SqueezeLight: A Multi-Operand Ring-Based Optical Neural Network with Cross-Layer Scalability

Jiaqi Gu *Graduate Student Member, IEEE*, Chenghao Feng *Graduate Student Member, IEEE*, Hanqing Zhu, Zheng Zhao, Zhoufeng Ying *Member, IEEE*, Mingjie Liu *Graduate Student Member, IEEE*, Ray T. Chen *Fellow, IEEE*, and David Z. Pan *Fellow, IEEE*

***Abstract*— Optical neural networks (ONNs) are promising hardware platforms for next-generation artificial intelligence acceleration with ultra-fast speed and low energy consumption. However, previous ONN designs are bounded by one multiply-accumulate operation per device, showing unsatisfying scalability. In this work, we propose a scalable ONN architecture, dubbed `SqueezeLight`. We propose a nonlinear optical neuron based on multi-operand ring resonators (MORRs) to squeeze vector dot-product into a single device with low wavelength usage and built-in nonlinearity. A block-level squeezing technique with structured sparsity is exploited to support higher scalability. We adopt a robustness-aware training algorithm to guarantee variation tolerance. To enable a truly scalable ONN architecture, we extend `SqueezeLight` to a separable optical CNN architecture that further *squeezes in the layer level*. Two orthogonal convolutional layers are mapped to one MORR array, leading to order-of-magnitude higher software training scalability. We further explore augmented representability for `SqueezeLight` by introducing parametric MORR neurons with trainable nonlinearity, together with a nonlinearity-aware initialization method to stabilize convergence. Experimental results show that `SqueezeLight` achieves one-order-of-magnitude better compactness and efficiency than previous designs with high fidelity, trainability, and robustness. Our open-source codes are available at github.com/JeremieMelo/SqueezeLight.**

***Index Terms*—Nanophotonics, neural network hardware, optical computing, scalability, multi-operand micro-ring.**

I. Introduction

DEEP neural networks (DNNs) have demonstrated superior performance on various machine learning tasks. However, the escalating computation demands of DNNs cast substantial challenges for traditional electrical digital computers in the post-Moore's era. With the advances in silicon photonics, optical neural networks (ONNs) demonstrate compelling potentials for neurocomputing with the intrinsic high parallelism and speed of light [1]–[4]. Prior ONNs based on Mach-Zehnder interferometers (MZIs) have been successfully demonstrated to achieve matrix multiplication using singular value decomposition [1]. A slimmed ONN [5] was proposed to cut down the area cost via a co-design methodology. Butterfly-style ONNs [6]–[8] demonstrated a more compact design for neurocomputing in the general frequency domain. Besides coherent ONNs, incoherent ONNs have been explored to reduce area cost using micro-ring resonators (MRRs). MRR-based ONNs [9], [10] have been demonstrated to implement weight matrices in MRR weight banks leveraging wavelength-division multiplexing (WDM) techniques.

However, prior state-of-the-art (SoTA) ONN designs still encounter scalability issues in terms of high area cost. Though MRR-based ONN is considered one of the most compact ONNs given the small MRR device sizes [2], [9]–[12], it reaches the current area lower bound, i.e., one optical device per multiply-accumulate (MAC) operation. It is technically challenging to further compactness improvement by using traditional MRRs. Moreover, the high usage of wavelength limits the scalability of MRR-ONNs since practical weight matrix dimensions are far beyond the maximum wavelengths supported by modern dense WDM (DWDM) techniques, leading to unsatisfying throughput due to weight bank reuse [12]. MRR-ONNs also encounter robustness concerns under various noises and variations [12].

To break the current area lower bound of integrated ONNs, in this work, we propose a novel ONN architecture that squeezes matrix operations into arrays of ultra-compact multi-operand micro-ring resonators (MORRs), dubbed `SqueezeLight`, to enable scalable, efficient, and robust optical neurocomputing. We extend `SqueezeLight` to a separable optical CNN architecture with trainable MORR nonlinearity, showing augmented expressiveness and order-of-magnitude higher software training scalability than the original MORR-based CNN. The main contributions are as follows,

- **Scalability**: we explore the analog usage of multi-operand ring resonators to construct an ultra-compact ONN architecture with built-in nonlinearity, surpassing prior integrated ONNs by one order of magnitude in footprint.
- **Efficiency**: we employ fine-grained structured pruning in `SqueezeLight` for a quadratic efficiency boost.
- **Robustness**: we propose a sensitivity-aware learning technique to overcome thermal crosstalk and device vari-

This work was supported in part by the Multidisciplinary University Research Initiative Program through the Air Force Office of Scientific Research under Contract FA 9550-17-1-0071, monitored by Dr. Gernot S. Pomrenke. The preliminary version has been presented at the IEEE Design, Automation & Test in Europe Conference (DATE) in 2021.

Jiaqi Gu, Chenghao Feng, Hanqing Zhu, Mingjie Liu, Ray T. Chen and David Z. Pan are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, TX, USA (e-mail: jqgu@utexas.edu). Zheng Zhao is with Synopsys Inc., CA, USA.
Zhoufeng Ying is with Alpine Optoelectronics, silicon photonics department, CA 94538, USA.

ations to improve noise resilience.

- **Trainability**: We extend our DATE version of `SqueezeLight` [13] to a novel separable optical CNN architecture with order-of-magnitude higher training scalability to support million-parameter ONNs.
- **Expressiveness**: We explore parametric MORR neurons with trainable nonlinearity to fortify the advantages of built-in nonlinearity of `SuqeezeLight`, leading to an average of +2.1% accuracy improvement on various vision recognition tasks.

The remainder of this paper is organized as follows. Section II introduces the background for ONN architectures and multi-operand micro-ring resonators. Section III illustrates details about `SqueezeLight` with efficiency and robustness optimization techniques. Section IV analyzes and compares our hardware cost and features with previous ONN architecture designs. Section V demonstrates an extension to an MORR-based separable optical CNN with augmented training scalability and expressiveness. Section VI shows the optical simulation and reports the experimental results for `SqueezeLight`, followed by the conclusion in Section VII.

II. Preliminaries

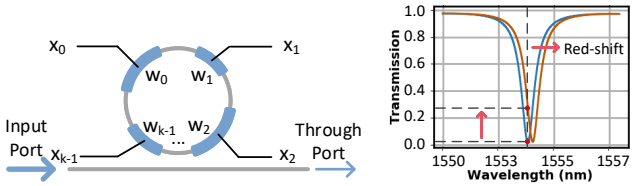
This section introduces the background knowledge of ONNs and our motivations.

A. Various Neural Network Designs

Convolutional neural networks (CNNs) learn discriminative representation via convolution-based linear operations. Kernelized NNs [14] have shown competitive performance by replacing convolutions with nonlinear projection kernels. Various linear and nonlinear convolution variants with better efficiency and robustness have been proposed, e.g., hyperbolic tangent convolution [15] and *AdderNet* [16]. In this work, we leverage the analog computing power of multi-operand ring resonators to construct compact optical neurons with built-in nonlinearity, achieving scalable optical neurocomputing with competitive model expressiveness.

B. Optical Neural Architectures

Recently, ONN architectures have been rapidly evolving [1], [2], [5]–[7], [9]–[11], [17], [18]. Coherent ONNs have been demonstrated to implement computation-intensive general matrix multiplication (GEMM) for ultra-fast NN inference, e.g., MZI-based ONNs [1], [5] and FFT-based ONNs [6]–[8]. Incoherent ONNs push the limits in circuit footprint by using MRR weight banks to implement matrices [9], [10]. However, the scalability of MRR-based ONNs is inevitably limited by the size of MRR weight banks and the high usage of wavelength. To break through the ONN scalability bound, in this work, we propose a more compact ONN architecture `SqueezeLight` with a lower device and wavelength usage.



(a) (b)

Fig. 1: (a) All-pass k-operand MORR. (b) Through port light intensity transmission of an all-pass MORR.

C. Multi-Operand Ring Resonators

A multi-operand logic gate (MOLG) has been experimentally demonstrated to achieve multi-operand Boolean functions on a single MRR, achieving ultra-compact optical digital computing [19]. Figure 1(a) shows the structure of an all-pass multi-operand ring resonator (MORR). Unlike the traditional MORR with a single controller, an MORR has k active phase shifters independently controlled by k electrical signals x, each creating a phase shift $\phi_i(x_i)$. k phase shifts are accumulated $\phi = \sum_i \phi_i(x_i)$ and lead to a spectrum redshift $\Delta\lambda$, such that the transmitted light intensity on the through port changes accordingly. The transmission spectrum of an MORR is demonstrated in Fig. 1(b). A k-operand all-pass MORR has the following transfer function,

$$y = f(\phi) = \left| \frac{r - ae^{-j\phi}}{1 - rae^{-j\phi}} \right|^2 d, \ \phi = \sum_{i=0}^{k-1} \phi_i(x_i), \ \phi_i(x_i) \propto w_i x_i^2, \tag{1}$$

where x_i is the electrical input voltage, $\phi_i(\cdot)$ is the phase shift response curve of the actuator, ϕ is the accumulated round-trip phase shift of the MORR, r and a are self-coupling coefficient and single-pass amplitude transmission factor, and $d, y \in [0, 1]$ are the light intensity on the input port and through port, respectively. The weight w_i on the i-th input can be encoded into different actuator arm lengths, different material properties, different input ranges, reconfigurable controller resistances, etc [19]. Instead of using MORR as a digital logic gate [19], we explore the *analog* usage of MORRs for optical neuromorphic computing.

III. Proposed Optical Neural Network Architecture

In this section, we present design details on the proposed `SqueezeLight` shown in Fig. 2 and introduce essential techniques for scalability and efficiency improvement. We summarize key notations for `SqueezeLight` in Table I.

A. MORR-based Nonlinear Neuron

Different from the prior GEMM-based ONN design concept that only focuses on universal linear operations, we target unique nonlinear optical neurocomputing based on an ultra-compact MORR device. Recall that in Eq. (1), we can *squeeze length-k dot-product into the round-trip phase shift of a single MORR*. This dot-product result will be *probed* by the input light signal and activated by the MORR nonlinear transmission

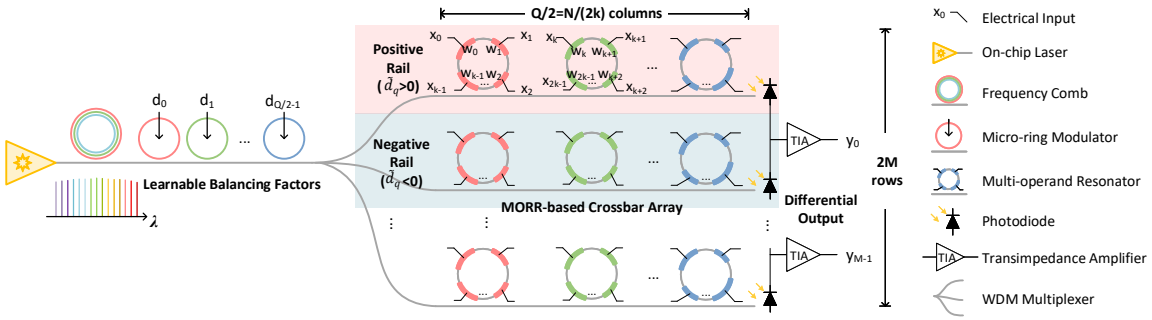


Fig. 2: Proposed MORR-based ONN architecture `SqueezeLight` with learnable neuron balancing.

