

Fully analog end-to-end online training with real-time adaptability in integrated photonics

Bhavin Shastri

shastri@ieee.org

Queen's University <https://orcid.org/0000-0001-5040-8248>

Zhimu Guo

Queen's University <https://orcid.org/0000-0001-9320-7913>

A Aadhi

IIT Delhi

Ammar Ibrahim

Queens University

Adam McCaughan

National Institute of Standards and Technology <https://orcid.org/0000-0002-8553-6474>

Alexander Tait

Queen's University <https://orcid.org/0000-0002-9774-4131>

Nathan Youngblood

University of Pittsburgh <https://orcid.org/0000-0003-2552-9376>

Sonia Buckley

National Institute of Standards and Technology <https://orcid.org/0000-0003-2809-9287>

Article

Keywords: Analog Training, Online Tracking, Adaptive Training, In-Situ Training, Online Learning

Posted Date: October 20th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11350311/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: There is **NO** Competing Interest.

Fully analog end-to-end online training with real-time adaptability in integrated photonics

Zhimu Guo^{1*†}, A. Aadhi^{2†}, Ammar A. Ibrahim¹, Adam N. McCaughan³,
Alexander N. Tait⁴, Nathan Youngblood⁵, Sonia M. Buckley³,
Bhavin J. Shastri^{1,4*}

¹Centre for Nanophotonics, Department of Physics, Engineering Physics, and Astronomy, Queen's University, Kingston, K7L 3N6, ON, Canada.

²Optics and Photonics Centre, Indian Institute of Technology Delhi, New Delhi, 110016, India.

³National Institute of Standards and Technology, Boulder, 80305, CO, USA.

⁴Smith Engineering, Department of Electrical and Computer Engineering, Queen's University, Kingston, K7L 3N6, ON, Canada.

⁵Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, 15261, PA, USA.

*Corresponding author(s). E-mail(s): 15zg11@queensu.ca; shastri@ieee.org;

Contributing authors: aadhi@iitd.ac.in; ammar.ibrahim@queensu.ca;

adam.mccaughan@nist.gov; atait@ieee.org; nathan.youngblood@pitt.edu;

sonia.buckley@nist.gov;

[†]These authors contributed equally to this work.

Abstract

Neuromorphic hardware is emerging as a transformative platform for artificial intelligence and machine learning, offering unprecedented computational speed and energy efficiency. Although several analog neuromorphic photonic systems have been demonstrated in recent years, state-of-the-art architectures still face key challenges, particularly in achieving real-time online training with high-speed signals, fully on-chip end-to-end analog implementation, online tracking, and adaptive learning under dynamic environmental conditions. Here, we demonstrate the first hardware realization of an on-chip photonic multiplexed gradient descent (MGD) algorithm that performs in-situ, on-the-fly training directly in the analog domain, which significantly reduces computational overhead and latency. Our monolithically integrated foundry-fabricated silicon photonic processor uses a microring resonator weight-bank array and a balanced photodetector with in-situ gradient computation for implementing the analog end-to-end online training. The direct gradient measurement is achieved via a perturbative approach, without requiring conventional high-precision optical or digital matrix multiplication. The experimentally demonstrated photonic implementation of MGD features fully analog, end-to-end adaptive learning, allowing simultaneous inference and training on the same hardware platform. As a result, the photonic processor can learn locally and dynamically adjust to adopt its training parameters, even in complex and varying environments. For example, our classification results demonstrate robust performance under the presence of temperature fluctuations, crosstalk and limited bit precision. This work bridges the gap between analog neuromorphic processors, online training, and practical deployment in next-generation artificial intelligence, which potentially advances autonomous, real-time photonic neuromorphic systems.

Keywords: Analog Training, Online Tracking, Adaptive Training, In-Situ Training, Online Learning

