips like Taichi have achieved 160 TOPS/W in practical AI tasks [7], and on-chip platforms based on Mach–Zehnder interferometers and microring resonators have enabled scalable, programmable architectures [8–13]. However, most ONNs still depend on pre-trained electronic weights mapped onto photonic hardware, a strategy that overlooks critical physical nonidealities such as thermal drift and device noise.

To overcome these limitations, on-chip training approaches have been developed to update model parameters directly within the optical domain. Techniques based on optical backpropagation [4], feedback-assisted hybrid learning [6], and forward-only updates [14–17] have improved adaptability to hardware imperfections. Among them, forward propagation training schemes [13,14] avoid the complexity of optical feedback and are better suited to real-time, scalable implementation. Besides, while some advanced *in situ* training methods are effective, they necessitate

additional hardware for optical field measurement [18] and require the device to exhibit Lorentz symmetry [19]. Consequently, challenges remain in parallel gradient estimation, computation and energy efficiency, and dynamic model compression. These factors impede the scalability of physical neural networks, making it difficult for them to match the performance of their electronic counterparts.

One promising direction to address these constraints is knowledge distillation [20,21]. By transferring the output behavior of a large teacher network to a smaller student model, it enables on-the-fly compression while maintaining accuracy [22,23]. Crucially, forward-propagation-based training avoids chain-rule-based optical backpropagation and enables gradient estimation through measured forward differences, thereby better matching the forward-measurement nature of optical systems. In particular, on-chip forward distillation provides layer-wise teacher supervision for the optical student network, reducing the dependence on long-range error propagation and improving training stability under hardware nonidealities.

Another longstanding obstacle in optical computing lies in system calibration. Variations in fabrication and operating conditions often require time-consuming spatial and temporal characterization [6,24,25]. To reduce this burden, reinforcement learning-based calibration strategies have been introduced, enabling more efficient error compensation during the testing phase without the need for full hardware recharacterization [26]. However, conventional deep reinforcement learning (DRL) methods, such as deep deterministic policy gradient (DDPG), rely on experience replay, which can suffer from instability and slow convergence when faced with dynamic environments and long-term computation. This instability hampers the algorithm's effectiveness in real-world applications. To overcome these limitations, the proximal policy optimization (PPO) algorithm [27] has emerged as a more stable and sample-efficient alternative. PPO avoids the instability caused by experience replay by using on-policy updates and clipped objective functions, making it more suitable for real-time calibration in dynamic optical systems.

Here, we introduce a distillation-guided optical neural network (DGONN) as an on-chip training framework that addresses the intertwined challenges of parallel gradient estimation, computational efficiency, and energy efficiency in ONNs. The DGONN employs a forward distillation algorithm for hardware-aware on-chip training based on optical forward measurements, avoiding chain-rule-based optical backpropagation while retaining forward difference gradient estimation for tunable photonic parameters. In this framework, the block-MZI optical parameterization is responsible for reducing the trainable parameter count while maintaining accuracy and robustness. In addition, to ensure stable operation in dynamic environments, we integrate an electronically implemented PPO module for real-time calibration, which compensates for hardware-induced perturbations without requiring full system recharacterization. The effectiveness of the DGONN across diverse benchmark datasets demonstrates its potential as a compact, energy-efficient, and fully trainable photonic AI platform—bringing ONNs one step closer to practical deployment in real-world intelligent systems.

2. RESULTS

A. Electric-Optic Hybrid Neural Network System with On-Chip Distillation Training

Figure 1 shows the overall architecture of the DGONN, which is based on an electric-optical hybrid neural network. The architecture fully exploits the high expressiveness of electronic neural networks and the ultrafast, energy-efficient inference capabilities of integrated photonics, in which an EDNN supervises the training of an on-chip ONN via knowledge distillation. We propose an improved distillation method by constructing the EDNN as the teacher network, as shown in Fig. 1(a). Unlike conventional knowledge distillation schemes, we adopt a feature distillation approach: specifically, we first utilize the teacher network EDNN to extract features from the input data, obtaining electric feature representations $DF_N = \{DL_1, DL_2, \ldots, DL_N\}$, where N denotes the depth of the network. We use the output from the feature extraction layer of the EDNN as the input to the student network ONN. The output of each optical block layer (OBL) in the ONN, after passing through the electronic activation function, can be represented as $OF_N = \{OBL_1, OBL_2, \ldots, OBL_N\}$. Furthermore, we compute per-layer losses between the corresponding outputs of the ONN and the EDNN, enabling the model to learn effective feature distributions at every stage and thereby facilitating the training of the ONN.

For a given input sample x, the teacher feature and the student output at the nth distilled layer are denoted as $DL_n(x)$ and $OBL_n(x)$, respectively. Both have the same feature dimension K_n after dimension matching, where K_n denotes the feature dimension of the nth layer. The layer-wise feature distillation loss $\mathcal{L}_n(x)$ is defined as

$$\mathcal{L}_n(x) = \frac{1}{K_n} \sum_{k=1}^{K_n} \left|\left| DL_{n,k}(x) - OBL_{n,k}(x) \right|\right|_2^2, \tag{1}$$

where $DL_{n,k}(x)$ denotes the kth component of the EDNN teacher feature at the nth layer, and $OBL_{n,k}(x)$ denotes the kth component of the ONN student output at the corresponding layer. Accordingly, Loss1 and Loss2 in Fig. 2(b) correspond to the layer-wise losses L_1 and L_2, respectively, and are calculated between the corresponding ONN outputs and EDNN teacher features.

The total distillation loss for the input sample x is then computed as

$$\mathcal{L}_{\text{dist}}(x) = \sum_{n=1}^{N} \lambda_n \mathcal{L}_n(x), \tag{2}$$

where λ_n is a weighting coefficient used to balance the distillation effects at different layers. The overall framework of our method involves using the features extracted by the teacher network EDNN as inputs to the student network ONN. Simultaneously, we compare and compute losses for each layer's outputs during the training process. Through end-to-end optimization, the model is ultimately trained to achieve effective knowledge transfer. Figure 1(b) shows the on-chip ONN method for loading database features, where an optical splitter divides a beam of light into equal parts on-chip, ensuring that the phases of the inputs entering the MZIs array remain relatively consistent. Through cascaded multi-layer 1×2 MZI structures, the system achieves signal demultiplexing with a factor of 2^N. The data are loaded onto the optical carrier via intensity modulation using a thermo-optical