TABLE I: Notations used in `SqueezeLight`.

$M \times N$	Matrix dimensions
$P \times Q$	Grid dimensions in blocking
k	Block size
k_{max}	Max #operands in an MORR
k'	#Non-zero weights per row after pruning
$w/\boldsymbol{W}$	Weights/weight matrix
x	Input signals
ϕ	Round-trip phase shift
$\hat{\phi}$	Round-trip phase shift after crosstalk
$\Delta\phi$	Phase noise
$f(\cdot)$	Nonlinear $y-\phi$ transmission
d	Learnable balancing factors
G	TIA gain
$\tilde{d}$	Balancing factor that absorbs G
$\gamma/\boldsymbol{\Gamma}$	Intra-MORR crosstalk coupling factor/matrix

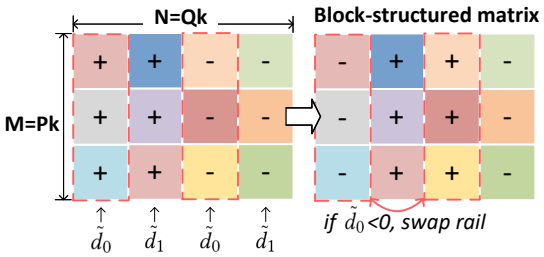


Fig. 3: Block-structured matrices with learnable balancing factors.

curve. The idle device will be initially calibrated to the on-resonance state, where the transmitted light intensity reaches the minimum. Then, k input voltage control signals will jointly create a phase shift to modulate the input light intensity. Hence, we model the MORR neuron as,

$$y = f(\sum_{i=0}^{k-1}\phi_i)d \propto f\Big(\sum_{i=0}^{k-1} w_i x_i^2\Big)d, \quad \text{s.t. } \mathrm{w_i} \geq 0 \qquad (2)$$

where $f(\cdot)$ is the nonlinear $y-\phi$ transmission curve. Note that we are justified to assume all MORR nonlinear curves are identical because the shape of $f(\cdot)$ keeps almost the same within the practical wavelength range [20].

B. `SqueezeLight` Architecture

Based on the above MORR neuron, we propose a novel ONN architecture `SqueezeLight` shown in Fig. 2. We assume to map an $M \times N$ weight matrix onto this MORR array. The matrix is partitioned into $P \times Q$ sub-matrices with size of $k \times k$, where $P = \lceil (M/k), Q = \lceil (N/k) \rceil \rceil$. `SqueezeLight` starts with an on-chip frequency comb to generate multiple wavelengths $(\lambda_0, \lambda_1, \cdots)$. Then, narrow-band MRRs are placed as wavelength-specific modulators $\boldsymbol{D} = (d_0, \cdots, d_{Q/2-1}) \in [0,1]$ to achieve an adaptive dynamic MORR transmission range. Modulated probing light signals are evenly distributed into $2M$ rows. By placing a series of MORRs to form an array, we can implement a nonlinear ONN layer. Theoretically, we need total $2M$ rows and $\frac{Q}{2} = \frac{N}{2k}$ columns to implement an $M \times N$ weight matrix $\boldsymbol{W}$. The q-th MORR in one row will resonate at the wavelength λ_q and apply projection on a segment of length-k vector as $y_q = f(\sum_{i=0}^{k-1} w_{qi} x_{qi}^2) d_q$. At the end of the m-th row, a photo-detector will detect accumulated light intensity as $I_m = \sum_{q=0}^{Q/2-1} y_{mq}$.

Differential Detection for Full-range Weights. Typically, limited by the physical implementation, the weights are restricted to be non-negative, which could limit the model representability. Hence we introduce a differential structure for full-range outputs and augment the expressiveness with learnable neuron balancing factors shown in Fig. 2. One row is halved into two adjacent rows as the positive rail I^+ and negative rail I^- respectively. The differential photo-current structure at the end enables full-range of outputs, equivalently forcing half weights, i.e., weights on rail I^-, to be non-positive values,

$$y_m = G(I_m^+ - I_m^-) = G\Big(\sum_{q=0}^{Q/2-1} y_{mq} - \sum_{q=Q/2-1}^{Q-1} y_{mq}\Big), \qquad (3)$$

where G is the gain of the transimpedance amplifier (TIA), which can be used to extend the signal range. A direct benefit from this differential structure is that we can save 50% of wavelength usage by partitioning one width-Q row into two width-$\frac{Q}{2}$ rails.

Learnable Balancing Factors. With d=1, all MORRs are treated with the same importance as they have the same dynamic range $y_{mq} \in [0,1], \forall q$, which loses the degree of freedom to assign different weights to different partial product results. To resolve this, we allow learnable MORR balancing factors $\tilde{\boldsymbol{D}} = \{\tilde{d}_q | \tilde{d}_q \in [-G_{max}, G_{max}], \tilde{d}_q = \tilde{d}_{q\ (mod)\ \frac{Q}{2}},\ q \in [0, Q-1]\}$ and encode them in the MRRs at the beginning. Note that the maximum TIA gain G_{max}

expands the implementable range to $\tilde{d} \in [-G_{max}, G_{max}]$. A column of MORRs share the same balancing factor as they share the same wavelength. Hence the MORR neuron is augmented as follows,

$$y_m = \sum_{q=0}^{Q-1} f\Big(\sum_{i=0}^{k-1} w_{mqi} x_{qi}^2\Big)\tilde{d}_q. \tag{4}$$

A natural question is how we can achieve full-range balancing factors as all-pass MRRs can only achieve non-negative transmission modulation. It turns out that by simply swapping two MORRs on the opposite rails, one can equivalently realize a negative factor $\tilde{d} < 0$ as shown in Fig. 3. This technique enables a learnable output range for different MORRs and thus boosts the expressiveness of `SqueezeLight`.

C. Peripheral Units

We briefly discuss peripheral units, with all system-level details being omitted since advanced system-level and architectural innovations in photonic NN accelerators [10], [11] are mostly applicable to `SqueezeLight` as well.

1) Normalization: Normalization operations, e.g., BatchNorm, can be implemented by the TIA gain and voltage signal offset with negligible latency overhead.

2) Nonlinear Activation: Since MORR-based neurons have built-in nonlinearity, extra electrical activations are not required.

3) Electrical Dataflow: The input signals/weights are loaded from high-bandwidth SRAM or ultra-fast photonic racetrack memory banks [21] and converted to analog signals through electrical digital-to-analog converters (DACs). The photo-currents are amplified by TIAs. Direct optical-electrical-optical (O-E-O) conversions will be used to cascade ONN layers without voltage-to-transmission encoding.

D. Area Reduction via Block-Squeezing

Thanks to the MORR device, we can squeeze a vector dot-product into one micro-ring. To achieve a quadratically more compact design, we further squeeze a matrix into one MORR via a block-squeezing method. Inspired by structured neural networks that restrict the weight matrix structure [6], [22] for better efficiency, we introduce this concept to `SqueezeLight` for higher compactness. An $M \times N$ block-structured matrix $\boldsymbol{W}$ contains $P \times Q$ square sub-matrices $\{w_{pq}\}_{p,q=0}^{P,Q}$, each being a $k \times k$ structured matrix. We use a circulant matrix as an example [22], where each column is essentially the circular shift of its length-k primary vector on the first column. Due to row-wise parameter sharing, the sub-matrix multiplication $\boldsymbol{w}_{pq} \cdot x_q$ can be efficiently *squeezed* into one k-operand MORR. Figure 4 visualizes the mapping from a 4×4 structured sub-matrix to an MORR. At time step t=0, we implement the first row. Then we shift the inputs around the ring to align with corresponding weights on the second row and repeat this process. After k time steps with input rotation, we reuse the same MORR and finish an entire circulant matrix multiplication. In this way, we successfully achieve $O(k^2)$ times device usage reduction and save k times wavelength usage.

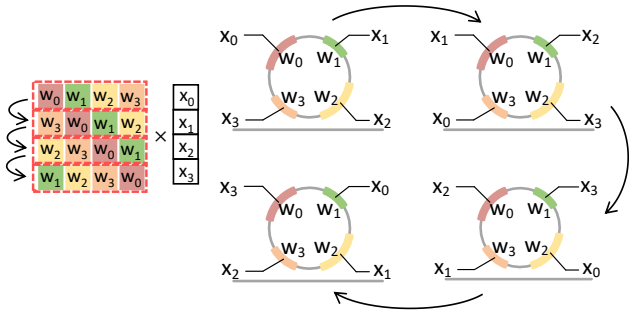


Fig. 4: Squeezing a 4×4 block into one MORR using 4 cycles. The right part unfolds the *input rotation* mechanism temporally in 4 cycles on a single MORR.

E. Sparsity Exploration via Fine-Grained Structured Pruning

For an $M \times N$ block-structured matrix, the total component usage adds up to $\frac{N}{2}$ MRRs and $(\frac{MN}{k^2})$ k-operand MORRs. Given fixed M and N, a larger k means fewer blocks and less MORR usage. However, implementing MORRs with too many actuators can be challenging in practice. If one has a tight device usage budget, it is preferable to use a large sub-matrix that could exceed the MORR capacity, i.e., $k > k_{max}$. To overcome this, we prune each sub-matrix with a fine-grained structured sparsity. In Fig. 5, less important entries in the primary vector are forced to zero, leaving k' non-zero weights. The same sparsity pattern will be automatically imposed on other columns according to the pre-defined matrix structure. Our block-squeezing technique allows mapping the pruned sparse block with $k' \leq k_{max}$ into one MORR to maintain the highest compactness. We adopt a two-stage pruning procedure with learning rate rewinding to train `SqueezeLight` with structured sparsity, described in Alg. 1. We first pre-train and prune the weights with a target sparsity. Then we re-train the model from scratch with a rewound learning rate to achieve better accuracy than traditional post-training fine-tuning.