1 Introduction

Artificial intelligence (AI) technology has advanced rapidly across many fields, including materials science, natural language processing, healthcare, computer vision, and autonomous vehicles [1–5]. Brain-inspired neural network architectures, such as deep, recurrent, and convolutional networks that learn patterns from data to solve a wide range of computational problems, including complex classification, object detection and prediction. Unlike conventional digital processors, photonic neural networks offer several inherent advantages, such as low latency, massive parallelism, and high bandwidth. Efficient training of these photonic neural networks is implemented via iterative parameter updates (weights) using gradient-based (e.g., backpropagation) and non-gradient based (e.g., evolutionary methods) algorithms [6–8]. The gradient-based methods are computationally efficient, allow fast convergence and scale effectively to very large models. Typically, inference and training are two-step processes: inference is performed first, and the error is then calculated separately to update the weights in most of the in-situ neuromorphic hardware platforms [9–12]. However, the simultaneous execution of inference and training is critical to achieving fully autonomous analog learning [11, 13–15]. Recently, various analog neuromorphic hardware systems have attempted to bridge this gap by allowing in-situ gradient computations using digital computers directly in analog hardware, including variants of backpropagation, online pruning, and other analog-compatible algorithms [16–20]. Despite the in-situ digital platform significantly reducing communication overhead and latency, it demands high-performance GPUs, high-bit precision, and longer training times. However, training neural networks in modern AI technology is challenging due to the need for large-scale parameter optimization, hyperparameter tuning, high computational cost, continuous learning, and real-time adaptation, thereby necessitating the implementation of training algorithms on the hardware itself. To address this, we implement a fully analog, end-to-end in-situ training system capable of (i) real-time online learning, where training and inference are executed concurrently; (ii) on-the-fly training, where training initiates immediate updates to new input targets; (iii) adaptive correction to environmental variations without interrupting inference; and (iv) system monitoring and error tracking with desirable accuracy.

Recent advances in integrated photonic architectures, such as Mach-Zehnder interferometric (MZI) meshes, microring resonator arrays, metamaterials, MZI modulators and nanostructures, have shown fast vector-matrix multiplications (MVM), thus can significantly improve computational efficiency for a wide range of applications, including image processing, natural language processing, signal processing, autonomous systems and robotics [9, 22–26]. These photonic hardware architectures support the implementation of direct training algorithms in the optical domain, such as backpropagation, forward propagation (error propagation), equilibrium propagation, direct feedback alignment, finite-difference methods, and using various on-gradient approaches for autonomous in-situ learning [27]. In particular, the error propagation approach is one of the most efficient methods for training neural networks, in which input data is propagated forward while errors are propagated backwards or forward at layer by layer to compute the gradient. Although these algorithms implemented in photonic systems have shown promising solutions, they have limitations in real-time applications and adoption for training due to the absence of simultaneous inference and weight updates. The major drawback of error propagation algorithms is that they are efficient only for feed-forward neural network architectures. They are layer-dependent, require the chain rule to compute gradients, and impose hardware-specific constraints. The direct feedback alignment replaces the chain rule with a random weight matrix for gradient computation in error propagation but suffers from validation and shallow neural network limitations. Physical local learning is an alternative algorithm to error propagation that updates parameters using only locally available information. Similarly, finite-difference methods can directly measure the gradient but are limited in scalability. These algorithms increase computational complexity as the system scales. Alternatively, non-gradient approaches, such as genetic and particle swarm optimization algorithms, are widely integrated into the system via software-based in-situ training. However, convergence is highly challenging on these architectures, particularly at large scale. As a result, achieving efficient training on large datasets, dynamic inputs, or high-speed signals with hardware-agnostic algorithms remains a major challenge.

Here, we experimentally demonstrate the first hardware realization of a multiplexed gradient descent algorithm (MGD) on a photonic on-chip microring resonator weight bank array, allowing in-situ, on-the-fly training directly in the analog domain. This approach significantly reduces computational overhead and latency compared to conventional digital training [21]. We demonstrate the photonic implementation of the MGD algorithm on a foundry-manufactured, CMOS-compatible