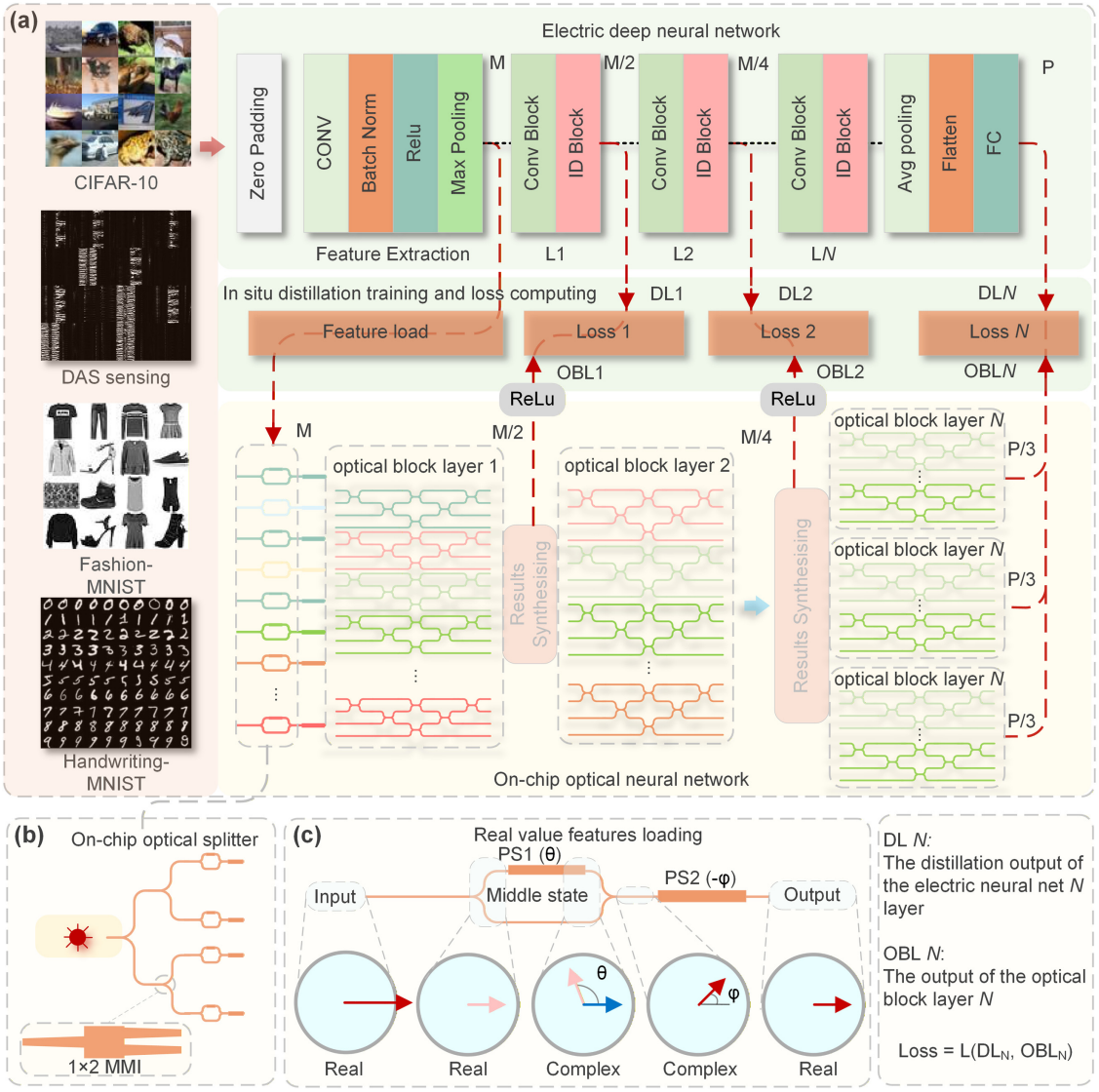

Fig. 1. Schematic diagram of the electric-optic hybrid neural network system with on-chip distillation training. (a) Multiple benchmark data modalities, including CIFAR-10 images, distributed acousto-optic sensing (DAS) signals, Fashion-MNIST images, and handwriting digits, serve as input sources to the hybrid model. An EDNN composed of convolutional layers, identity (ID) blocks, and fully connected layers extracts hierarchical feature representations from the input data. On-chip layer-wise distillation is employed, where feature outputs from each EDNN layer are fed into the corresponding OBLs, and the layer-wise distillation losses are computed to guide training. The on-chip optical neural network consists of an integrated photonic circuit with multiple cascaded optical block layers performing matrix–vector multiplications. (b) On-chip optical splitter: after light from a single source enters the chip, the optical path is separated through a cascade of multiple 1×2 MMIs to achieve multi-channel data loading. (c) Real-valued features are loaded into the optical system via phase shifters (PS1 and PS2), enabling real-to-complex mapping for efficient optical computation.

single-drive modulator. However, this approach introduces an additional phase component, resulting in modulation chirp. As a consequence, the output signal becomes complex-valued, even though the input features of the ONN are real-valued. Therefore, we designed the signal loading approach shown in Fig. 1(c), which employs PS1 for intensity modulation and PS2 for phase restoration. Details on real-valued loading are provided in Section 4.

B. Forward Algorithm Flow in Optical Neural Networks

In conventional electronic neural networks, the backpropagation algorithm computes gradients based on the chain rule of derivatives during the reverse pass. However, implementing the traditional backpropagation algorithm in ONNs is highly challenging. This difficulty arises primarily from the analog and continuous nature of optical signal processing, which is fundamentally mismatched with the discrete and high-precision operations required by conventional gradient-based learning methods. Given these challenges, an alternative approach has been developed where forward algorithms are employed to compute gradients. The standard forward algorithm framework is shown in Fig. 2(a). In this setup, the input data $\boldsymbol{X_1}$ are passed through each layer of the network, which is represented by the function $f_o(\boldsymbol{X_i}, \boldsymbol{\theta_i})$, where $\boldsymbol{X_i}$ is the input to the ith layer, and $\boldsymbol{\theta_i}$ represents the parameters of that layer. The forward pass for each layer can be described as

$$f_o(\boldsymbol{X_1}, \boldsymbol{\theta_1}) \rightarrow f_o(\boldsymbol{X_2}, \boldsymbol{\theta_2}) \rightarrow f_o(\boldsymbol{X_3}, \boldsymbol{\theta_3}) \ldots \rightarrow f_o(\boldsymbol{X_N}, \boldsymbol{\theta_N}). \quad (3)$$

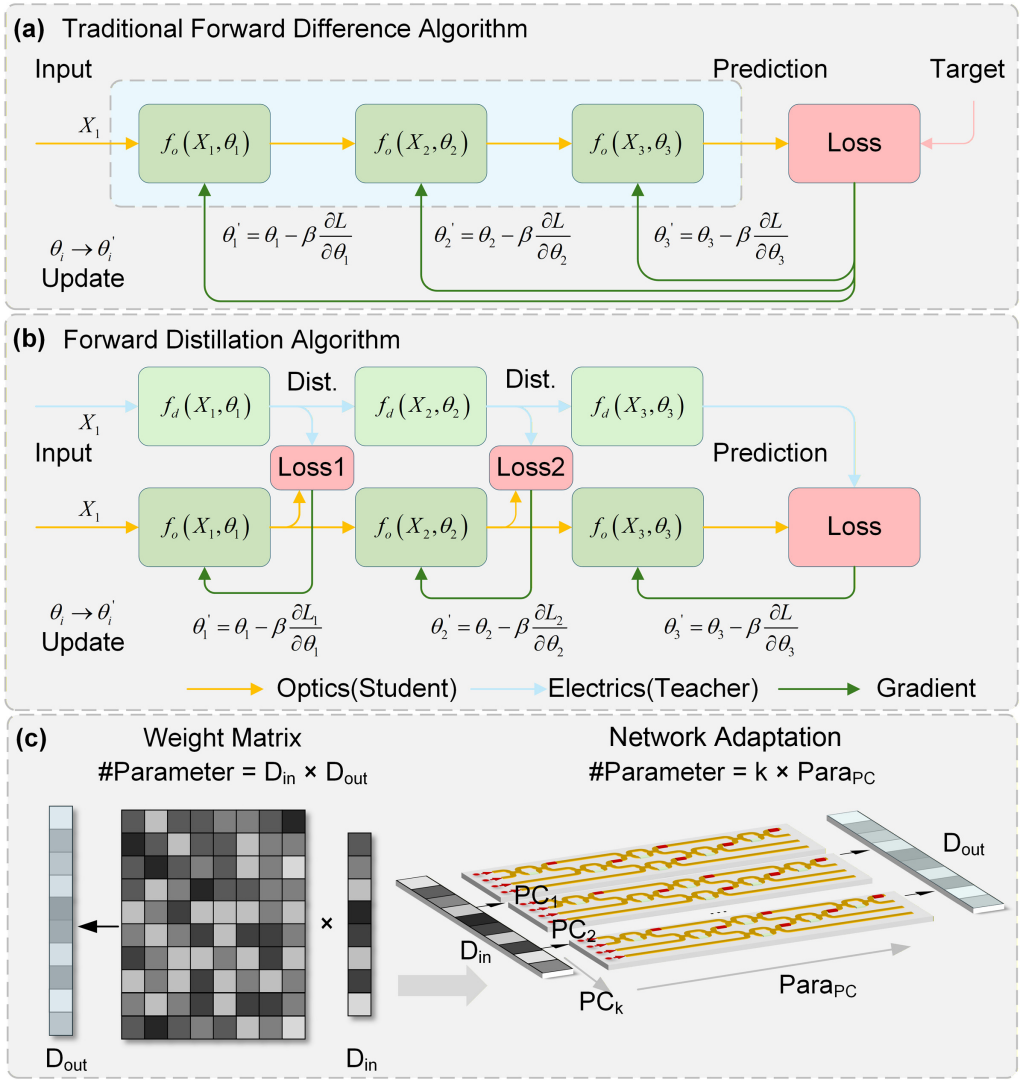

Fig. 2. Forward propagation-based gradient computation and weight loading in optical neural networks. (a) Forward algorithm, where input data X_1 passes through multiple layers of the network, and the loss is computed to update the parameters of each layer. (b) Forward distillation algorithm, where each layer is equipped with a distillation loss function (Loss1 and Loss2), improving gradient propagation and network performance during training. (c) Hardware architecture, illustrating the weight matrix of the DGONN and its corresponding adaptation to the EDNN system.