F. Robustness Boost via Sensitivity-Aware Optimization

For analog computing, noise robustness is considered a practical concern [1], [12], [17], [23]–[25]. For MORRs, we consider random phase variations and intra-MORR crosstalk as the main non-ideal effects. The random variations can be estimated as a Gaussian noise on the phase shift $\Delta\phi \in \mathcal{N}(0, \sigma^2)$. We formulate the dynamic intra-MORR crosstalk among k actuators as $\hat{\boldsymbol{\Phi}} = \boldsymbol{\Gamma} \cdot \boldsymbol{\Phi}$ governed by a coupling matrix $\boldsymbol{\Gamma}$,

$$\begin{pmatrix} \hat{\phi}_0 \\ \hat{\phi}_1 \\ \cdots \\ \hat{\phi}_{k-1} \end{pmatrix} = \begin{pmatrix} \gamma_{0,0} & \gamma_{0,1} & \cdots & \gamma_{0,k-1} \\ \gamma_{1,0} & \gamma_{1,1} & \cdots & \gamma_{1,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{k-1,0} & \gamma_{k-1,1} & \cdots & \gamma_{k-1,k-1} \end{pmatrix} \begin{pmatrix} \phi_0 \\ \phi_1 \\ \cdots \\ \phi_{k-1} \end{pmatrix}, \tag{5}$$

Note that the crosstalk effect $|\hat{\phi}_0 - \phi_0|$ is dynamically determined by the weight w and input x, but the coupling matrix

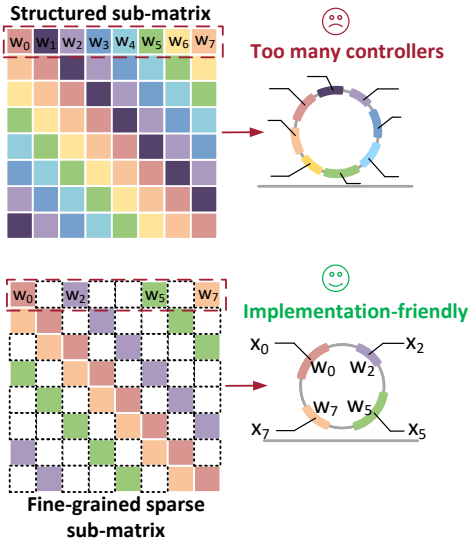


Fig. 5: Fine-grained pruning enables squeezing a 8×8 structured block into a 4-op MORR.

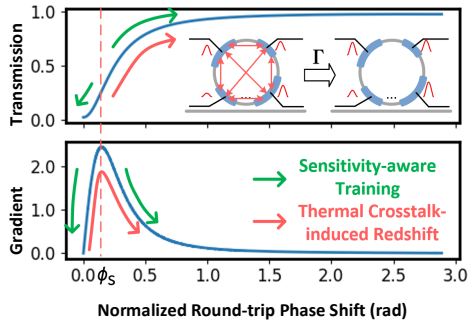


Fig. 6: Transmission curve f and its gradient $\nabla_\phi f$ with thermal crosstalk and sensitivity-aware training.

$\mathbf{\Gamma}$ is constant after manufacturing. The self-coupling factor $\gamma_{i,i} = 1$ and all mutual coupling factors $\gamma_{i,j}$ are basically determined by the spacing among phase shifters [26]. Hence we assume that they share the same value γ. We found that intra-MORR crosstalk is equivalent to a constant scaling factor on ϕ as follows,

$$\hat{y}_m = \sum_{q=0}^{Q-1} f\Big((1+(k'-1)\gamma)\phi_{mq} + \Delta\phi\Big)\tilde{d}_q. \tag{6}$$

This equation implies that pruning can reduce the crosstalk noise since only the left k' actuators have crosstalk after pruning. To better understand the sensitivity of MORR neurons to crosstalk, we show the transmission curve in Figure 6. We observe that the transmission curve $f(\cdot)$ has different sensitivity (gradient) at different wavelengths. Crosstalk effects induce an extra redshift in the spectrum, forcing all $\phi < \phi_s$ to have higher sensitivity and $\phi \geq \phi_s$ to have less sensitivity. Based on this observation, we introduce a sensitivity-aware optimization method to improve the robustness of `SqueezeLight`, shown in Alg. 1. We adopt the following the objective to train an

Algorithm 1 Training algorithm of `SqueezeLight` with fine-grained structured pruning and sensitivity-aware optimization.

Input: Initial weights $\boldsymbol{W}^0 \in \mathbb{R}^{P\times Q\times k}$ and $\tilde{\boldsymbol{D}}^0 \in \mathcal{R}^{Q/2}$, pruning percentage $T = 1-\frac{k'}{k}$, pretraining step t_{pre}, initial step size η^0, decay factor β, penalty weight α, variation $\Delta\phi$, and crosstalk coupling matrix $\mathbf{\Gamma}$;
Output: Converged $\boldsymbol{w}^t$, $\boldsymbol{d}^t$, and a pruning mask $\mathcal{M} \in \mathbb{Z}^{P\times Q\times k}$;

for $t \leftarrow 1, \cdots, t_{pre}$ **do** ▷ Stage 1: Pretraining
  $\mathcal{L} \leftarrow \mathcal{L}_0^t(x; \boldsymbol{W}^{t-1}, \tilde{\boldsymbol{D}}^{t-1})$
  $(\boldsymbol{W}^t, \tilde{\boldsymbol{D}}^t) \leftarrow (\boldsymbol{W}^{t-1}, \tilde{\boldsymbol{D}}^{t-1}) - \eta^{t-1}(\nabla_{\boldsymbol{W}}\mathcal{L}, \nabla_{\tilde{\boldsymbol{D}}}\mathcal{L})$
  $\eta^t \leftarrow \eta^{t-1}\beta$ ▷ Learning rate decay
$\eta^t \leftarrow \eta^0, \mathcal{M} \leftarrow 1$ ▷ Learning rate rewinding and initialize mask
for all $\boldsymbol{W}^t_{pqi} \in \boldsymbol{W}^t$ **do**
  if $\boldsymbol{W}^t_{pqi} < \texttt{percentile}(\boldsymbol{W}^t_{pq}, T)$ **then**
    $\mathcal{M}_{pqi} \leftarrow 0$ ▷ Compute pruning mask
while not converged **do** ▷ Stage 2: Fine-grained pruning
  $\mathcal{L} \leftarrow \mathcal{L}_0^t(x; \mathcal{M}\odot\boldsymbol{W}^{t-1}, \tilde{\boldsymbol{D}}^{t-1}, \mathbf{\Gamma}, \Delta\phi) + \alpha\mathcal{L}_S(\mathbf{\Gamma}, \Delta\phi)$ ▷ Sensitivity-aware regularization
  $(\boldsymbol{W}^t, \tilde{\boldsymbol{D}}^t) \leftarrow (\boldsymbol{W}^{t-1}, \tilde{\boldsymbol{D}}^{t-1}) - \eta^{t-1}(\nabla_{\boldsymbol{W}}\mathcal{L}, \nabla_{\tilde{\boldsymbol{D}}}\mathcal{L})$
  $\eta^t \leftarrow \eta^{t-1}\beta$ ▷ Learning rate decay

L-layer `SqueezeLight`,

$$\mathcal{L} = \mathcal{L}_0(x; \boldsymbol{W}, \tilde{\boldsymbol{D}}, \mathbf{\Gamma}, \Delta\phi) + \alpha \sum_{l,m,q=0}^{L-1,M-1,Q-1} \nabla_\phi f(\hat{\phi}_{lmq} + \Delta\phi), \tag{7}$$

where $\mathcal{L}_0(x; \boldsymbol{W}, \tilde{\boldsymbol{D}}, \mathbf{\Gamma}, \Delta\phi)$ is the task-specific loss with noise injection, and the second term, denoted as $\mathcal{L}_S(\mathbf{\Gamma}, \Delta\phi)$, is a sensitivity-aware penalty term weighted by α. This method jointly considers variations and crosstalk with a gradient-based sensitivity penalty, enabling close-to-ideal test accuracy.

IV. Hardware Feasibility and Efficiency

We theoretically analyze the hardware feasibility and efficiency, and qualitatively compare essential features with previous ONNs.

A. MORR Physical Feasibility

Our MORR leverages the analog property of a successfully demonstrated digital MOLG [19]. We discuss how to encode weights and apply inputs to the analog MORR device. We can use high-speed DACs and high-speed E-O controllers to switch the input signals. Weight reprogramming is much less frequent than input signal switching. There are multiple possible approaches to implementing weights as modulation coefficients. If the weights are pre-defined and fixed, we can simply use controller length to encode the weights with zero energy cost in weight encoding. If the weights need a dynamic update, we can implement the weights as power scaling factors on the input signals, e.g., program the electrical attenuation units to modulate the input signals. Low-speed electrical attenuators are enough to handle low-frequency weight reprogramming in most NN workloads.

Note that one may have concerns about the limited finesse of the low-Q MORR we show. This is a proof-of-concept example and not necessarily the most suitable ring design for

SqueezeLight. In later simulation results, we show that our MORR array works well with high-Q MORRs. Simply shrinking the range of round-trip phase shift ϕ, either by scaling down the power of phase tuning signal x or reducing the tuning coefficient w, can create the same $y-\phi$ nonlinear curve as the low-Q MORR. Hence, we do not require a flat MORR spectrum. Instead, MORRs with high quality values and finesse are actually preferred to enable a larger WDM capacity for higher throughput and less spectrum crosstalk.

B. Symbolic Analysis on Area, Latency, and Power

In Table II, our architecture outperforms three coherent ONNs by a large margin [1], [5], [6]. We focus on the comparison with the most compact designs MRR-ONN-1 [10] and MRR-ONN-2 [9] in terms of area cost $\mathcal{A}$, latency τ, and power $\mathcal{P}$. We assume the current DWDM capacity supports maximum B different wavelengths [27], [28].

First, the size and power of an MRR and a k-operand MORR can be assumed the same since they have the same phase tuning range, i.e., half of the resonance curve. Therefore, we focus on the number of resonators in the discussion. We denote the computation efficiency as $\mathcal{E} = (\mathcal{A}\mathcal{P}\tau)^{-1}$. SqueezeLight achieves the following improvement over two MRR-ONNs when the matrix dimension is smaller than the DWDM capacity, i.e., $N < B$,

$$\frac{\mathcal{A}_{ours}}{\mathcal{A}_{prev}} \approx \frac{\mathcal{P}_{ours}}{\mathcal{P}_{prev}} \approx \frac{1}{k^2},\ \frac{\tau_{ours}}{\tau_{prev}} = \frac{k\lceil N/B \rceil}{\lceil N/(2kB) \rceil} = k,\ \frac{\mathcal{E}_{ours}}{\mathcal{E}_{prev}} \approx k^3. \tag{8}$$

Once the matrix width is larger than the maximum number of wavelengths available as $\frac{N}{2k} < B < N$, we can achieve,

$$\frac{\mathcal{A}_{ours}}{\mathcal{A}_{prev}} \approx \frac{\mathcal{P}_{ours}}{\mathcal{P}_{prev}} < \frac{2}{k},\ \frac{\tau_{ours}}{\tau_{prev}} = \frac{k}{\lceil \frac{N}{B} \rceil},\ \frac{\mathcal{E}_{ours}}{\mathcal{E}_{prev}} \approx \frac{Bk^3}{N} > \frac{k^2}{2}. \tag{9}$$

If the weight matrix is even larger, i.e., $B < \frac{N}{2k}$, we have

$$\frac{\mathcal{A}_{ours}}{\mathcal{A}_{prev}} \approx \frac{\mathcal{P}_{ours}}{\mathcal{P}_{prev}} \approx \frac{2}{k},\ \frac{\tau_{ours}}{\tau_{prev}} \approx \frac{1}{2},\ \frac{\mathcal{E}_{ours}}{\mathcal{E}_{prev}} \approx \frac{k^2}{2}, \text{if } B < \frac{N}{2k}. \tag{10}$$

It can be observed that our ONN gains more hardware efficiency advantage as B scales up, thus our scalability grows together with the development of the DWDM technology.