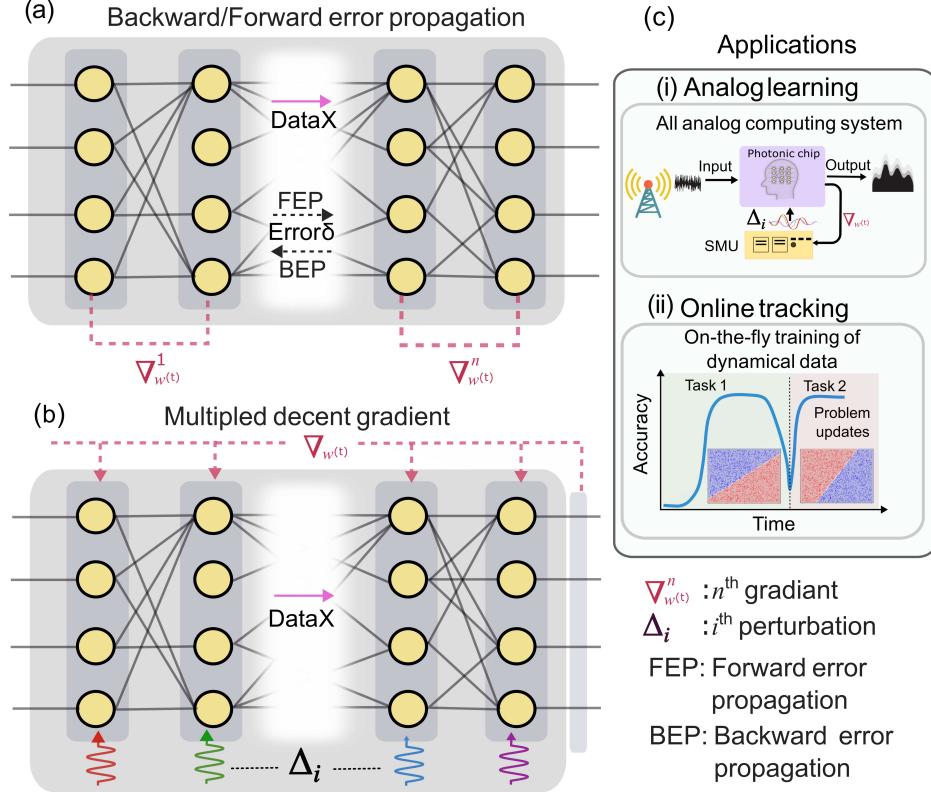


Fig. 1 Artificial intelligence training approaches and the analog end-to-end adaptive training system: a) Widely adopted training methods implemented on analog learning hardware platforms using an error propagation. b) A perturbative approach demonstrated by the multiplexed gradient descent algorithm [21]. This perturbative approaches can be applied to a wide range of machine learning models, including feedforward neural networks and recurrent neural networks. c) These models give rise to a variety of real-life applications, enabling hardware-agnostic analog learning directly using analog signals. This further enables our analog hardware to use its high bandwidth and low latency in online tracking tasks, and creates the flexibility to adapt to various system- and platform-level parameter changes such as crosstalk.

silicon photonic integrated circuit (PIC), achieving analog, end-to-end adaptive learning with simultaneous inference and training on the same hardware. The gradient is computed by applying perturbations to the network parameters and measuring the loss function via the intrinsic dynamics of the device. In this approach, the gradient is measured directly at the output as an analog signal, bypassing costly digital computation. During training, the architecture physically computes the gradient and performs local weight updates entirely within the analog domain without the requirement of any knowledge of the internal structure of the network. Our system can classify linear tasks with $> 90\%$ accuracy and nonlinear tasks with $> 80\%$ accuracy within a few iterations, while also demonstrating real-time online tracking at a gigabaud inference rate. We also demonstrate that our system autonomously adapts to environmental changes, such as temperature drifts and internal thermal crosstalk, by dynamically adjusting training parameters in real-time without external intervention, thereby maintaining operational continuity with minimal downtime. The MGD implementation is hardware-agnostic and can therefore be applied to a wide range of machine learning models and real-world applications. Our platform is robust to fabrication variations and system noise, does not require sophisticated calibration procedures [28]. In contrast to other coherent architectures, our incoherent WDM-compatible implementation of the MGD algorithm offers enhanced scalability and stability while relaxing hardware constraints toward real-time, self-adaptive, analog end-to-end learning of a neuromorphic photonic architecture for the next-generation AI.

2 Hardware implementation

Figure 1 illustrates both conventional error propagation training approaches (backpropagation and forward propagation) and the current implementation of MGD-based analog end-to-end adaptive training system. As shown in Fig. 1 (a), the error propagation approaches are used to calculate the