The network output is compared to the target, resulting in a loss function L. Gradients of the loss with respect to the parameters $\boldsymbol{\theta_i}$ are then computed for each layer, and the parameters are updated using gradient descent:

$$\boldsymbol{\theta}_{\boldsymbol{i}}^{'} = \boldsymbol{\theta_i} - \beta \frac{\partial L}{\partial \boldsymbol{\theta_i}}, \tag{4}$$

where β is the learning rate. This method avoids direct back-propagation and instead utilizes gradients derived from forward differences for parameter updates, which aligns better with the analog nature of optical systems. However, the traditional forward algorithm requires computing gradients for each layer sequentially during training, which limits the potential for parallelization and is prone to vanishing gradients in deep networks. To address these issues, the forward distillation algorithm introduces a distillation mechanism at each layer, supervising intermediate layers through additional distillation loss functions (e.g., Loss1 and Loss2). This design enables fully parallel computation of local gradients across layers, in contrast to traditional forward-only training, which requires sequential gradient estimation. As a result, the training process is decoupled across layers, significantly reducing computation time and mitigating vanishing gradients in deeper networks. These losses help mitigate gradient degradation, enhance training stability, and encourage the learning of more discriminative feature representations at each layer. As shown in Fig. 2(b), the parameter update rule under this paradigm is

$$\boldsymbol{\theta}_{\boldsymbol{i}}^{\prime} = \boldsymbol{\theta_i} - \beta \frac{\partial L_{\boldsymbol{i}}}{\partial \boldsymbol{\theta_i}}. \tag{5}$$

Here, $L_{\boldsymbol{i}}$ represents the individual distillation losses for the intermediate layers. As a teacher model, the output of each layer of the EDNN corresponds to the output of the ONN. A loss is calculated between the outputs, enabling the EDNN to guide the ONN. This operation allows for the distributed parallel computation of gradients for each layer during the on-chip training of the ONN, significantly reducing the training time.

Figure 2(c) specifically shows the process of dimensionality transformation performed by the DGONN and the EDNN.

In the EDNN, a linear transformation from $D_{in} \times 1$ to $1 \times D_{out}$ requires $D_{in} \times D_{out}$ parameters. In contrast, the DGONN leverages the structured linear transformations of MZI subnets to achieve complex mappings with significantly fewer parameters. Specifically, dense linear layers trained in neural networks typically exhibit substantial parameter redundancy; by coupling knowledge distillation with the reuse of MZI subnetworks, our framework compresses the representational basis of a general linear transform into a block-diagonal subspace. This approach markedly reduces the parameter count while maintaining model performance. The parameter count for a single MZI subnet is $Para_{PC}$, and to achieve the same dimensional transformation task, only k MZI subnets are needed, resulting in a total parameter count of $k \times Para_{PC}$. In the photonic chip composed of MZI subnets, $Para_{PC}$ corresponds to the number of phase shifters in the MZI subnets. To enable flexible adaptation to different feature dimensionalities across layers, we allow for dimensionality reduction or expansion by stacking MZI subnets and applying optical filtering. This could enable the network to match layer-wise requirements and enhance representational capacity. For example, when the input is of dimension M and the output is $M/2$, downsampling can be applied. Conversely, when the input is of dimension M and the output is $2M$, the dimensionality can be increased by stacking two $M \times M$ MZI subnets, which enhances the capacity to process higher-dimensional data. The specific theory and the training process related to the MZI subnets can be found in Section S4 of Supplement 1.

C. Numerical Experiments for the DGONN

We assessed the effectiveness of the proposed DGONN approach by simulation of an on-chip multi-stage MZI architecture for neural networks, using datasets from CIFAR-10 [28], Fashion-MNIST [29], Handwriting-MNIST [30,31], and DAS [32]. The relevant results are shown in Fig. 3 and Table 1. As shown in Fig. 3(a), the teacher model, EDNN, is constructed with fully connected layers whose number and configuration vary depending on the specific classification task. CIFAR-10, Fashion-MNIST, and Handwriting-MNIST are used for the 10-class classification task, while DAS is used for the four-class classification task. The DGONN model adopts the same number of layers as the corresponding EDNN for both training and testing. The specific parameters for the feature extraction part of the EDNN are provided in Section S6 of Supplement 1. Due to the use of 4×4 MZI as the underlying architecture, when performing the 10-class classification task, the final output layer requires three parallel 4×4 MZIs to produce the 10-class output. First, the feature extraction part of the EDNN is constructed, which includes a convolutional section that extracts features from the raw data. As shown in Fig. 3(e), the input CIFAR-10 dataset and the gradient-weighted class activation mapping (Grad-CAM) after feature extraction are displayed, compressing the features of all datasets into a 512-dimensional vector. Then, the fully connected part of the EDNN is pre-trained. As shown in Fig. 3(b), we compare the accuracy and recall of the conventional EDNN, the NDNN (an ONN trained via the forward difference algorithm without distillation), and our proposed DGONN across four datasets. On average, the DGONN achieves a 6.5% improvement in accuracy compared to the NDNN. In essence, the DGONN is a distilled variant of the EDNN, yet its performance degradation is approximately 1% relative to the original EDNN. The specific training curves of the EDNN-guided DGONN across the four datasets are shown in Fig. 3(f). Furthermore, as shown in Fig. 3(c), the DGONN significantly reduces computational and model complexity, achieving up to two orders of magnitude reduction in parameter count compared to the EDNN, along with FLOPs reductions of 98.53%, 98.15%, 98.09%, and 98.3% across the four datasets, respectively. For the four datasets, the corresponding models can reduce the parameter count by more than 95%. Figure 3(d) shows the confusion matrix for CIFAR-10 in the DGONN, achieving an accuracy of 95.27%. The CIFAR-10 dataset consists of 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Compared with the NDNN, the accuracy improves by 8.27%. A key innovation of the DGONN is its support for parallel computation, with each 4×4 MZI unit performing computations independently. Combined with the optical distillation framework, this design enhances computational and energy efficiency by an impressive factor of 90%. On CIFAR-10, the EDNN and the DGONN exhibit diverging trends in accuracy as the number of layers increases; both models experience performance degradation due to overfitting, as shown in Fig. S11 of Supplement 1. Figures 3(g) and 3(h) show the t-SNE plots for the NDNN and the DGONN on the CIFAR-10 dataset, based on the output from the previous layer of the respective networks. It can be observed that the DGONN achieves better t-SNE results, enabling precise classification of each category. In contrast, categories such as "frog," "airplane," and "deer" in the NDNN are clustered together, showing lower separability. Figure S12 of Supplement 1 shows that the DGONN requires fewer forward passes during training than the NDNN. Comprehensive network parameters and performance metrics are shown in Table 1, while confusion matrices for all datasets are shown in Fig. S13 of Supplement 1.

We then conducted physical experiments on the Fashion-MNIST and FSDD datasets to evaluate the performance of our DGONN model [29,33]. To accomplish this, we employ the on-chip MZI architecture shown in Figs. 4(a) and 4(b). First, a 1550 nm laser (Santec TSL510, 1 mW output power and $\sim$ 200 kHz linewidth) is coupled into the on-chip MZI through the grating coupler located at the upper right side. After propagating through a long waveguide, the light is split by a splitter array on the left, which consists of three 1×2 MMIs and divides the light into four paths. These four optical paths enter the data-loading region of the on-chip MZI. Four parallel phase shifters serve as intensity modulators, enabling real-valued data loading with phase correction provided by PS2. The modulated signals then enter the subnetwork module composed of cascaded MZIs, and subsequently into multiple detection channels of the power meter (HP 81531A, bandwidth: DC to 300–400 Hz). The data are loaded into both the input array and the MZI array using a multi-channel current source (NI 9266, 24 kS/s update rate per channel and 16-bit resolution). Above Fig. 4(c), we compare the feature distributions of the EDNN and the DGONN on two datasets. Both network models use a three-layer architecture. During training, a forward distillation algorithm is applied, with the EDNN guiding the DGONN in the training process. As the number of layers increases, errors accumulate. However, our results show that the MAE on the two datasets are 0.053 and 0.0237, respectively, demonstrating that the feature distribution of the EDNN can be almost perfectly restored. Additionally, when comparing the training performance with the NDNN, we observe that our proposed DGONN converges faster and achieves higher accuracy. For all samples, the MSE is shown in Fig. S14 of