C. Qualitative Feature Comparison

In Table II we compare several key features of 6 ONN designs. Previous ONNs mainly focus on general matrix multiplication and offload the nonlinear activation to the electrical domain. In contrast, our proposed neuron leverages the built-in nonlinearity in MORRs to eliminate the overhead from electrical activation, enabling higher speed and efficiency. In terms of model expressivity, MRR-ONN-1 [10] has a limited solution space with only positive weights, while our designs support full-range weights with augmented representability via learnable balancing factors. SqueezeLight also benefits from lower control complexity and higher efficiency due to direct signal encoding $v_x = x$, while previous MRR-ONNs require additional nonlinear mapping to encode inputs/weights into voltage signals $v_x = \sqrt{\phi^{-1}(f^{-1}(x))}$.

D. Quantitative System Performance Evaluation

We give a more rigorous performance analysis on SqueezeLight and compare it with MRR-ONNs.

Compute Density and Delay. We assume to implement a 256×256 block-structured (k=8) weight matrix. We assume the ring spacing is 60 μm. The 4-op MORR radius is 20 μm, and the MRR radius is 5 μm. The WDM capacity is 16. If the MORR array contains 32×16 4-op MORRs, it takes roughly $32\times16\times100^2\mu m^2$. Given the same footprint budget and same WDM capacity, we can construct a 64×16 MRR weight bank.

Taking into account the delay by modulators (10 ps), photodetectors (10 ps), ADCs (100 ps), and the optical path ($100\mu m \times 16 \times n_g/c = 21.3\ ps$). The total delay of our MORR array is 141.3 ps, which corresponds to an operating frequency of 7 GHz. Every cycle, our MORR array can finish 8192 FLOPs. The compute density of SqueezeLight is $\frac{16\times256\times2\ OPs}{141.3ps\times(32\times16\times100\times100\mu m^2)} = 11.3\ TOPS/mm^2$. It takes SqueezeLight 16 cycles (2.26 ns) to implement the 256×256 matrix.

The latency for the 64×16 MRR weight bank is $10+10+100+(70\mu m\times2\times16\times n_g/c) = 179.7\ ps$, which corresponds to an operating frequency of 5.6 GHz. Therefore, the compute density for MRR-ONN is $\frac{64\times16\times2\ OPs}{179.7ps\times(64\times16\times70\times70\mu m^2)} = 2.3\ TOPS/mm^2$. It takes the MRR-ONN 64 cycles (11.5 ns) to implement this 256×256 weight matrix.

Power. We consider power consumption including laser, 8-bit DAC, 8-bit ADC, ring locking, and ring programming. We use an 8-bit 10 GSPS ADC [32], which consumes 39 mW per channel. Each high-speed microring modulator approximately achieves 18 fJ/bit [30], which corresponds to the power P_{ring} of 0.126 mW under 7 GHz. The static locking power P_{lock} of each ring is around $P_{lock} \approx 0.5P_\pi = 9.75\ mW$ [30], [33]. For high-speed input x modulation, each DAC power is $P_{DAC} = 3.92\ mW$ [34], [35]. Since weight configuration is much less frequent than input signals, typically, the weight DAC dynamic power can be ignored. Based on the detection sensitivity and circuit insertion loss, the laser power [36] is $P_{laser} = \frac{h\nu}{\eta\times\mathtt{IL}}2^{2N_b+1}\times\mathtt{freq.} = 131.62\ mW$, where $h\nu$ is the photon energy at 1550 nm, η is the laser efficiency (0.2), IL is the insertion loss (0.25 dB/ring), and N_b is the resolution (8-bit). The power consumption for a 32×16 MORR array is

$$\begin{aligned}&((32\times16+16)P_{lock} + (32\times16)P_{ring}) + P_{laser}\\ &\quad + 256P_{DAC} + 16P_{ADC}\\ &\approx 5212.5 + 131.62 + 1003.52 + 624.00\ mW\\ &\approx 6.972\ W.\end{aligned} \tag{11}$$

The energy efficiency of the MORR array is $\frac{16\times256\times2\ OPs}{141.3\ ps\times6.972\ W} = 8.32\ TOPS/W$.

For the 64×16 MRR weight bank, P_{ring} is 0.101 mW under 5.6 GHz [30]. Each MRR needs extra weight configuration power [33] $P_w = P_\pi/(2\times\mathtt{finesse}) \approx 0.4875\ mW$, where the finesse is around 20 [30]. The total power is

$$\begin{aligned}&(16P_{ring} + (16\times64)(P_{lock}+P_w)) + P_{laser} + 16P_{DAC} + 64P_{ADC}\\ &\approx 10640.8 + 214.99 + 50.18 + 2496\ mW \approx 13.402\ W.\end{aligned} \tag{12}$$

TABLE II: Symbolic hardware cost and qualitative feature comparison. The matrix is $M \times N$ with size-k blocks. B is the DWDM capacity. For a fair comparison, the device counts are converted to #MRRs based on real device sizes [1], [6], [29]. The area ratio β_a and power ratio β_p between one MZI (240×40 μm^2 [1], ∼48mW [29]) and one MRR (20×20 μm^2,∼10 mW [30]) are β_a=24 and β_p=4.8.

	MZI-ONN [1]	Slim-ONN [5]	FFT-ONN [6]	MRR-ONN-1 [10]	MRR-ONN-2 [9]	SqueezeLight
#MRRs	$\beta_a MN$	$\sim \frac{\beta_a}{2} MN$	$\sim \frac{\beta_a}{4} MN$	$M \min(N, B)$	$M \min(N, B)$	$\frac{2M}{k} \min(\frac{N}{2k}, B)$
#Wavelength	1	1	1	$\min(N, B)$	$\min(N, B)$	$\min(\frac{N}{2k}, B)$
Latency	1	1	1	$\lceil \frac{N}{B} \rceil$	$\lceil \frac{N}{B} \rceil$	$k\lceil \frac{N}{2kB} \rceil$
Power	$\beta_p MN$	$\sim \frac{\beta_p}{2} MN$	$\sim \beta_p MN$	$M \min(N, B)$	$M \min(N, B)$	$\frac{2M}{k} \min(\frac{N}{2k}, B)$
Nonlinearity	Electrical	Electrical	Electrical	Electrical	Electrical	Built-in
Output range	Non-negative only	Non-negative only	Non-negative only	Non-negative only	Full range	Full range
Control complexity	High	Medium-High	High	High	High	Medium

TABLE III: Comprehensive performance comparison between SqueezeLight and MRR-ONN. †To keep the same area cost, SqueezeLight uses 16 32×16 MORR arrays, and MRR-ONN uses 16 64×16 MRR weight banks in the accelerator. We use *DNN-Chip Predictor* [31] to search for optimal hierarchical tiling strategy for SqueezeLight and MRR-ONN, respectively, and use their optimal tiling strategies for energy simulation.

Design	Area ↓ (mm^2)	Power ↓ (W)	Latency ↓ (ps)	Operate Freq ↑ (GHz)	Comp. Density ↑ (TOPS/mm^2)	Energy Eff. ↑ (TOPS/W)	† Sys. Energy ↓ (μJ)
SqueezeLight	5.12	6.972 (-48%)	141.3 (-21.4%)	7.0 (+25%)	11.3 (4.9×)	8.32 (9.8×)	0.2440 (-63.5%)
MRR-ONN	5.02	13.402	179.7	5.6	2.3	0.85	0.6676

The energy efficiency of the MRR weight bank is $\frac{64\times16\times2\ OPs}{179.7\ ps\times13.402\ W} = 0.850\ TOPS/W$.

System Energy Cost. We use a DNN-Chip Predictor [31] to simulate a 256×256 fully connected layer with a four-level memory hierarchy, including DRAM, SRAM-based global buffer (GB), network-on-chip (NoC) which describes the spatial data tiling and the parallelism of the system, and register files (RF). For SqueezeLight, we use 16 32×16 MORR array in the accelerator. For MRR-ONN, we use 16 64×16 MRR weight banks in the accelerator. We searched for optimal tiling strategies for them and applied them to those two accelerators. The basic memory energy model is based on Eyeriss [37]. MRR-ONN consumes 0.5135 μJ on data movement. It consumes 0.1541 μJ in computation. The total energy consumption of MRR-ONN is 0.6676 μJ. In contrast, our sparse block-squeezing technique helps save 94% of the weight loading cost, such that SqueezeLight only consumes 0.2237 μJ on data movement. Plus the 0.0158 μJ in computation, the total energy consumption SqueezeLight is 0.2440 μJ, achieving 63.5% overall energy reduction. We summarize the above analysis in Table III.

V. Extension to MORR-based Separable CNN with Augmented Trainability

To enable a real scalable ONN design, the three most important metrics are *representability*, *hardware efficiency*, and *software trainability*. Based on the nonlinear MORR neuron, we have demonstrated an ONN architecture with high *hardware efficiency* and *representability* in Fig. 2. However, the unsatisfying *trainability* of the MORR-based ONN fundamentally restricts the scalability of SqueezeLight. Specifically, for convolution (CONV) layers, partial convolution results for each MORR need to be stored and activated by the built-in nonlinearity. Such a mechanism turns out to consume considerable GPU memory and training time. This software trainability issue motivates us to design a more suitable architecture based on MORR arrays that can fully unleash the scalability advantages of SqueezeLight with augmented trainability.

A. MORR-based Separable CNN with Layer-Squeezing

An important trade-off in MORR-based ONN design is between representability and trainability. Hence, we propose an MORR-based separable CNN architecture that coincides with an advanced neural network design concept, i.e., *depth-wise separable convolution* (DSCONV).