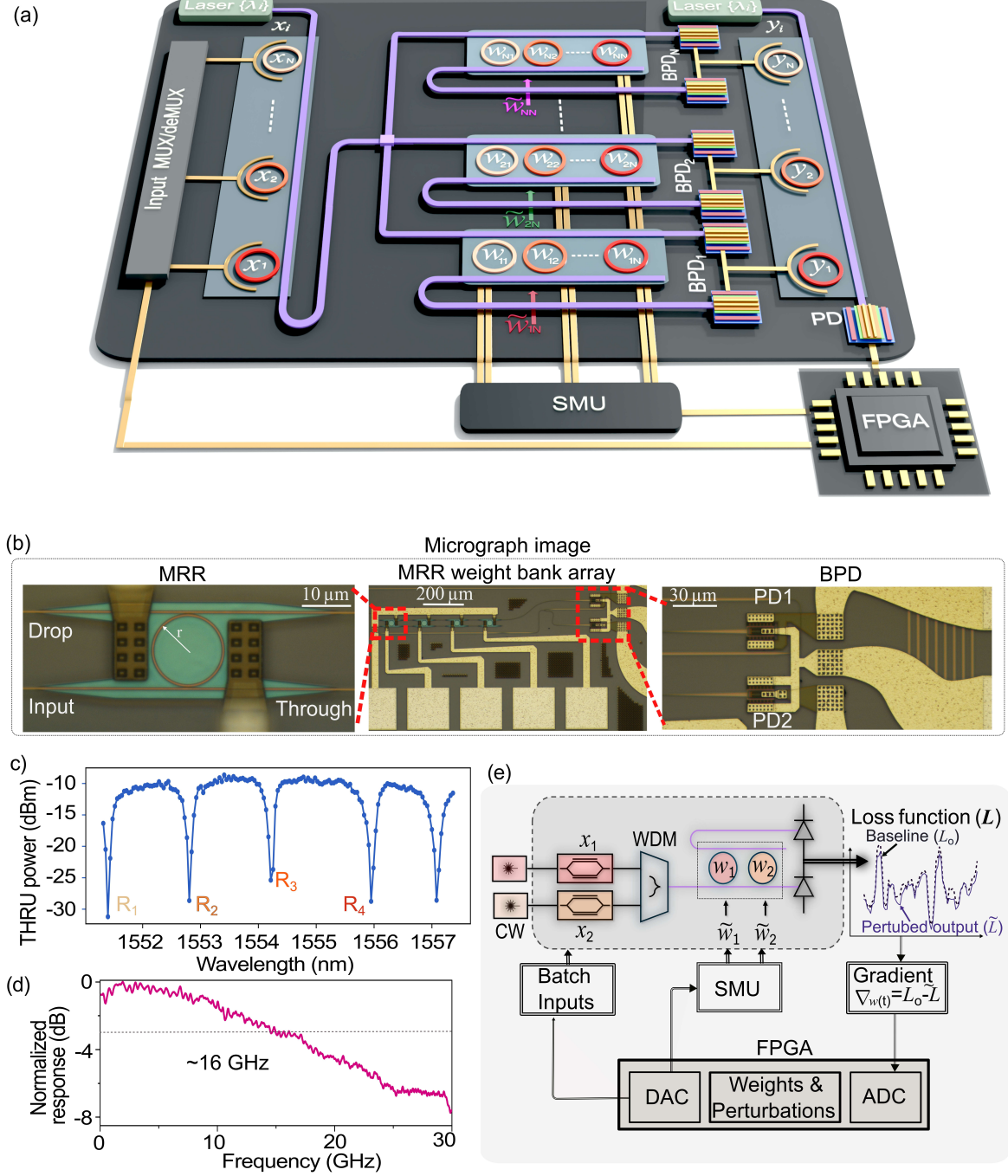


Fig. 2 Schematics for experiment setup and device characterizations: a) Our vision for a wavelength-multiplexed integrated neuromorphic photonic hardware that uses microring resonator (MRR) weight bank and on-chip balanced photodetectors (BPDs). The inputs x_i are amplitude-encoded on respective wavelength channels using i^{th} microring modulators, all of which are broadcast to all neurons in the same layer. Each weight bank implements all the weights w_{NN} for the particular neuron plus the corresponding perturbations $\tilde{w}_{NN}$, and the balanced photodetector BPD_N signals are sent to N microring modulators. The biasing of the microring modulator determines the type of nonlinear activation function and produces output y_i for that neuron. b) Micrographs of the electrically-packaged integrated photonic chip and the device-under-test, with labels for the electrical connections for the weights and photodetector bias. c) Spectrum of the MRR weight bank. d) Bandwidth measurement of the integrated BPDs. e) The experimental setup used in this paper: a single-layer, 2-input and 2-weight system consists of two optical input channels, x_1 and x_2 , each modulated by a DAC on the RFSoc FPGA via a Mach-Zehnder modulator (MZM). A WDM arrayed-waveguide grating (WDM AWG) multiplexes both channels into a single waveguide, which is fed to the add-drop MRR weight bank integrated on chip. The MRRs are controlled using two SMU channels, which communicate with the RFSoc FPGA to actuate the weights w_1 , and w_2 , as well as perturbations $\tilde{w}_1$ and $\tilde{w}_2$. The weight bank optical output is detected through a pair of integrated BPDs, and the output signals L_0 and $\tilde{L}$ are measured with an ADC channel on the RFSoc FPGA.