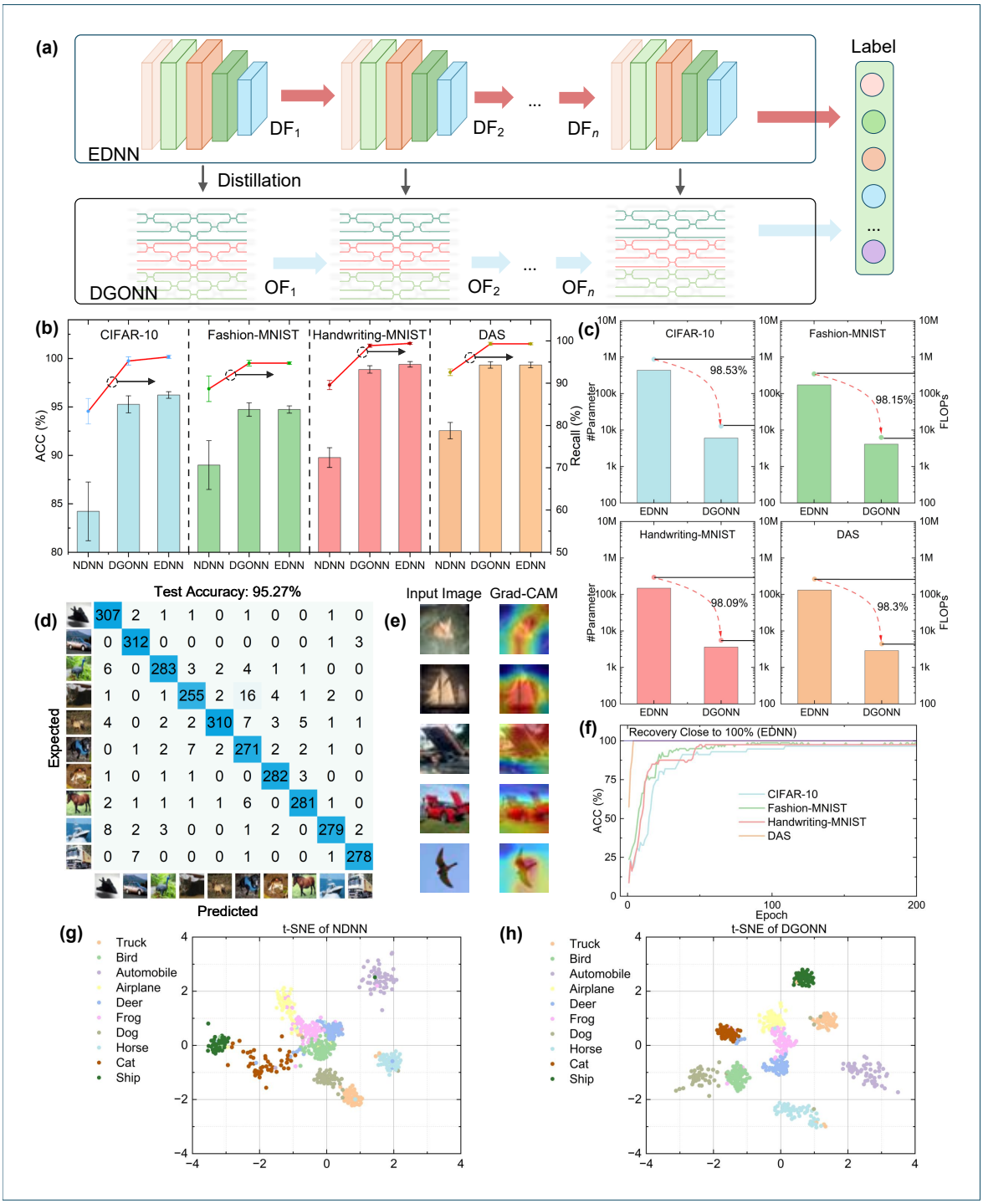

Fig. 3. Numerical experimental results of the DGONN and the EDNN, with the datasets including CIFAR-10, Fashion-MNIST, Handwriting-MNIST, and DAS. (a) The scheme of the EDNN and the DGONN. (b) Comparison of accuracy and recall before and after distillation across the four datasets. (c) Comparison of the number of parameters and FLOPs between the EDNN and the DGONN for the four datasets. (d) Confusion matrix after distillation on the CIFAR-10 dataset. (e) Input image and Grad-CAM of the feature extraction layer output for CIFAR-10. (f) Training process of the DGONN across the four datasets under the supervision of the EDNN. (g) and (h) t-SNE visualizations of the non-distilled (NDNN) and the DGONN on CIFAR-10.

Supplement 1. Furthermore, Fig. 4(d) shows a comparison of the confusion matrices for both models. The DGONN consistently achieves higher accuracy, demonstrating its better suitability for on-chip training.

D. DRL-Based MZI Subnet Calibration

During the training process of the DGONN, it is necessary to calibrate the photonic chip to ensure proper operation. The calibration procedure ensures that all MZI units are precisely

Table 1. Comparison of Accuracy and Parameter Count for the Four Tasks

Task	EDNN Accuracy	NDNN Accuracy	DGONN Accuracy	Recovery	EDNN Parameters	DGONN Parameters
CIFAR-10	96.23%	87%	95.27%	99%	1,221,248	11,760
Fashion-MNIST	94.73%	91.33%	94.73%	100%	172,672	4080
Handwriting-MNIST	99.4%	90.67%	98.87%	99.46%	148,096	3600
DAS	99.67%	93.33%	99.67%	100%	132,096	2880