DSConv contains a depth-wise convolution (DWConv) with per channel convolution and a point-wise convolution with 1×1 kernels (PWConv), which can be taken as a low-rank decomposition of an original CONV. Such advanced convolution is widely used in efficient NN architectures, e.g., MobileNet-family, to trim unnecessary computations without degrading the representability. The most exciting observation is the perfect match between DSConv and our MORR array, shown in Fig. 7. In other words, we *squeeze DWConv and PWConv layers into one MORR array*. A feature patch with size of $C_{in} \times K \times K$ will convolve with the DWConv kernel $\boldsymbol{W}_D \in \mathbb{R}^{C_{in}\times1\times K\times K}$, corresponding to C_{in} length-K^2 dot-products. Hence, we can assign a row of C_{in} MORRs to implement it. Note that each k-operand MORR corresponds to one $K \times K$ CONV filter. Then, PWConv will perform pointwise linear projection on all channels with a kernel $\boldsymbol{W}_P \in \mathbb{R}^{C_{out}\times C_{in}\times1\times1}$ and generate the final feature map. The pointwise linear projection can be directly mapped to the MRR-based balancing factors. To achieve balanced output, we need to split the MORR array into a positive and a negative array, each implementing half of DSConv. The negative array equivalently achieves the negative half of $\boldsymbol{W}_P$. The reason why we do not adopt negative/positive rails on the same array is that we want to maximize the parameter space of $\boldsymbol{W}_P$

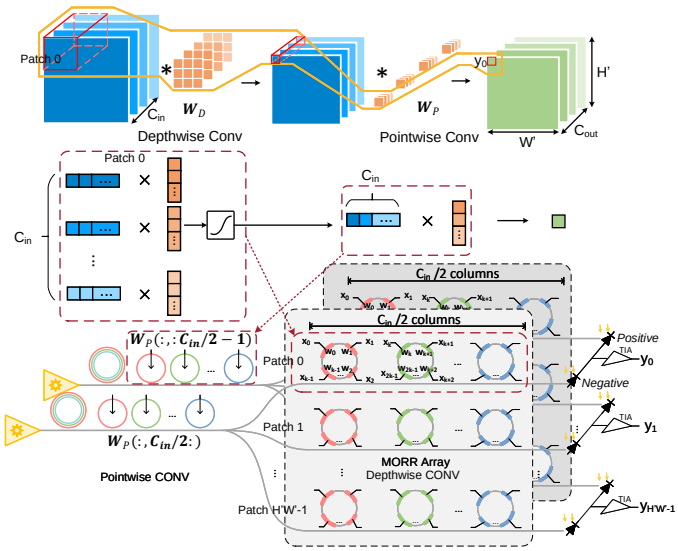


Fig. 7: Architecture of separable `SqueezeLight`. Squeeze depthwise and pointwise convolutional layers into one MORR array.

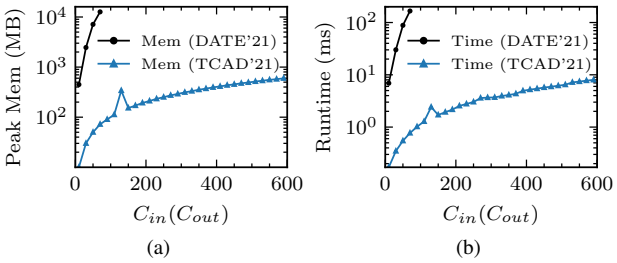


Fig. 8: Peak GPU memory consumption (a) and average GPU runtime (b) evaluation on an MORR-based CONV3x3 layer (DATE'21) and a `DSConv`3x3 layer (TCAD'21) with different input/output channels.

without weight sharing. Theoretically, there will be $H'W'$ rows to map all feature patches. For different output channels, we can either duplicate the array for C_{out} times or reuse the array and sequentially reprogram the MRR-based $\boldsymbol{W}_P$.

Compared with the original MORR CONV engine, this augmented `DSConv` engine has the following advantages.

Excellent Trainability. When mapping one CONV layer to the original MORR array using *im2col*, the largest intermediate partial product feature map contains $H'W'BPQk \approx H'W'BC_{out}C_{in}/k$ elements. In contrast, the largest feature map in the `DSConv` module only contains $H'W'BC_{in}$ elements. The training memory footprint is approximately improved by C_{out}/k times, which significantly boosts the software trainability of our `SqueezeLight`. Figure 8 shows *2-order-of-magnitude higher memory efficiency and runtime reduction* of the augmented `SqueezeLight` compared with the original MORR-based CONV engine [13]. Hence, by filling the trainability gap, all three aforementioned key metrics for scalable ONNs are met.

Compressed Model Size. This benefit naturally comes from the low-rank parameter space of `DSConv`. The weight size is reduced from $C_{out}C_{in}K^2$ to $C_{in}K^2 + C_{out}C_{in}$, with a compression ratio of $\sim K^2$.

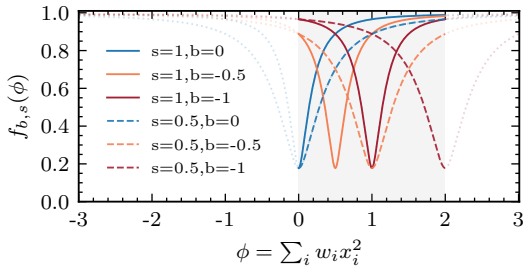


Fig. 9: Trainable nonlinearity curve of parametric MORR neurons with different bias b and scale s. Curves highlighted in the shadow region are the activation functions applied to the dot-product ϕ.

Patch-Level Parallelism. The original MORR array essentially performs sequential matrix-vector multiplication, which processes one feature patch at one time. In contrast, the augmented MORR array maps multiple image patches to different rows in parallel, which share the same group of MRRs. Another advantage of this patch-level parallelism is the massive reuse of MRRs for $\boldsymbol{W}_P$. With the extensive MRR reuse, the advantages of ultra-compact MORR neurons will not be diluted by the usage of MRRs.

B. Parametric MORR Neuron via Trainable Nonlinearity

Thanks to the augmented software trainability of the MORR-based separable CONV engine, we are able to efficiently explore the representability of `SqueezeLight` deeper. Moving beyond the fixed nonlinear transmission curve of an MORR, we further explore more expressivity in our MORR-based neuron via trainable nonlinearity. Inspired by previous work on learning activations for NNs, we try to adapt the shape and the sharpness of the nonlinear curve by tuning the phase bias b and the input scaling factor s,

$$f_{b,s}(\phi) = f(s\phi + b) = f(s\sum_{i=0}^{k-1} w_i x_i^2 + b). \tag{13}$$

Figure 9 shows how b and s change the nonlinearity applied to the dot-product results. A dedicated biasing current can be applied to the actuators on MORRs to tune the curve's center wavelength. By scaling the heating power range of x with the factor s, the sharpness of the nonlinearity can also be efficiently tuned. Such tunable nonlinearity introduces extra non-convexity, leading to stronger representability than conventional activation functions, e.g., ReLU. In later section, we will show the performance benefits from our trainable MORR neuron.

C. Nonlinearity-aware Initialization

A proper initialization is critical to the convergence of nonlinear nonconvex optimization problems, especially for DNN training. Though various normalization methods, e.g., BatchNorm, can relax the sensitivity of DNN learning to parameter initialization, the built-in nonlinearity of MORRs

still requires appropriate weight distribution to avoid dot-product values falling into saturation ranges. The second reason for a specialized initialization method is the potential activation explosion due to normalized MORR output. Each MORR has a normalized output range of $[0, 1]$, such that the final activation magnitude is nearly proportional to the number of MORRs cascaded on one row.

Therefore, we show a nonlinearity-aware initialization algorithm to maintain nearly constant variance after layer cascading. We first assume the input x is normalized with zero center, i.e., $\mathbb{E}[x] = 0, \mathbb{D}[x] = \sigma_x^2$, and the statistics of non-negative weights are denoted as $\mathbb{E}[w]$ and $\mathbb{D}[w]$. Based on $\phi = \sum_{i=0}^{k-1} w_i x_i^2$, thus we can derive the variance of the accumulated round-trip phase shift of a k-segment MORR since x and w are independent random variables,

$$\begin{aligned}\mathbb{D}[\phi] =& k(\mathbb{E}^2[w]\mathbb{D}[x^2] + \mathbb{E}^2[x^2]\mathbb{D}[w] + \mathbb{D}[w]\mathbb{D}[x^2]) \\ =& k\sigma_x^4(2\mathbb{E}^2[w] + 3\mathbb{D}[w]).\end{aligned} \tag{14}$$

In our algorithm, the weights will be sampled from a non-negative uniform distribution, i.e., $w \sim \mathcal{U}(0, L)$. Thus Eq. (14) can be rewritten as,

$$\mathbb{D}[\phi] = k\sigma_x^4\left(2\left(\frac{L}{2}\right)^2 + \frac{3L^2}{12}\right) = \frac{3k\sigma_x^4 L^2}{4}. \tag{15}$$

To solve L analytically, we need to know $\mathbb{D}[\phi]$. Considering the inter-MORR crosstalk due to spectrum leakage, a typical spectral distance between two adjacent wavelengths are at least 4 FWHM, where the full width half maximum (FWHM) represents the peak width when the energy is reduced to 50%. We heuristically and conservatively set a constraint to the maximum tuning range ($\pm 2\sigma_\phi$) of the round-trip phase shift, i.e., $4\sqrt{\mathbb{D}[\phi]} \approx 3$ FWHM. Now we given the uniform distribution for the weights w,

$$\texttt{morr\_uniform}(w) \sim \mathcal{U}\left(0, \sigma_x^2 \text{ FWHM}\sqrt{\frac{3}{4k}}\right). \tag{16}$$

So far, we properly initialize weights w considering the MORR transmission curve $f(\cdot)$. The next step is to initialize the learnable balancing factors $\tilde{\boldsymbol{D}}$ to keep the variance of the final activation the same as that of inputs x, i.e., $\mathbb{D}[y] = \mathbb{D}[x]$. The target distribution of balancing factors is zero-centered normal distribution, i.e., $\tilde{d} \sim \mathcal{N}(0, \sigma_{\tilde{d}}^2)$. Given the differential detection result $y = \sum_{q=0}^{Q-1} f(\phi)\tilde{d}_q$, we can rewrite it as $y = \sum_{q=0}^{Q/2} \Delta f(\phi)\tilde{d}_q$, where $\Delta f(\phi)$ is the equivalent differential MORR transmission between positive and negative rails. We have $\mathbb{E}[\Delta f(\phi)] = 0$. Then we can derive the variance of the final activation,

$$\begin{aligned}\mathbb{D}[y] =& \frac{Q}{2}(\mathbb{E}^2[\Delta f(\phi)]\mathbb{D}[\tilde{d}] + \mathbb{E}^2[\tilde{d}]\mathbb{D}[\Delta f(\phi)] + \mathbb{D}[\Delta f(\phi)]\mathbb{D}[\tilde{d}]) \\ =& \frac{Q}{2}\mathbb{D}[\Delta f(\phi)]\sigma_{\tilde{d}}^2 = Q\mathbb{D}[f(\phi)]\sigma_{\tilde{d}}^2 = \sigma_x^2.\end{aligned} \tag{17}$$

The variance of the balancing factor is given by

$$\mathbb{D}[\tilde{d}] = \frac{\sigma_x^2}{Q\ \mathbb{D}[f(\phi)]} \approx \frac{\sigma_x^2}{Q \cdot g_f^2 \mathbb{D}[\phi]} = \frac{16\sigma_x^2}{9Q \cdot g_f^2 \cdot \text{FWHM}^2}, \tag{18}$$

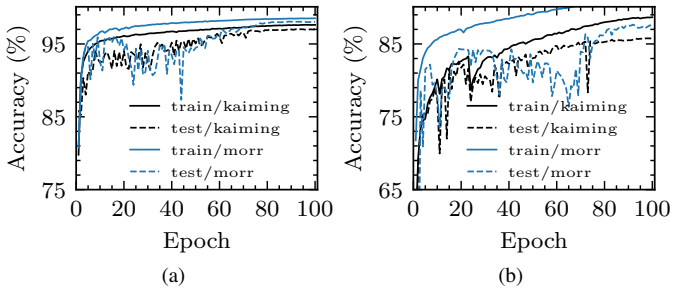


Fig. 10: Compare the training and test accuracy curves on MNIST (a) and FashionMNIST (b). We compare our proposed `morr_uniform` with the kaiming initializer.

where g_f is a linear approximation to the gradient of the nonlinear transmission, i.e, $g_f = \frac{f(\phi_c + 2FWHM) - f(\phi_c)}{2FWHM}$, where ϕ_c is the on-resonance phase shift.