gradient at every layer, however, implementing it in hardware is not the most efficient way to compute the gradient, as it leads to huge MVM operations. In addition, error propagation is required to form symmetric backward/forward connections and perform synchronized computations, thus considered as “biologically implausible” and incompatible with large-scale hardware systems. On the other hand, the MGD algorithm, shown in Fig. 1 (b), introduces random perturbations to each weight and estimates the gradient at the output. The method is less sensitive to initialization and gradually converges toward the minimum loss function, thereby avoiding issues such as exploding gradients. Our perturbative approach is based on finite-difference approximations, converges rapidly (within a few iterations), requires minimal memory, and is highly robust to noise. Figure 1 (c) shows the native implementation of the hardware-agnostic analog learning algorithms for real-world applications, including fully end-to-end analog learning and online tracking. The Fig. 2(a) shows the schematic illustration of the training algorithm implemented on a wavelength division multiplexing (WDM) platform within a fully integrated on-chip photonic system. The integrated photonic hardware architecture uses microring resonator (MRR) weight bank and on-chip balanced photodetectors (BPDs) in a wavelength-division multiplexed scheme. The inputs x_i are amplitude-encoded on respective wavelength channels using i^{th} input microring modulators. An $N \times N$ microring weight-bank array implements the weights w_{NN} plus the corresponding perturbations $\tilde{w}_{NN}$, and the balanced photodetector BPD_N performs the MVM operation in the optical domain. Typically, the nonlinearity can be accessed through the photodetector’s quadratic response or, as shown in Fig. 2(a), by using microring modulators.

The micrograph of the fabricated WDM photonic neural network, including the microring resonator weight bank and balanced photodetectors (BPD), is shown in Fig. 2(b). The drop port responses of the microring weight bank array consist of four MRRs and the measured bandwidth of the designed single-mode germanium photodetector is shown in Fig. 2 (c) and (d), respectively. As shown in Fig. 2(c), the ~ 5 dB variation in extinction ratio across the ring throughputs arises from fabrication imperfections and wafer-level thickness variations. Typically, such nonuniformities can be compensated by adjusting the input power. However, because the MGD algorithm is hardware-agnostic, variations in the MRR extinction ratios are inherently incorporated into the gradient calculation and corresponding weight updates, assisting the training approach to maintain strong performance despite non-ideal hardware conditions. Fig. 2(e) presents a schematic of the actual experimental setup. Although our monolithic integrated photonic chip comprises four MRRs and an integrated balanced photodetector. To demonstrate the feasibility of the MGD algorithm, we implemented a proof-of-concept system comprising two MRR weight-bank arrays and a BPD. The temporally varying input signals are amplitude-encoded onto two distinct wavelength channels and sent them into the MRR weight bank for weight updates and training. Each microring resonator has an independently tunable weight, configured for a specific wavelength channel, and a continuous perturbation is applied to the weight value. In our experiment, thermo-optic heaters in the MRRs are controlled independently using separate source-measure units (SMUs), which set the voltage for each MRR. These voltages actuate MRR resonances via in-resonator photoconductive heaters (IRPHs) using the thermo-optic effect [29]. The precise control of individual MRR transmission is important to achieve the effective weight precision. In our system, thermo-optic heaters precisely control each MRR, tuning transmission from -1 to +1. The current setup provides an effective resolution of 6.72 bits without additional stabilization or system-level modifications. A dithering-based control scheme can further improve bit precision to 9 bits [30]. The weighted input values are summed at the balanced photodetector, resulting in an efficient matrix/vector multiplication in the electrical domain. Although the output loss function can be obtained as an analog signal, we employ digital-to-analog converters (DACs) and analog-to-digital converters (ADCs) integrated within an RFSoc FPGA to enable fast input encoding, controlled weight updates, and gradient measurements. The input data are processed in batches, therefore, the input signal rate must exceed the weight update rate by a factor equal to the batch size. Here, the bandwidth limitation of the DAC restricts the input signal rate to 1 GBaud with 8 samples per symbol, while the ADC samples at 4 GS/s. The FPGA is used to provide experimental flexibility in input encoding and signal detection; however, by replacing the FPGA with a high-speed arbitrary waveform generator and substituting thermo-optic heaters with p-n junction-based modulators, the system could support significantly higher input data rates (up to 16 GBaud) and faster weight-modulation speeds (on the order of sub-microseconds). The incoherent WDM broadcast and weight architecture eliminate the need for precise control of phase shifters, mitigating interference and instability commonly encountered in architectures based on coherent matrix multiplications. Further details of the experimental setup are provided in the Methods section.