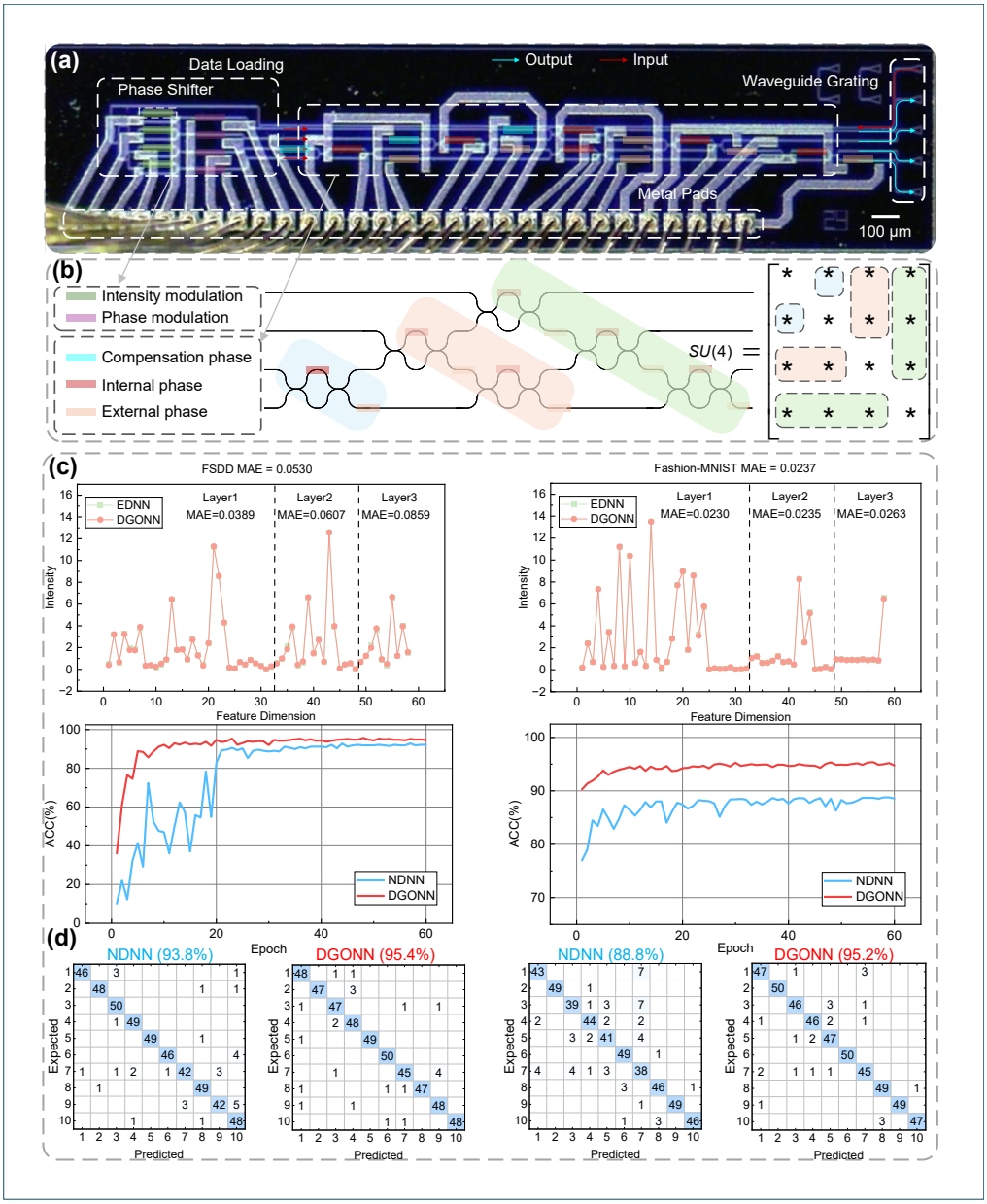

Fig. 4. Experimental results of machine learning implemented on a photonic chip. (a) Photonic chip used for experimental validation, showing key components such as phase shifters, metal pads, and waveguide gratings. (b) The phase shifter parameters in the photonic chip have different functions for different phase shifters within the chip, which can be used to construct a unitary matrix SU(4) based on the 4 × 4 MZI architecture. (c) Comparison of feature layer distributions and training curves for the EDNN and DGONN models on the FSDD (left) and Fashion-MNIST (right) datasets. Additionally, the corresponding training curves for the NDNN and DGONN models are shown, highlighting their performance across the two datasets. (d) Confusion matrices of the NDNN and DGONN models on the FSDD (left) and Fashion-MNIST (right) datasets.

biased at their ideal operating points during the forward distillation algorithm. This calibration procedure delivers three critical benefits: (1) effectively suppresses abrupt phase current jumps during differential iterations; (2) prevents gradient vanishing phenomena caused by operation point deviation into the nonlinear region; and (3) eliminates output intensity oscillations induced by inappropriate step sizes. However, static calibration alone cannot accommodate the continuous environmental fluctuations (e.g.,

thermal drift or electrical noise) that affect the phase stability of MZI elements during training and inference. To address this limitation, real-time dynamic calibration is introduced. It enables the system to: (1) adaptively compensate for environmental disturbances in phase modulation; (2) significantly enhance the convergence efficiency of MZI array parameter optimization.

We performed the calibration using both the conventional testing method and a DRL-based calibration method. Figure 5(a) shows the PPO-based DRL algorithm primarily employed in the experimental study of this work, which transforms the calibration problem into a Markov decision process [34], enabling reinforcement learning algorithms to automatically optimize phase control parameters. In this calibration process, the physical MZI chip is treated as the environment. The state of the PPO agent contains the phase-shifter currents and the measured optical powers from the four output ports. The action is a continuous current vector directly applied to the phase shifters, with each channel constrained within the experimentally allowed range of 0–10 mA. The reward is defined by the sum of squared deviations between the target and measured output power distributions, so that the policy is optimized toward the desired optical response. The detailed mathematical definition of the state, action, and reward function is provided in Section S3 of Supplement 1.

Figure 5(b) shows the stages involved in the DRL-based calibration procedure. When driving the MZM array to produce a normalized optical field of [1, 0, 1, 0], the system was first prebiased to the identity matrix and then deliberately deviated by environmental fluctuations to activate the calibration system; subsequently, the MZI phase modulators coordinately adjusted their parameters to make the output power converge toward the target distribution [1, 0, 1, 0]. In addition, Fig. S17 of Supplement 1 shows an on-chip random-target calibration test using a nonbinary target distribution [0.8, 0.1, 0.9, 0.2], indicating that the PPO-based calibration is not restricted to the binary response. The study concurrently evaluated four distinct calibration algorithms for the on-chip calibration task. Figure 5(c) shows the convergence accuracy and error rates on the validation set. For a fair comparison, PPO, DDPG, GA, and PSO were evaluated under the same on-chip sampling budget, with the same maximum number of optical power measurements. Notably, the PPO algorithm demonstrated superior performance across both metrics, achieving a mean convergence accuracy of 94.0% with an error margin of $\pm 2.1\%$, thereby outperforming the other three approaches.

To more systematically evaluate the effectiveness of DRL in stabilizing the on-chip optical system, we compared two calibration methods: DDPG and PPO. DDPG relies on experience replay, storing historical data for policy updates. However, this approach is sensitive to dynamic environmental changes, leading to potential instability in the optimization process. In contrast, PPO is an on-policy algorithm that updates the policy based solely on the most recent environmental observations [27], avoiding issues caused by outdated data. As a result, PPO is particularly suitable for real-time calibration in optical systems, offering better adaptability to noise and hardware drift.

As shown in Fig. 5(d), PPO converges to a stable optimum within just 26 episodes, while Fig. 5(e) shows that DDPG requires around 80 episodes and exhibits noticeable oscillations and slower convergence due to its off-policy nature. These results highlight the superior robustness and adaptability of PPO for online calibration under dynamic conditions. The overhead of PPO calibration mainly comes from on-chip sampling and electronic policy inference. In our implementation, the PPO policy is obtained before on-chip training through interaction with the chip. After this initial calibration stage, only a few policy steps are typically needed for fine tuning when the measured output deviates from the target response. Thus, the additional online overhead during DGONN training is relatively small compared with the optical measurements required by forward-difference training, although larger environmental drift may require extra sampling for policy updating. Figure 5(f) shows the system classification performance on the Fashion-MNIST dataset, evaluated under conditions of intentionally induced weight misalignment. The system obtained without PPO calibration shows a significantly degraded accuracy of 69.2%. In contrast, the system obtained after the application of the online calibration protocol successfully recovers the performance, achieving a high accuracy of 95.2%. Figure 5(g) shows a corresponding evaluation on the FSDD dataset, where the uncalibrated accuracy drops to 62.8% due to environmental deviations. After real-time calibration, the system accuracy is restored to 95.4%, demonstrating the critical role of dynamic calibration in maintaining the computational integrity of the DGONN against operational perturbations.

The PPO calibration policy has a certain degree of transferability on the same chip. When the laboratory conditions remain relatively stable, a previously trained policy can usually be reused. If ambient temperature or chip thermal state changes lead to calibration errors, the policy can be fine-tuned with a small number of additional on-chip samples. For different chip instances, direct reuse is more difficult because fabrication variations change the current-phase response and the output power landscape. Chip-specific fine tuning is therefore still needed, and cross-chip transfer will be explored in future work.

3. DISCUSSION

In this study, we propose the DGONN, an on-chip training framework designed to address key challenges in ONNs, such as gradient estimation and control complexity. By adopting a forward distillation algorithm-based training scheme, the DGONN avoids chain-rule-based optical backpropagation and improves trainability under the constrained block-MZI optical parameterization, aligning well with the forward-measurement nature of optical systems.

To clarify the contribution of each component, we summarize the proposed framework as a three-part design. First, the block-MZI optical parameterization is responsible for the major reduction in trainable parameters. Second, the proposed layer-wise distillation does not further reduce the number of photonic parameters but significantly recovers the accuracy loss caused by the constrained optical parameterization. Third, PPO-based calibration does not alter the model architecture but restores the computational performance under intentional hardware perturbations. A detailed component-wise ablation analysis is shown in Table S2 of Supplement 1.

The input data are a 4×1 real-valued vector, which undergoes a multiplication operation with a 4×4 complex matrix, resulting in a 4×1 complex vector. This process requires 56 FLOPs. Due to the use of thermo-optic modulation in the input array, it has a limited modulation speed on the order of 50 kbaud. Assuming the detector has sufficient response speed, the computational throughput of the system is primarily limited by the input rate, which