Now, we show an ablation study to validate the effectiveness of the proposed nonlinearity-aware initialization method. From the training curves shown in Figure 10, we observe considerably faster convergence and higher test accuracy by using our proposed MORR-aware initialization method. In the following experiments, we will use the proposed initialization by default.

VI. Experimental Results

We conduct optical simulation to validate the functionality and evaluate `SqueezeLight` on MNIST [38], FashionMNIST (FMNIST) [39], SVHN [40], CIFAR-10 [41], and CIFAR-100 dataset. All models are implemented with a PyTorch-centric ONN library TorchONN [42]. All ONNs are trained for 100 epochs using the Adam optimizer. Quantization-aware training [43] is applied to perform 8-bit weight/input/activation quantization.

A. Functionality Validation via Optical Simulation

One MORR Neuron. The MORR-based neuron is simulated using the commercial Lumerical INTERCONNECT tool for functional validation. Figure 11 plots the theoretical and simulated outputs of a 4-operand MORR under 1- to 4-bit precision. The design specification of the MORR is as follows. Radius $R = 20\mu m$, transmission coefficient $r = 0.8985$, attenuation factor $a = 0.8578$, effective index $n_{eff} = 2.35$. The central resonance wavelength is 1554.252 nm. We assume the 4-op MORR is programmed with 1- to 4-bit weights w, and we apply 1- to 4-bit voltage signals x to its controllers. We use Lumerical INTERCONNECT to simulate the intensity transmission of this MORR under given input/weights. The detector sensitivity is set to 1 A/W. Only the insertion loss of MORR is considered, while the loss in the waveguide is ignored. The derived neuron model has a high fidelity with $<1\%$ relative error compared with simulation results.

MORR Array. We further simulate a 2×4 MORR array with 4 MRRs to implement balancing factors, together with 4 4-op MORRs on the positive rail and another 4 MORRs on the

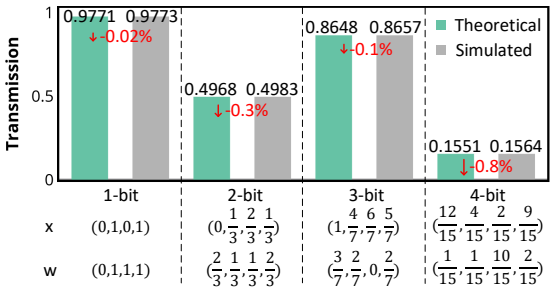


Fig. 11: Compare theoretical and simulated results of an 4-op MORR.

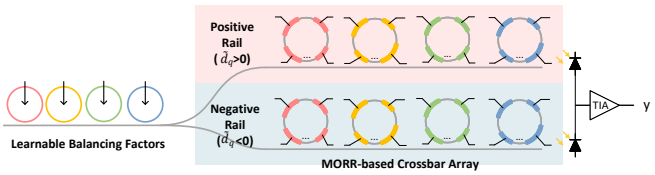


Fig. 12: 2×4 MORR array used in simulation.

negative rail, as shown in Fig. 12. In the end, we add the differential photo-detection. All electrical voltage controls are of 4-bit precision. We use 1550, 1554, 1558, and 1562 nm as WDM sources. For the above 4 resonance wavelengths, we design the rings with a radius of 10.08 μm, 10.10 μm, 10.13 μm, and 10.16 μm, respectively. The transmission coefficient is $r = 0.98$, and the attenuation coefficient is $a = 0.97$. This MORR has much higher Q values and larger FSR than the 20 μm MORR used in the single MORR neuron simulation, which can enable higher WDM capacity with less spectrum crosstalk issue. In 4 test cases, the simulation results slightly deviate from the theoretical values due to wavelength misalignment, spectrum crosstalk, MORR insertion loss, etc., which validate the functionality of `SqueezeLight`.

B. Compare `SqueezeLight` with Prior MRR-ONNs

In Table V, we compare the test accuracy among three ONNs: 1) MRR-ONN-1 with all-pass MRRs [10], 2) MRR-ONN-2 with add-drop MRRs [9], and 3) our proposed `SqueezeLight` without pruning (Ours). In all dataset and ONN settings, `SqueezeLight` achieves comparable test accuracy with 20-30× fewer ring resonators, 8× lower wavelength usage, and ∼80% fewer parameters.

C. Quantization

We also evaluate our architecture with low-bit quantization. Even binarized `SqueezeLight` can achieve >95% accuracy on MNIST with the *large* model, and >98% accuracy can be maintained with 2∼8 bit precision. Note that prior work has demonstrated MRR weight banks with higher than 7-bit weight precision [35]. Our `SqueezeLight` can work will with low-bit weight precision, which further justifies the practicality of our design.

TABLE IV: Length-16 4-bit nonlinear vector-product simulated on a 2×4 4-op MORR array with 4 MRRs.

Test case	Simulated $\hat{y}$	Theoretical y	Error $\lvert\hat{y} - y\rvert$
1	0.3709	0.3708	0.0001
2	-0.1070	-0.0811	0.0259
3	-0.5916	-0.6170	0.0254
4	0.8505	0.8717	0.0212

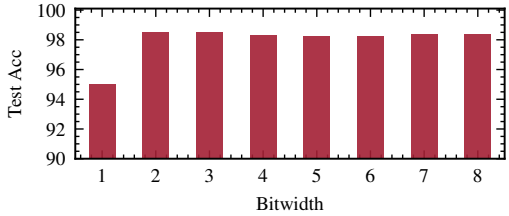


Fig. 13: 1- to 8-bit quantization of `SqueezeLight` on MNIST.

D. Fine-Grained Structured Pruning

In Table VI, the pruned `SqueezeLight` only requires 4-operand MORRs to implement sparse sub-matrices with k'=4, which reduces the manufacturing and control complexity with no accuracy loss. Moreover, the saved 30% parameters lead to less weight storage cost. This enables us to achieve better scalability by squeezing larger blocks into one MORR with negligible accuracy loss.

E. Variation Robustness Evaluation

In Fig. 14, we evaluate the variation robustness on 1) MRR-ONN-1, 2) MRR-ONN-2, 3) unpruned `SqueezeLight` (Ours), 4) pruned `SqueezeLight` (Ours-P), and 5) ours with pruning and robustness-aware training (Ours-PR). In the presence of the additional intra-MORR crosstalk, our ONN shows lower accuracy than other MRR-ONNs if no pruning or noise-aware training is performed. When we apply fine-grained structured pruning, the crosstalk sources are cut down from $k = 8$ to $k' = 4$, achieving improved noise tolerance. With sensitivity-aware training based on Eq. (7), `SqueezeLight` can stably maintain above 97% accuracy, which is reasonably close to the ideal accuracy, while other ONNs suffer from a sharply-degrading trend as the noise intensity increases. Therefore, our proposed lightweight robustness-aware training guarantees `SqueezeLight` to have reliable inference performance even under practical non-ideal variations.

F. Extended MORR-based Separable CNN

In Table VII, we thoroughly evaluate the scalability and effectiveness of the extended separable CNN architecture on various learning tasks and models. On large models, the training

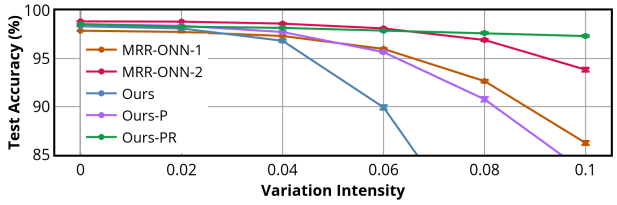


Fig. 14: Robustness evaluation of the *large* model on MNIST. The error bar shows ±1σ over 20 runs, e.g., 0.04 means γ=0.04 and std. $\Delta\phi$=0.04. *Ours-PR* means our pruned model with sensitivity-aware training (α=0.02).

TABLE V: Accuracy and hardware cost comparison. *small* model is C32K5S2-BN-C32K5S2-BN-F10, where *C32K5S2* is 5×5 convolution with 32 kernels and stride 2, *BN* is BatchNorm, and *F10* is a linear layer. *large* model is C64K5S2-BN-C64K5S2-BN-F10. We use $k = 8$ in convolutional layers and $k = 4$ in the final classifier. *#Device*, *#λ*, and *#Param* are the number of used resonators, wavelengths, and parameters, respectively. Normalized ratios are shown in the parenthesis. All models are trained with 8-bit weight/input/activation quantization.

Dataset	Model	MRR-ONN-1 [10]				MRR-ONN-2 [9]				Ours			
		Test Acc.	#Device	#λ	#Param	Test Acc.	#Device	#λ	#Param	Test Acc.	#Device	#λ	#Param
MNIST	small	97.81	39.90 K (23.86)	1152(8)	38 K	98.55	39.90 K (23.86)	1152(8)	38 K	98.01	1.67 K (1.00)	144(1)	8 K
MNIST	large	97.89	130.97 K (31.64)	2304(8)	127 K	98.84	130.97 K (31.64)	2304(8)	127 K	98.36	4.14 K (1.00)	288(1)	22 K
FMNIST	small	86.97	39.90 K (23.86)	1152(8)	38 K	89.52	39.90 K (23.86)	1152(8)	38 K	86.65	1.67 K (1.00)	144(1)	8 K
FMNIST	large	87.75	130.97 K (31.64)	2304(8)	127 K	90.30	130.97 K (31.64)	2304(8)	127 K	87.21	4.14 K (1.00)	288(1)	22 K
CIFAR-10	large	48.79	143.37 K (28.50)	3136(8)	139 K	61.69	143.37 K (28.50)	3136(8)	139 K	58.29	5.03 K (1.00)	392(1)	26 K

TABLE VI: Fine-grained structured pruning evaluation. *#8op* represents the number of 8-operand MORRs. *Ours-P* represents all convolutional layers are pruned from k=8 to k'=4.