Fundamentally, the algorithm implements a zeroth-order gradient descent and estimates the gradient from perturbations introduced to the network parameters. In the MGD algorithm, perturbation is applied to each weight individually. This perturbation affects the loss function at the output, and the resulting change is used to update the weights. If an independent perturbation is applied at each weight values, the amplitude of the overall perturbation is multiplexed, resulting in an immediate change in the loss function. As a result, if the parameters are independently modulated at frequencies $\omega_1, \omega_2, \omega_3$, etc., it will induce variations in the loss function. In the multiplexed gradient descent algorithm, all the parameters are being perturbed and updated simultaneously. Although the weight perturbations can take any pairwise-uncorrelated time-varying functions, such as sinusoidal, square, or randomly generated, we used a randomly generated binary perturbation for simplicity. The perturbation $\tilde{w}_i(t)$ follows a random binary sequence of $(-1, 1)$, with an amplitude of Δw_i . Each weight and its perturbation is implemented as a voltage signal applied to the MRRs in the weight bank. At the start of each training epoch, we upload all samples from the first batch to the DACs on the FPGA, which then modulate the input samples onto the optical carriers. As training progresses, the random binary sequence generates a time-varying perturbation signal, causing corresponding variations in the loss function measurements. The instantaneous loss function variations ($\Delta L_i(t)$) consist of a perturbation-free baseline loss function L_0 and a perturbed loss function $\tilde{L}_i(t)$ generated by the weight perturbation, $\tilde{w}_i(t)$, at each neuron. The loss function is computed using the network output, $y(t)$, and the training target $\hat{y}(t)$. The weight update w_i incorporates the instantaneous measurements of the variation component of the loss function and its corresponding weight perturbation. Afterwards, we repeat the same procedure and iterate through the rest of the batches to complete an epoch, and progress through all epochs to finish training. Further details on the derivation of the MGD algorithm for direct gradient measurement, the experimental setup, and the training workflow are provided in the **Supplementary Sections 1 and 2**.

3 Results

3.1 Online tracking

Typically, training is performed once on a fixed dataset, where all training data are available beforehand to ensure efficient learning. However, real-time forecasting, anomaly detection, object tracking, and adaptive control systems are time-sensitive tasks that require continuous monitoring of dynamically evolving data [31–33]. As a result, the newly encountered data frequently deviate significantly from the training data due to variations in operating conditions, environmental factors, or underlying data distributions. In such situations, many conventional algorithms experience a substantial degradation in prediction and classification performance. Furthermore, these methods generally learn each task independently, often requiring additional memory and computational resources for retraining, need rapid and continuous parameter updates, dynamic learning rules, and operation in varying conditions, while ensuring timely decision-making. Implementing a hardware-agnostic algorithm capable of adapting to environmental variations typically involves updating samples, modifying learning rules, and/or adjusting operating conditions. We successfully implement an online tracking algorithm, where the hardware updates network weights on-the-fly as soon as new data arrives. Here, the online tracking algorithm is achieved due to the advantages of simultaneous training and inference, while the learning rule for classification continuously adapts. Our MGD framework enabling continuous tracking and adaptation to evolving data while reducing the need for extensive retraining and extra memory. The dataset consists of 2,000 samples uniformly distributed in a 2D space, with the correct separation between the two classes evolving over time as reflected by updates in the learning rule signal.

To explain the dataset and learning rule in more detail, we first use a linear classification as an example, where the learning rule is defined by an arbitrary line ($a \cdot x_1 - b \cdot x_2 = 1$), where a and b are real numbers. The parameters a and b are defined according to the learning rule to classify a given set of data. When the learning rule changes, that is, when the coefficients a and b change, the processor subsequently classifies the data based on the updated learning rule. Furthermore, when the data set changes or evolves dynamically, the processor continues to classify the data based on the defined learning rule. In the experiment, all the network parameters are initialized to random values, and the gradient is measured by comparing baseline and perturbed loss function values, without and with perturbations to the network parameters, respectively. These parameters are updated after each

batch, and training continues until convergence is reached. Preliminary training results of linear and non-linear classification tasks are given in the **Supplementary Section 3**.