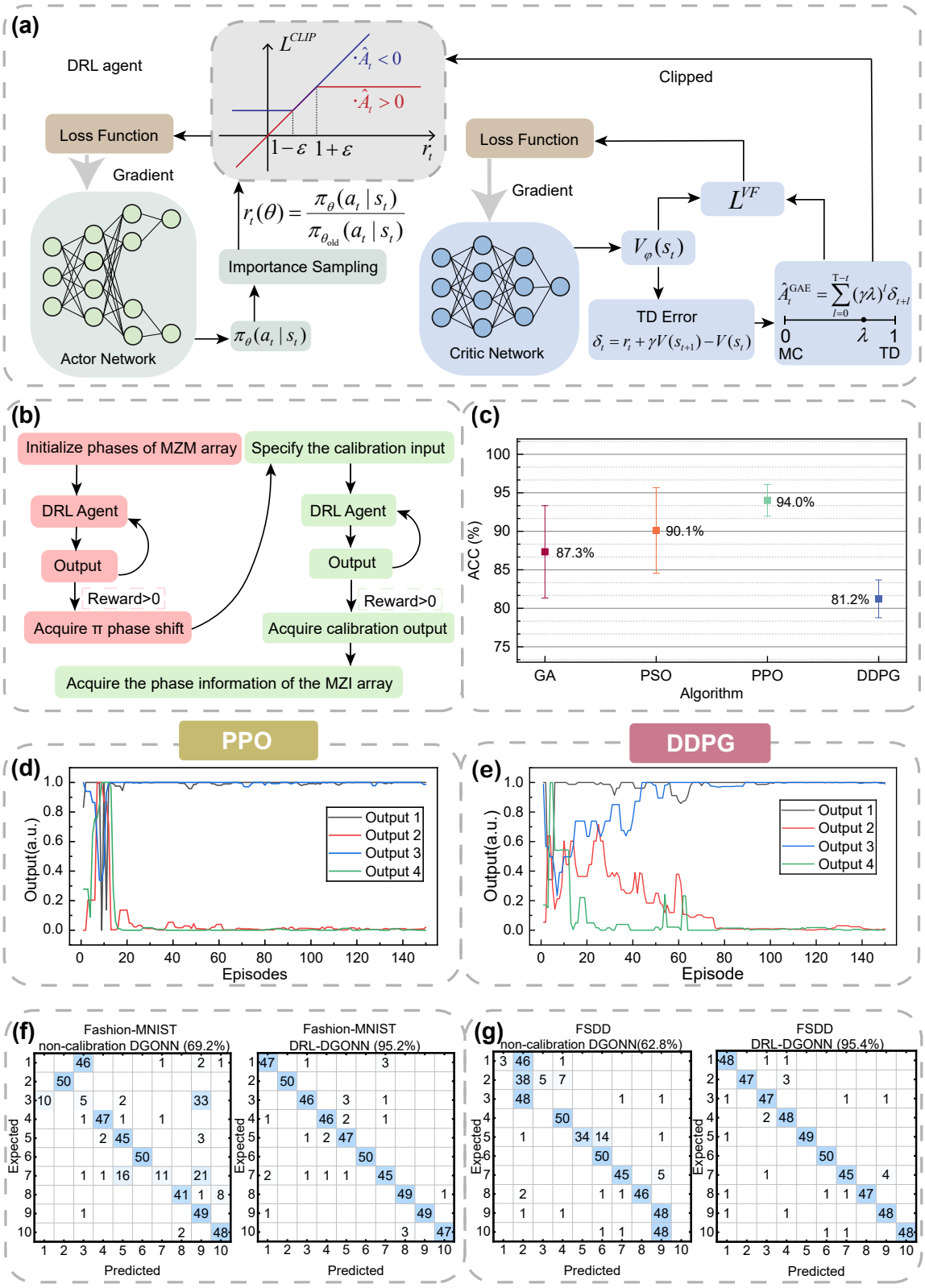

Fig. 5. Schematic of deep reinforcement learning-based calibration and optimization for the MZI system. (a) Flowchart of the DRL agent algorithm based on PPO. (b) Flowchart of the calibration process for the MZI input Mach–Zehnder modulator (MZM) array and the 4 × 4 Reck-configuration MZI using DRL. A DRL agent dynamically optimizes phase modulation parameters through environment interaction, enabling efficient calibration. (c) Accuracy and error values of on-chip training with calibration parameters obtained from four algorithms (GA, PSO, PPO, and DDPG) (based on Fashion-MNIST). (d) Output curves of the PPO algorithm during the calibration of an MZI. The evolution of the four output powers toward the target values is shown. (e) Output curves of the DDPG algorithm during the calibration of an MZI. The evolution of the four output powers toward the target values is shown. (f) Confusion matrix evaluating the robustness of the calibrated DGONN system under intentional weight misalignment on the Fashion-MNIST dataset. (g) Confusion matrix evaluating the robustness of the calibrated DGONN system under intentional weight misalignment on the FSDD dataset.

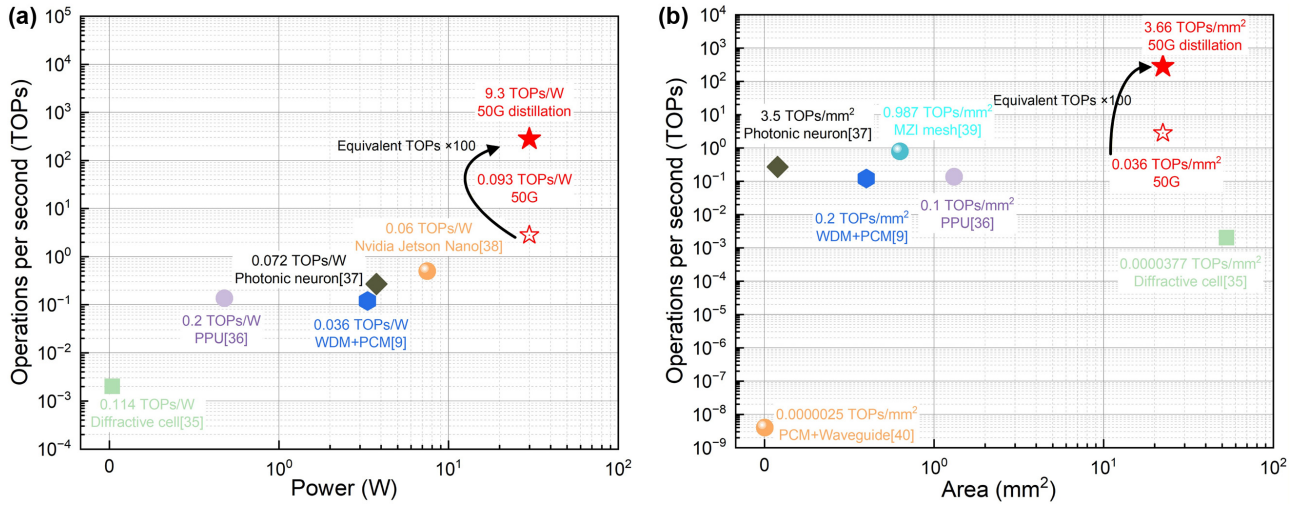

Fig. 6. TOPS, area, and power comparison. (a) Diffractive cell [35], PPU [36], WDM + PCM [9], photonic neuron [37], and Nvidia Jetson Nano [38]. (b) Diffractive cell, PPU, WDM + PCM, photonic neuron, MZI mesh [39], and PCM + waveguide [40].

is determined by the thermo-optic modulator and calculated as 56×50 kbaud $= 2.8$ MFLOPs. In addition, by using high-speed modulators and detectors as inputs and outputs, and expanding the size of the MZI array or adopting multi-wavelength schemes [41], the theoretical calculation speed can be further improved, for example, when the modulator rate is 50 Gbaud [42], the theoretical speed can reach 2.8 TOPS. In addition, similar to the method described in Taichi [7], our proposed approach effectively compresses the data volume. The DGONN reduces the number of trainable parameters by 97.57%–99.04% compared with the dense EDNN teacher models across the four numerical tasks. After parameter compression, the DGONN provides an approximately 100-fold equivalent improvement in computational throughput compared with the dense EDNN baseline, with a corresponding reduction in the normalized computational energy required per equivalent operation. In addition, the layer-wise distillation strategy reduces the effective MZI-subnet forward operations by 37.7% compared with the NDNN during training, which can further reduce the measurement-dominated dynamic training energy, as derived in Section S8 of Supplement 1. In our experiments, power consumption primarily stems from the laser, the NI source, the driver of modulator, and the photodetectors. The estimated power consumption, based on the components used in our measurement setup, is approximately 29.96 W, with about 90% of the energy consumption attributed to benchtop instruments (for details on speed and power consumption estimation, see Section S7 and Table S3 of Supplement 1). Figure 6(a) shows the relationship between computational efficiency and power consumption for various photonic computing paradigms, including diffractive cell, PPU, WDM + PCM, and photonic neuron. Our DGONN system achieves 9.3 TOPS/W, demonstrating promising application potential. At the same time, the area efficiency of the chip should also be taken into consideration, and our DGONN system can reach 3.66 TOPS/mm^2.