Dataset	Model	Ours				Ours-P			
		Acc.	#8op	#4op	#Param	Acc.	#8op	#4op	#Param
MNIST	small	98.01	416	864	8 K	98.02	0	1280	6 K
MNIST	large	98.36	1632	1728	22 K	98.58	0	3360	16 K
FMNIST	small	86.65	416	864	8 K	86.50	0	1280	6 K
FMNIST	large	87.21	1632	1728	22 K	87.36	0	3360	16 K
CIFAR-10	large	58.29	1680	2352	26 K	60.52	0	4032	19 K

TABLE VII: Compare accuracy of separable `SqueezeLight` with fixed and learnable MORR nonlinearity on various tasks and models. We further prune convolutional kernels from k=9 to k'=4 to make it implementable with 4-operand MORRs. The suffix *-L* and *-P* represent using trainable MORR nonlinearity and structured pruning, respectively. The settings for CNN-2 are C64-C64-Pool5-F10. The settings for CNN-3 are C64-C64-C64-Pool5-F10. All convolutional layers (except for the first layer) in the model are implemented by the proposed MORR-based separable convolution.

Model	CNN-2	CNN-3	VGG-8		
Dataset	MNIST	FMNIST	SVHN	CIFAR-10	CIFAR-100
Ours	98.07	87.66	93.09	83.61	56.92
Ours-L	**98.67**	89.07	93.75	84.78	58.61
Ours-LP	98.37	**90.65**	**93.82**	**86.31**	**60.83**

of the original MORR CNN [13] fails due to prohibitive GPU memory and runtime cost. Thanks to the superior software trainability of our MORR-based separable convolution, we can scale the extended `SqueezeLight` to *million-parameter* ONN models, e.g., VGG-8, on various vision recognition datasets. Meanwhile, our separable MORR-based architecture saves ∼9× parameters compared with the original Conv-based ONN model, leading to significant storage cost reduction.

We further compare `SuqeezeLight` with and without trainable MORR nonlinearity. Figure. 15 visualizes the learned channel-wise MORR nonlinearity curves in two `DSConv` layers of VGG-8. We observe that the `SqueezeLight` explores various monotonic or non-monotonic activation functions with augmented representability than a fixed zero-bias MORR nonlinearity curve. Our trainable MORR neurons boost the representability to effectively compensate for the performance loss from parameter compression, leading to an average of ∼1.1% test accuracy improvement on 5 learning tasks.

Note that the 3×3 depthwise convolution maps 9 weights to 1 MORR, which exceeds the typical capacity of 4 operands per MORR. We apply structured pruning to leave 4 non-zero weights in each depthwise convolutional kernel and demonstrate an average 2.13% accuracy improvement in Table VII.

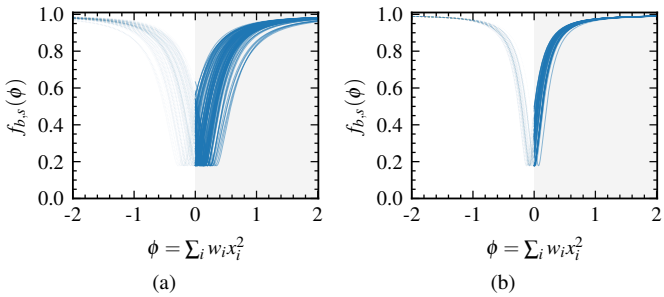


Fig. 15: Learned MORR nonlinearity for the 1st (a) and 3rd (b) `DSConv` layers in VGG-8 on CIFAR-10. Each curve represents the nonlinearity curve of one input channel.

VII. Conclusion

In this work, we propose a novel ONN architecture `SqueezeLight` to break the compactness record of previous designs with higher scalability and efficiency. An MORR-based optical neuron with built-in nonlinearity is proposed to squeeze vector dot-product into a single device. A block-squeezing technique with fine-grained structured pruning is proposed to further squeeze a matrix into an MORR to enable a quadratically more compact ONN design. We introduce sensitivity-aware training to enable close-to-ideal neurocomputing with high noise robustness. We give a theoretical analysis and thorough comparison to show the scalability and efficiency advantage of `SqueezeLight`. We extend `SqueezeLight` to an MORR-based separable CNN architecture with layer-wise squeezing and learnable nonlinearity, showing order-of-magnitude higher software training scalability and expressiveness improvement. Experiments show that `SqueezeLight` breaks the area lower bound of previous MRR-based ONNs with 20-30× better scalability and competitive expressiveness.

References

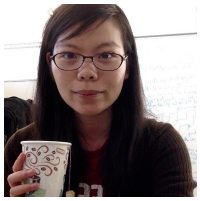
[1] Y. Shen, N.C. Harris, S. Skirlo *et al.*, "Deep learning with coherent nanophotonic circuits," *Nature Photonics*, 2017.
[2] M. Miscuglio and V.J. Sorger, "Photonic tensor cores for machine learning," *Applied Physics Review*, 2020.
[3] Q. Cheng, J. Kwon, M. Glick, M. Bahadori, L.P. Carloni *et al.*, "Silicon Photonics Codesign for Deep Learning," *Proceedings of the IEEE*, 2020.
[4] B.J. Shastri, A.N. Tait, T.F. de Lima, W.H.P. Pernice, H. Bhaskaran *et al.*, "Photonics for Artificial Intelligence and Neuromorphic Computing," *Nature Photonics*, 2021.
[5] Z. Zhao, D. Liu, M. Li *et al.*, "Hardware-software co-design of slimmed optical neural networks," in *Proc. ASPDAC*, 2019.
[6] J. Gu, Z. Zhao, C. Feng *et al.*, "Towards area-efficient optical neural networks: an FFT-based architecture," in *Proc. ASPDAC*, 2020.

[7] J. Gu, Z. Zhao, C. Feng *et al.*, "Towards Hardware-Efficient Optical Neural Networks: Beyond FFT Architecture via Joint Learnability," *IEEE TCAD*, 2020.
[8] C. Feng, J. Gu, H. Zhu, Z. Ying, Z. Zhao *et al.*, "Silicon photonic subspace neural chip for hardware-efficient deep learning," *arXiv preprint arXiv:2111.06705*, 2021.
[9] A.N. Tait, T.F. de Lima, E. Zhou *et al.*, "Neuromorphic photonic networks using silicon photonic weight banks," *Sci. Rep.*, 2017.
[10] W. Liu, W. Liu, Y. Ye, Q. Lou, Y. Xie *et al.*, "Holylight: A nanophotonic accelerator for deep learning in data centers," in *Proc. DATE*, 2019.
[11] F. Zokaee, Q. Lou, N. Youngblood *et al.*, "LightBulb: A Photonic-Nonvolatile-Memory-based Accelerator for Binarized Convolutional Neural Networks," in *Proc. DATE*, 2020.
[12] F. Sunny, A. Mirza, M. Nikdast, and S. Pasricha, "CrossLight: A Cross-Layer Optimized Silicon Photonic Neural Network Accelerator," in *Proc. DAC*, 2021, pp. 1069–1074.
[13] J. Gu, C. Feng, Z. Zhao, Z. Ying, M. Liu *et al.*, "SqueezeLight: Towards Scalable Optical Neural Networks with Multi-Operand Ring Resonators," in *Proc. DATE*, Feb. 2021.
[14] J. Mairal, P. Koniusz, Z. Harchaou, and C. Schmid, "Convolutional Kernel Networks," in *Proc. NeurIPS*, 2014.
[15] W. Liu, Z. Liu, Z. Yu *et al.*, "Decoupled Networks," in *Proc. CVPR*, 2018.
[16] Y. Chen, X. Dai, M. Liu *et al.*, "Dynamic convolution: Attention over convolution kernels," in *Proc. CVPR*, 2020.
[17] J. Gu, Z. Zhao, C. Feng, H. Zhu, R.T. Chen *et al.*, "ROQ: A noise-aware quantization scheme towards robust optical neural networks with low-bit controls," in *Proc. DATE*, 2020.
[18] J. Gu, Z. Zhao, C. Feng, W. Li, R.T. Chen *et al.*, "FLOPS: Efficient On-Chip Learning for Optical Neural Networks Through Stochastic Zeroth-Order Optimization," in *Proc. DAC*, 2020.
[19] Z. Ying, C. Feng, Z. Zhao *et al.*, "Integrated multi-operand electro-optic logic gates for optical computing," *Appl. Phys. Lett.*, 2019.
[20] E. Timurdogan, Z. Su, C.V. Poulton *et al.*, "AIM Process Design Kit (AIMPDKv2.0): Silicon Photonics Passive and Active Component Libraries on a 300mm Wafer," in *Optical Fiber Communication Conference*, 2018.
[21] Z. Sun, W. Wu, and H.H. Li, "Cross-Layer Racetrack Memory Design for Ultra High Density and Low Power Consumption," in *Proc. DAC*, 2013.
[22] C. Ding, S. Liao, Y. Wang, Z. Li, N. Liu *et al.*, "CirCNN: Accelerating and Compressing Deep Neural Networks Using Block-Circulant Weight Matrices," in *Proc. MICRO*, 2017, pp. 395–408.
[23] Z. Zhao, J. Gu, Z. Ying *et al.*, "Design technology for scalable and robust photonic integrated circuits," in *Proc. ICCAD*, 2019.
[24] Y. Zhu, G.L. Zhang, B. Li *et al.*, "Countering Variations and Thermal Effects for Accurate Optical Neural Networks," in *Proc. ICCAD*, 2020.
[25] A. Mirza, F. Sunny, P. Walsh, K. Hassan, S. Pasricha *et al.*, "Silicon Photonic Microring Resonators: A Comprehensive Design-Space Exploration and Optimization under Fabrication-Process Variations," *IEEE TCAD*, pp. 1–1, 2021.
[26] M. Milanizadeh, D. Aguiar, A. Melloni, and F. Morichetti, "Canceling thermal cross-talk effects in photonic integrated circuits," *J. Light. Technol.*, 2019.
[27] D.T.H. Tan, A. Grieco, and Y. Fainman, "Towards 100 channel dense wavelength division multiplexing with 100ghz spacing on silicon," *Opt. Express*, 2014.
[28] J. Yu and X. Zhou, "Ultra-high-capacity dwdm transmission system for 100g and beyond," *IEEE Communications Magazine*, 2010.
[29] N.C. Harris *et al.*, "Efficient, compact and low loss thermo-optic phase shifter in silicon," *Opt. Express*, 2014.
[30] J. Sun, R. Kumar, M. Sakib, J.B. Driscoll, H. Jayatilleka *et al.*, "A 128 gb/s pam4 silicon microring modulator with integrated thermo-optic resonance tuning," *Journal of Lightwave Technology*, vol. 37, no. 1, pp. 110–115, 2019.
[31] Y. Zhao, C. Li, Y. Wang, P. Xu, Y. Zhang *et al.*, "Dnn-chip predictor: An analytical performance predictor for dnn accelerators with various dataflows and hardware architectures," in *Proc. ICASSP*, 05 2020, pp. 1593–1597.
[32] "Adc (analog-to-digital converters) – alphacore." https://www.alphacoreinc.com/adc-analog-to-digital-converters/.
[33] A.N. Tait, "Quantifying power use in silicon photonic neural networks," *arxiv preprint, arXiv:2108.04819*, 2021.
[34] B.S.G. Pillai *et al.*, "End-to-end energy modeling and analysis of long-haul coherent transmission systems," *J. Light. Technol.*, 2014.
[35] C. Huang *et al.*, "A silicon photonic–electronic neural network for fibre nonlinearity compensation," *Nat. Electron.*, 2021.
[36] M.A. Nahmias, T.F. de Lima, A.N. Tait, H. Peng, B.J. Shastri *et al.*, "Photonic multiply-accumulate operations for neural networks," *JSTQE*, 2020.
[37] Y. Chen, T. Krishna, J. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks," in *Proc. ISSCC*, 2016, pp. 262–263.
[38] Y. LeCun, "The MNIST database of handwritten digits," http://yann.lecun.com/exdb/mnist/, 1998.
[39] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," *CoRR*, vol. abs/1708.07747, 2017. [Online]. Available: http://arxiv.org/abs/1708.07747
[40] Y. Netzer, T. Wang, A. Coates, A. Bissacco *et al.*, "Reading Digits in Natural Images with Unsupervised Feature Learning," in *Proc. NIPS*, 2011.
[41] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.
[42] J. Gu, H. Zhu, C. Feng, Z. Jiang, R.T. Chen *et al.*, "L2ight: Enabling On-Chip Learning for Optical Neural Networks via Efficient in-situ Subspace Optimization," in *Proc. NeurIPS*, 2021.
[43] S. Zhou, Z. Ni, X. Zhou, H. Wen, Y. Wu *et al.*, "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients," *arXiv preprint arXiv:1606.06160*, 2016.