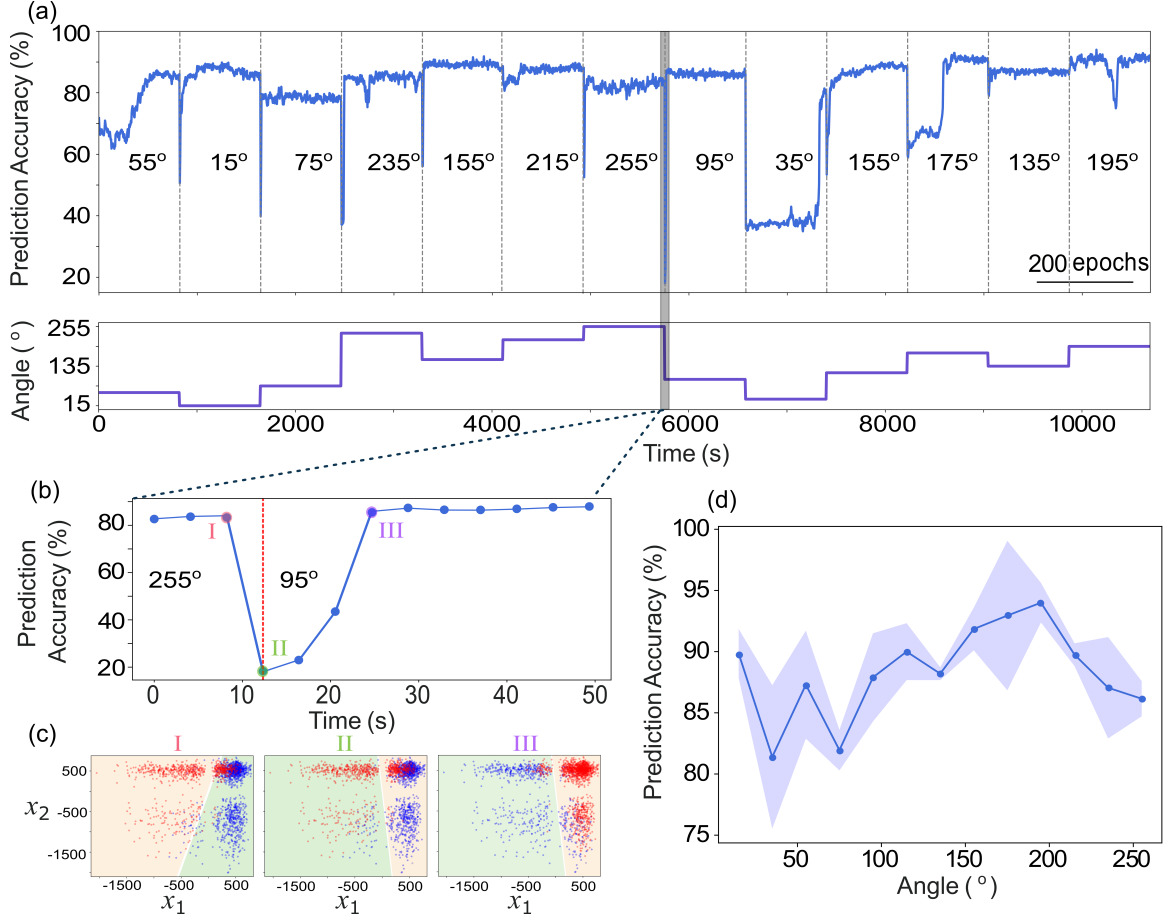


Fig. 3 Online tracking results: the separation rule rotates through a sequence of three different angles calculated with respect to the x-axis, and we demonstrate our system successfully adapts to the logical changes. a) Training accuracy values measured experimentally at thirteen unique angles ranging from 15° to 255° with a step size of 20° , updated after every 200 epochs. The thirteen different angles are randomly shuffled, and are plotted against time in the subplot here. b) Zoomed-in plot of the epochs near the transition from 255° to 95° , together with three scatter plots showing the system output before, at, and after the angle change. c) Best prediction accuracy achieved at every angle, plotted with the corresponding standard deviation recorded.

To experimentally demonstrate the online tracking task, we varied the classification criteria every 800 seconds, equivalent to 200 training epochs, without providing any additional information to our analog system during training. Initially, the separation between the two classes was set at a 55° angle relative to the horizontal axis. Subsequently, the separation between the two classes was rotated through 13 randomly shuffled configurations, with angles ranging from 15° to 255° in 20° increments. The sequence of shuffled angles is illustrated in the subplot of Fig. 3(a). At the beginning of each angle change, there is an initial drop in prediction accuracy, as seen in Fig 3(a). However, the system quickly recovers to the new learning rule, achieving a prediction accuracy of approximately $>85\%$ within a few epochs (<10) for most angles. We also estimate the training time, the processor has a recovery time of 20 seconds whenever a new learning rule is updated. A zoomed-in plot of the 13 epochs around the transition from 255° to 95° , shown in Fig 3(b), confirms that our processor restores its prediction accuracy within just 4 epochs which is 20 seconds. The corresponding data sets at three instances are shown in Fig 3(c). Figure 3(d) shows the best prediction accuracy achieved at every angle, plotted with the corresponding standard deviation. However, we occasionally observe a momentary decrease in prediction accuracy after the system has reached convergence. While such fluctuations can be mitigated through fine-tuning hyperparameters, such as perturbation amplitude and learning rate, they cannot be entirely eliminated due to the inherent nature of the perturbation-based approach.