Moreover, we propose a novel method for real-valued matrix operations using a single MZI array combined with a forward difference algorithm. This approach enables direct light intensity output measurements via photodetectors, eliminating the need for complex phase information processing. Compared with conventional MZI-array-based methods relying on singular value decomposition (SVD)—which require multiple cascaded arrays and sophisticated hardware to isolate and control phase—our method offers a simpler implementation and significantly reduces system complexity and experimental cost. While the intensity-only measurement scheme simplifies the setup, it limits the system's ability to support accurate multi-stage computations, as phase effects from one stage can propagate and interfere with subsequent operations. Therefore, our current method is best suited as a lightweight platform for rapid validation of real-valued matrix computations in optical computing. Looking ahead, we plan to enhance computational capability by integrating phase compensation mechanisms directly into a single MZI array or incorporating a small number of additional arrays to partially recover SVD functionality. Furthermore, optimizing the robustness of the forward difference algorithm could improve scalability and performance, paving the way for more practical and cascadable optical computing architectures.

The present 4×4 MZI-mesh subnetwork provides an experimental validation of the DGONN, and the underlying MZI-mesh transformation can, in principle, be extended to larger $N \times N$ matrix operations using the same unitary decomposition principle. For larger input dimensions, the optical linear transformation can be implemented either by increasing the MZI mesh size or by arranging multiple MZI subnetworks in a blockwise parallel architecture. For tasks with more output classes, the final optical/electronic readout layer can be expanded so that the output dimension matches the required number of class logits, for example, by increasing the number of output MZI subnetworks or output channels. For deeper optical networks, additional optical block layers can be cascaded. Importantly, the proposed layer-wise distillation strategy provides local supervision to each optical layer, so the training does not rely on long-chain optical backpropagation through the entire network, which helps reduce gradient degradation and improves the trainability of deeper optical networks.

Scaling MZI meshes to larger dimensions and greater depths still faces practical bottlenecks, including thermal crosstalk, phase drift, accumulated insertion loss, detector noise, and calibration overhead. In our measurements, the output optical power showed a slight shift after approximately 30 min of continuous

operation, mainly due to slow changes in the ambient temperature and the chip thermal state. Accumulated insertion loss in cascaded optical layers can further reduce the optical power at later stages and increase the influence of detector noise. Among these, thermal crosstalk introduces systematic phase bias in larger MZI meshes, which further perturbs the classification margin; a quantitative analysis is provided in Section S8 of Supplement 1. The DGONN can partly mitigate these effects during training and operation through layer-wise distillation, *in situ* training, and PPO-based calibration, but large-scale MZI implementation still requires lower-loss photonic design, better thermal isolation, stable packaging, parallel readout electronics, and efficient control circuits.

Beyond MZI meshes, the proposed framework can, in principle, be transferred to other photonic neural network platforms, provided that the hardware has measurable forward outputs and tunable parameters controlled by external signals. Examples include microring-resonator arrays, diffractive optical networks, phase-change photonic processors, and wavelength-multiplexed photonic processors. In such systems, the input encoding, layer-wise teacher targets, optical forward operator, and PPO calibration policy need to be redesigned according to the specific device response.

Moreover, the present DGONN implementation focuses on MZI-based linear optical transformations, with nonlinear activation implemented outside the optical MZI mesh. Recent nonlinear photonic neuromorphic chips have demonstrated optical-domain nonlinear activation by combining photonic linear computing with nonlinear photonic neuron arrays, such as DFB-SA laser arrays [43]. The DGONN can, in principle, be extended to such systems because its training relies on measured forward outputs rather than analytical backpropagation through each optical component. The nonlinear activation can be treated as part of the measured forward operator, and the layer-wise distillation loss can be applied to the output after the nonlinear module. Integrating the DGONN with on-chip optical nonlinear activation units will be an important direction for future work.

4. MATERIALS AND METHODS

A. Device Fabrication

The chip is manufactured on a commercial SOI wafer, which contains 220 nm of top silicon and a 3 μm thick silicon dioxide insulating layer. The waveguide part of the optical structure is a 220 nm thick strip waveguide, while the grating coupler is a 70 nm shallow etched fan-shaped grating. A layer of oxide is deposited on the silicon waveguide, and a layer of TiN metal is deposited on the oxide layer as a heating electrode. Each heating electrode is 140 μm long and 4 μm wide.

B. DGONN Data Loading

On-chip optical splitter: after light from a single source enters the chip, the optical path is split by a cascade of multiple 1×2 MMIs to achieve multi-channel data loading. Real-valued features are loaded into the optical system via phase shifters PS1 and PS2, enabling real-to-complex mapping for efficient optical computation. It employs PS1 for intensity modulation and PS2 for phase restoration, with the basic principle as follows:

$$E_{\text{out}} = E_{\text{in}} \times \cos\left(\frac{\theta}{2}\right) e^{j\left(\frac{\theta}{2}\right)} e^{j\varphi}, \tag{6}$$

where E_{in} is the input optical field intensity, E_{out} is the output optical field intensity, θ is the phase shift introduced by the PS1, which acts as the modulation phase, and φ is the phase shift introduced by the PS2, used for phase compensation. To achieve this, the second phase shifter PS2 is configured to dynamically compensate for the phase term. In experimental practice, this compensation can be approximately implemented using a simplified linear relationship: $\varphi \approx -\theta/2$. This approximation is valid over the range $\theta \in [0, \pi]$ and enables the phase compensation to be implemented with low computational complexity or via a lookup table.

In the present implementation, the real-valued features extracted by the EDNN are normalized to the range [0, 1] before optical loading. Therefore, the amplitude term $cos(\theta/2)$ is used to encode non-negative normalized feature values rather than arbitrary signed values. The complex-valued transformation of the MZI mesh is then learned through the trainable phase parameters of the optical system. Thus, the input encoding provides non-negative real-valued amplitudes, while the MZI mesh supplies the complex-valued weights and phase mixing during optical propagation. For general signed real-valued features, the same framework can be extended by using offset normalization, dual-rail positive/negative encoding, or phase-sign encoding, where the magnitude is loaded into the optical amplitude and the sign can be represented by an additional $0/\pi$ phase shift.

C. Dataset Processing and Neural Network Training

In our numerical experiment, we utilized three standard classification datasets: CIFAR-10, the MNIST dataset, and the Fashion-MNIST dataset, along with the DAS dataset. We randomly selected a 70:30 split for training and testing data, so the sample sizes for each class in the training and testing sets are not exactly balanced. Each dataset contains 3000 test samples. We focused more on evaluating the classification performance on the test set. The training process can be found in Section S5 of Supplement 1.

For the DAS dataset, we construct an EDNN with a 512-256-4 architecture and a corresponding ONN with the same layer configuration. The 512-256 layer in the ONN corresponds to an optical parameter count of $(512/4) \times 15$ due to the underlying 4×4 MZI structure. For the Handwriting-MNIST, we build an ONN with a 512-256-64-10 architecture. In this case, the final output layer utilizes three 64-dimensional modules, each producing a four-dimensional output, which are then combined to form the 10-class output. For the Fashion-MNIST dataset, we adopt a deeper ONN with a 512-256-128-64-10 structure. For the CIFAR-10 dataset, the ONN model is configured as 512-512-256-128-64-10. Moreover, we evaluated the performance of the EDNN and DGONN models with varying depths (from two to nine layers) on CIFAR-10, as shown in Fig. S11 of Supplement 1.

For the on-chip experiments on Fashion-MNIST and FSDD, we constructed a dataset of 1200 samples for each task, with 700 samples used for training and 500 for testing, following the same preprocessing pipeline described in Section S6 of Supplement 1. The EDNN first extracts a 64-dimensional real-valued feature vector for each input sample, which is directly used as the input to the optical student network.

The optical neural network adopts a three-layer blockwise MZI architecture whose layer widths are 64-32-16-12-10, analogous to the notation used in the numerical studies. Specifically, the 64-dimensional input is partitioned into 16 four-dimensional blocks, each processed by a 4×4 MZI to yield a 32-dimensional representation. The second optical layer reuses eight 4×4 MZI subnetworks to reduce the dimensionality from 32 to 16. A 12-dimensional subspace is then selected from the 16-dimensional intermediate vector. The final optical transformation is implemented by three parallel 4×4 MZI subnetworks, whose outputs are concatenated to produce a 12-dimensional embedding; the first ten components are used as the logits for the 10-class classification task.

Real-valued features are loaded using the PS1/PS2 modulation scheme described in Section 4, enabling amplitude encoding with phase-restoration compensation. During training, the EDNN supplies layer-wise distilled targets for each of the three optical layers, and all photonic parameters are updated via forward-difference estimation using experimentally measured optical intensities. This ensures that the physical three-layer optical network follows the same distillation-guided optimization pipeline as the numerical DGONN.