Jiaqi Gu (S'19) received the B.E. degree in Microelectronic Science and Engineering from Fudan University, Shanghai, China in 2018. He is currently a post-graduate student studying for his Ph.D. degree in the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA. His current research interests include machine learning, efficient algorithm and architecture design for high-performance AI, next-generation AI computing with emerging technology, and GPU acceleration for VLSI design automation. He has received the Best Paper Award at IEEE TCAD 2021, the Best Paper Award at ASP-DAC 2020, the Best Paper Finalist at DAC 2020, the Best Poster Award at NSF Workshop on Machine Learning Hardware (2020), the ACM/SIGDA Student Research Competition First Place (2020), and the ACM Student Research Competition Grand Finals First Place (2021).

Chenghao Feng received the B.S. degree in physics from Nanjing University, Nanjing, China, in 2018. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX, USA. His research interests include silicon photonics devices and system design for optical computing and interconnect in integrated photonics.

Hanqing Zhu (S'20) received the B.E. degree in Microelectronic Science and Engineering from Shanghai Jiao Tong University, Shanghai, China in 2020. He is currently pursuing his Ph.D. degree in the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA. His current research interests include high-performance AI computing with emerging technologies, machine learning, and its application to VLSI physical design automation.



Zheng Zhao received the B.S. degree in automation from Tongji University, Shanghai, China, in 2012, and the M.S. degree in electrical and computer engineering from Shanghai Jiao Tong University, Shanghai, China, in 2015. She is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX, USA, under the supervision of Prof. D. Z. Pan. Her research interests include optical computing/interconnect, neuromorphic computing and logic synthesis.

Zhoufeng Ying (M'20) received the B.E. and M.E. degrees in optical engineering from Nanjing University, Nanjing, China, in 2014 and 2016, respectively, and Ph.D degree in electrical and computer engineering from University of Texas at Austin, Austin, TX, USA, in 2020. After his Ph.D degree, he joined Alpine Optoelectronics, as a senior silicon photonics designer.

Mingjie Liu Mingjie Liu received his B.S degree from Peking University and M.S. degree from the University of Michigan, Ann Arbor in 2016 and 2018, respectively. He is currently pursuing his Ph.D. degree in Electrical and Computer Engineering at The University of Texas at Austin. His current research interests include applied machine learning for design automation, and physical design automation for analog and mixed-signal integrated circuits.

Ray T. Chen (M'91–SM'98–F'04) received the B.S. degree in physics from the National Tsing Hua University, Hsinchu, Taiwan, in 1980, and the M.S. degree in physics and Ph.D. degree in electrical engineering from the University of California, in 1983 and 1988, respectively. He is the Keys and Joan Curry/Cullen Trust Endowed Chair with the University of Texas at Austin (UT Austin), Austin, TX, USA. He is the Director of the Nanophotonics and Optical Interconnects Research Lab, Microelectronics Research Center. He is also the Director of the AFOSR MURI-Center for Silicon Nanomembrane involving faculty from Stanford, UIUC, Rutgers, and UT Austin. In 1992, he joined the UT Austin to start the optical interconnect research program. From 1988 to 1992, he worked as a Research Scientist, Manager, and Director of the Department of Electro-Optic Engineering, Physical Optics Corporation, Torrance, CA, USA.

From 2000 to 2001, he served as the CTO, founder, and Chairman of the Board of Radiant Research, Inc., where he raised 18 million dollars A-Round funding to commercialize polymer-based photonic devices involving more than 20 patents, which were acquired by Finisar in 2002, a publicly traded company in the Silicon Valley (NASDAQ:FNSR). He served as the CTO, Founder, and Chairman of the Board of Radiant Research, Inc. from 2000 to 2001, where he raised 18 million dollars A-Round funding to commercialize polymer-based photonic devices involving over twenty patents, which were acquired by Finisar in 2002, a publicly traded company in the Silicon Valley (NASDAQ:FNSR). He also serves as the founder and Chairman of the Board of Omega Optics Inc. since its initiation in 2001. Omega Optics has received over five million dollars in research funding. His research work has been awarded over 145 research grants and contracts from such sponsors as Army, Navy, Air Force, DARPA, MDA, NSA, NSF, DOE, EPA, NIST, NIH, NASA, the State of Texas, and private industry. The research topics are focused on four main subjects: (1) Nano-photonic passive and active devices for bio- and EM-wave sensing and interconnect applications, (2) Thin film guided-wave optical interconnection and packaging for 2D and 3D laser beam routing and steering, (3) True time delay (TTD) wide band phased array antenna (PAA), and (4). 3D printed micro-electronics and photonics. Experiences garnered through these programs are pivotal elements for his research and further commercialization.

His group at UT Austin has reported its research findings in more than 970 publications, including over 100 invited papers and 74 patents. He has chaired or been a program-committee member for more than 130 domestic and international conferences organized by IEEE, SPIE (The International Society of Optical Engineering), OSA, and PSC. He has served as an editor, co-editor or coauthor for over twenty books. Chen has also served as a consultant for various federal agencies and private companies and delivered numerous invited talks to professional societies. Chen is a Fellow of IEEE, OSA, and SPIE. He was the recipient of the 1987 UC Regent's Dissertation Fellowship and the 1999 UT Engineering Foundation Faculty Award, for his contributions in research, teaching and services. He received the honorary citizenship award in 2003 from the Austin city council for his contribution in community service. He was also the recipient of the 2008 IEEE Teaching Award, and the 2010 IEEE HKN Loudest Professor Award. 2013 NASA Certified Technical Achievement Award for contribution on moon surveillance conformable phased array antenna. During his undergraduate years at the National Tsing Hua University he led the 1979 university debate team to the Championship of the Taiwan College-Cup Debate Contest.

David Z. Pan (S'97–M'00–SM'06–F'14) received his B.S. degree from Peking University in 1992, and his M.S. and Ph.D. degrees from University of California, Los Angeles (UCLA), in 1998 and 2000. From 2000 to 2003, he was a Research Staff Member with IBM T. J. Watson Research Center. He is currently a Full Professor and holder of the Silicon Laboratories Endowed Chair in Electrical Engineering at The University of Texas at Austin. His research interests include electronic design automation, design for manufacturing, machine learning and hardware acceleration, design/CAD for analog/mixed signal designs and emerging technologies. He has published over 430 journal articles and refereed conference papers, and is the holder of 8 U.S. patents. He has graduated 42 PhD/postdocs who are holding key academic and industry positions.

He has served as a Senior Associate Editor for ACM Transactions on Design Automation of Electronic Systems (TODAES), an Associate Editor for IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems (TCAD), IEEE Transactions on Very Large Scale Integration Systems (TVLSI), IEEE Transactions on Circuits and Systems PART I (TCAS-I), IEEE Transactions on Circuits and Systems PART II (TCASII), IEEE Design & Test, Science China Information Sciences, Journal of Computer Science and Technology, IEEE CAS Society Newsletter, etc. He has served in the Executive and Program Committees of many major conferences. He is the ISPD 2008 General Chair, DAC 2014 Tutorial Chair, ASP-DAC 2017 Program Chair, ICCAD 2018 Program Chair, and ICCAD 2019 General Chair, and DAC 2022 Panel Chair.

He has received a number of prestigious awards for his research contributions, including the SRC Technical Excellence Award in 2013, DAC Top 10 Author in Fifth Decade, DAC Prolific Author Award, ASP-DAC Frequently Cited Author Award, ASP-DAC Prolific Author Award, 20 Best Paper Awards at premier venues (TCAD 2021, ISPD 2020, ASP-DAC 2020, DAC 2019, GLSVLSI 2018, VLSI Integration 2018, HOST 2017, SPIE 2016, ISPD 2014, ICCAD 2013, ASP-DAC 2012, ISPD 2011, IBM Research 2010 Pat Goldberg Memorial Best Paper Award, ASP-DAC 2010, DATE 2009, ICICDT 2009, SRC Techcon in 1998, 2007, 2012 and 2015) and 18 additional Best Paper Award finalists, Communications of the ACM Research Highlights (2014), ACM/SIGDA Outstanding New Faculty Award (2005), NSF CAREER Award (2007), SRC Inventor Recognition Award three times, IBM Faculty Award four times, UCLA Engineering Distinguished Young Alumnus Award (2009), UT Austin RAISE Faculty Excellence Award (2014), Cadence Academic Collaboration Award (2019), and many international CAD contest awards, among others. His students have also won many awards, including the First Place of ACM Student Research Competition Grand Finals in 2018, ACM/SIGDA Student Research Competition Gold Medal (twice), ACM Outstanding PhD Dissertation in EDA (twice), EDAA Outstanding Dissertation Award (thrice), and so on. He is a Fellow of ACM, IEEE and SPIE.