Although the DGONN avoids chain-rule-based optical backpropagation, it still relies on forward-difference gradient estimation for tunable photonic parameters. A 4×4 MZI subnetwork with 15 trainable phase parameters requires 16 optical forward evaluations for one local update. Compared with the NDNN, which evaluates each perturbation through the downstream network using the final classification loss, the DGONN uses layer-wise distillation losses to shorten the effective sequential optical computation path and enable layer-wise parallel scheduling. Detailed evaluation counts, detector-limited training-time estimates, and noise-sensitivity analyses are provided in Section S8 of Supplement 1.

Funding. National Natural Science Foundation of China (U2001601, 62175100, 62175103, 61775094, 62341408); Fundamental Research Funds for the Central Universities (2024300421, 2024300447, 0213-14380264, 0213-14380265); Jiangsu Innovation Teams and AI & AI for Science Project of Nanjing University.

Acknowledgment. Yixin Zhang, Wei Jiang, and Ningmu Zou are co-corresponding authors.

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request. The software code for the DRL model and experimental control may be found online in [44].

Supplemental document. See Supplement 1 for supporting content.

REFERENCES

1. OpenAI, J. Achiam, S. Adler, L. Ahmad, *et al.*, "GPT-4 technical report," arXiv (2024).
2. J. Shalf, "The future of computing beyond Moore's Law," Phil. Trans. R. Soc. A **378**, 20190061 (2020).
3. M. Wei, J. Li, Z. Chen, *et al.*, "Electrically programmable phase-change photonic memory for optical neural networks with nanoseconds *in situ* training capability," Adv. Photonics **5**, 046004 (2023).
4. X. Guo, T. D. Barrett, Z. M. Wang, *et al.*, "Backpropagation through nonlinear units for the all-optical training of neural networks," Photonics Res. **9**, B71–B80 (2021).
5. S. Chen, Y. Li, Y. Wang, *et al.*, "Optical generative models," Nature **644**, 903–911 (2025).
6. L. G. Wright, T. Onodera, M. M. Stein, *et al.*, "Deep physical neural networks trained with backpropagation," Nature **601**, 549–555 (2022).
7. Z. Xu, T. Zhou, M. Ma, *et al.*, "Large-scale photonic chiplet Taichi empowers 160-TOPS/W artificial general intelligence," Science **384**, 202–209 (2024).
8. S. Hua, E. Divita, S. Yu, *et al.*, "An integrated large-scale photonic accelerator with ultralow latency," Nature **640**, 361–367 (2025).
9. J. Feldmann, N. Youngblood, M. Karpov, *et al.*, "Parallel convolutional processing using an integrated photonic tensor core," Nature **589**, 52–58 (2021).
10. J. Cheng, Y. Zhao, W. Zhang, *et al.*, "A small microring array that performs large complex-valued matrix-vector multiplication," Front. Optoelectron. **15**, 15 (2022).
11. S. R. Ahmed, R. Baghdadi, M. Bernadskiy, *et al.*, "Universal photonic artificial intelligence acceleration," Nature **640**, 368–374 (2025).
12. T. Xu, W. Zhang, J. Zhang, *et al.*, "Control-free and efficient integrated photonic neural networks via hardware-aware training and pruning," Optica **11**, 1039–1049 (2024).
13. Y. Wan, X. Liu, G. Wu, *et al.*, "Efficient stochastic parallel gradient descent training for on-chip optical processor," Opto-Electron. Adv. **7**, 230182 (2024).
14. Z. Huang, C. Wang, C. Zhang, *et al.*, "Brain-like training of a pre-sensor optical neural network with a backpropagation-free algorithm," Photonics Res. **13**, 915–923 (2025).
15. T. Zhou, L. Fang, T. Yan, *et al.*, "In situ optical backpropagation training of diffractive optical neural networks," Photonics Res. **8**, 940–953 (2020).
16. J. Spall, X. Guo, and A. I. Lvovsky, "Training neural networks with end-to-end optical backpropagation," Adv. Photonics **7**, 016004 (2025).
17. A. Momeni, B. Rahmani, M. Malléjac, *et al.*, "Backpropagation-free training of deep physical neural networks," Science **382**, 1297–1303 (2023).
18. S. Pai, Z. Sun, T. W. Hughes, *et al.*, "Experimentally realized *in situ* backpropagation for deep learning in photonic neural networks," Science **380**, 398–404 (2023).
19. Z. Xue, T. Zhou, Z. Xu, *et al.*, "Fully forward mode training for optical neural networks," Nature **632**, 280–286 (2024).
20. J. Xiang, S. Colburn, A. Majumdar, *et al.*, "Knowledge distillation circumvents nonlinearity for optical convolutional neural networks," Appl. Opt. **61**, 2173–2183 (2022).
21. T. Fang, J. Li, X. Zhang, *et al.*, "Classification accuracy improvement of the optical diffractive deep neural network by employing a knowledge distillation and stochastic gradient descent β-Lasso joint training framework," Opt. Express **29**, 44264–44274 (2021).
22. G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," arXiv (2015).
23. S. Xu, C. Peng, J. Long, *et al.*, "Harnessing negative signals: reinforcement distillation from teacher data for LLM reasoning," arXiv (2025).
24. T. Zhou, X. Lin, J. Wu, *et al.*, "Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit," Nat. Photonics **15**, 367–373 (2021).
25. T. Zhou, W. Wu, J. Zhang, *et al.*, "Ultrafast dynamic machine vision with spatiotemporal photonic computing," Sci. Adv. **9**, eadg4391 (2023).
26. Q. Yan, H. Ouyang, Z. Tao, *et al.*, "Multi-wavelength optical information processing with deep reinforcement learning," Light Sci. Appl. **14**, 160 (2025).
27. J. Schulman, F. Wolski, P. Dhariwal, *et al.*, "Proximal policy optimization algorithms," arXiv (2017).
28. A. Krizhevsky, "Learning multiple layers of features from tiny images," (2009).
29. H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," arXiv (2017).
30. L. Deng, "The MNIST database of handwritten digit images for machine learning research [best of the web]," IEEE Signal Process. Mag. **29**(6), 141–142 (2012).
31. Y. LeCun, B. Boser, J. S. Denker, *et al.*, "Backpropagation applied to handwritten zip code recognition," Neural Comput. **1**, 541–551 (1989).
32. F. Yu, K. Di, W. Chen, *et al.*, "Time-wavelength multiplexed photonic neural network accelerator for distributed acoustic sensing systems," Adv. Photonics **7**, 026008 (2025).
33. Jakobovski, "Free spoken digits dataset," GitHub (2017). https://github.com/Jakobovski/free-spoken-digit-dataset.
34. Y. Tian, M. A. Chao, C. Kulkarni, *et al.*, "Real-time model calibration with deep reinforcement learning," Mech. Syst. Signal Process. **165**, 108284 (2022).

35. H. H. Zhu, J. Zou, H. Zhang, *et al.*, "Space-efficient optical computing with an integrated chip diffractive neural network," Nat. Commun. **13**, 1044 (2022).
36. B. Bai, Q. Yang, H. Shu, *et al.*, "Microcomb-based integrated photonic processing unit," Nat. Commun. **14**, 66 (2023).
37. F. Ashtiani, A. J. Geers, and F. Aflatouni, "An on-chip photonic deep neural network for image classification," Nature **606**, 501–506 (2022).
38. NVIDIA, "Nvidia jetson nano".
39. H. Zhang, M. Gu, X. D. Jiang, *et al.*, "An optical neural chip for implementing complex-valued neural network," Nat. Commun. **12**, 457 (2021).
40. C. Wu, H. Yu, S. Lee, *et al.*, "Programmable phase-change metasurfaces on waveguides for multimode photonic convolutional neural network," Nat. Commun. **12**, 96 (2021).
41. X. Yu, Z. Wei, F. Sha, *et al.*, "Parallel optical computing capable of 100-wavelength multiplexing," eLight **5**, 10 (2025).
42. K. Li, D. Thomson, J. Zhu, *et al.*, "P-N dual-drive scheme enabling silicon photonics modulators operating at 200 GBaud," Optica **12**, 841–842 (2025).
43. S. Xiang, Y. Chen, H. Zhao, *et al.*, "Nonlinear photonic neuromorphic chips for spiking reinforcement learning," Optica **13**, 457–468 (2026).
44. N. Zou and S. Chen, "Photonic device calibration with PPO," GitHub, 2025, https://github.com/zouningmu/Photonic_Device_Calibration_with_PPO.