Notably, such brief moments of instability only occur sporadically and have a negligible impact on overall system performance.

3.2 Real-time adaptation

The traditional neuromorphic hardware uses static machine learning models, thus fails to operate in real-time, adopt and respond to dynamic environmental variation. As a result, the need for adaptive real-time learning, which incorporates continuous learning, context awareness, real-time response and automated decision-making is highly in demand. Typically, employing a photonic processor for training can compromise reliability due to environmental temperature fluctuations and thermal crosstalk between neighbouring heat-actuated weights or heat dissipation in the processor. Although performance can be partially improved through error-correction algorithms and hybrid analog-digital signal processing units, we demonstrate that the MGD algorithm integrated with the MRR weight-bank based analog system continuously adapts and readjusts in real time, supporting robust and stable performance. Our photonic processor provides fully integrated self-learning and self-correction capabilities, enabling dynamic adaptation to both external perturbations (e.g., global temperature or pressure variations) and on-chip non-idealities (e.g., thermal crosstalk) in real time.

As a proof of concept to demonstrate the real-time adaptation capability, we varied the temperature of the photonic chip using a temperature-controlled stage while keeping the learning rule fixed (i.e., fixed a and b). Typically, MRR-based photonic processor highly sensitive to temperature variations, therefore a temperature variation of just 2°C in the system can introduce up to a 50% change in the weight values. Initially, we trained the system at 25°C for 200 epochs, and the prediction accuracy reached approximately 90%. When the temperature was subsequently increased by 2°C , the prediction accuracy immediately dropped to below 50%, as the temperature change significantly altered the effective training parameters. However, due to a high loss value in the measured output gradient, the MGD training algorithm readjust the individual microring resonators to restore their weight actuations and minimize the loss function. As a result, after 200 epochs, the system recovered its original performance, achieving approximately 90% prediction accuracy. As shown in Fig. 4(a), the chip temperature was randomly varied between 23°C and 27°C . The training accuracy, measured every 200 epochs, shows that the system can tolerate global temperature variations of up to $\Delta T = 4^\circ\text{C}$ and then quickly recover its performance in less than 10 s. The variation between the optical weight values of 1 and -1 corresponds to actuation voltages of 0 V and 2 V, respectively. The mapping between the applied voltage and the resulting optical weight is approximately linear within this range. Due to its adaptive functionality and robustness, the hardware platform can be deployed under adverse, dynamically changing operating conditions while maintaining consistent performance.

Similarly, to demonstrate thermal crosstalk, we considered a 3-MRR-based weightbank architecture. We used MRR1 and MRR3 for classification by implementing weights, while MRR2 introduced temperature variation by injecting a steadily increasing bias voltage into the heater. As a result, MRR2 introduces thermal crosstalk on MRR1 and MRR3. To better understand the effect of thermal crosstalk on the weights w_1 and w_2 due to MRR2 and its temperature variation ΔT , we estimated the rate of equivalent weight change as $\Delta w_o/t_T$, where Δw_o is the equivalent percentage change in the optimal weight value due to the temperature change in MRR1 and MRR3 due to MRR2, and t_T is the time between temperature changes measured in minutes. Because the learning rule is the same (fixed a and b), the optimal weights will have the same numerical values despite changes to the physical parameters.

Further, we have investigated the system stability against thermal crosstalk and the recovery of training accuracy after a few epochs. Unlike Fig. 4(a), which illustrates the impact of a global temperature change due to environmental variation on the weights and the training process, Fig. 4(c) demonstrates the training and stability of the system in the presence of thermal crosstalk arising from neighbouring MRRs. We steadily increased the bias voltages of two neighbouring channels from 0 V to 5 V after the initial 200 epochs, with the bias voltage increasing by 25 mV per epoch to introduce thermal crosstalk, as shown in Fig. 4(c). Here, the average time for the temperature variation is approximately 4.8 minutes, and we found that the rate of equivalent weight change is around $23.7\% \cdot \text{min}^{-1}$ for weight 1 and around $5.48\% \cdot \text{min}^{-1}$ for weight 2. As shown in Fig. 4(c), instability occurs over a brief period as the bias voltages ramp up; however, our system quickly recovers and remains resilient to the effects of thermal crosstalk. This indicates that our system is capable of compensating thermal crosstalk due to any kind of heat dissipation in the processor, which induces temperature changes in the weight actuations. Currently, the weight actuation variations of

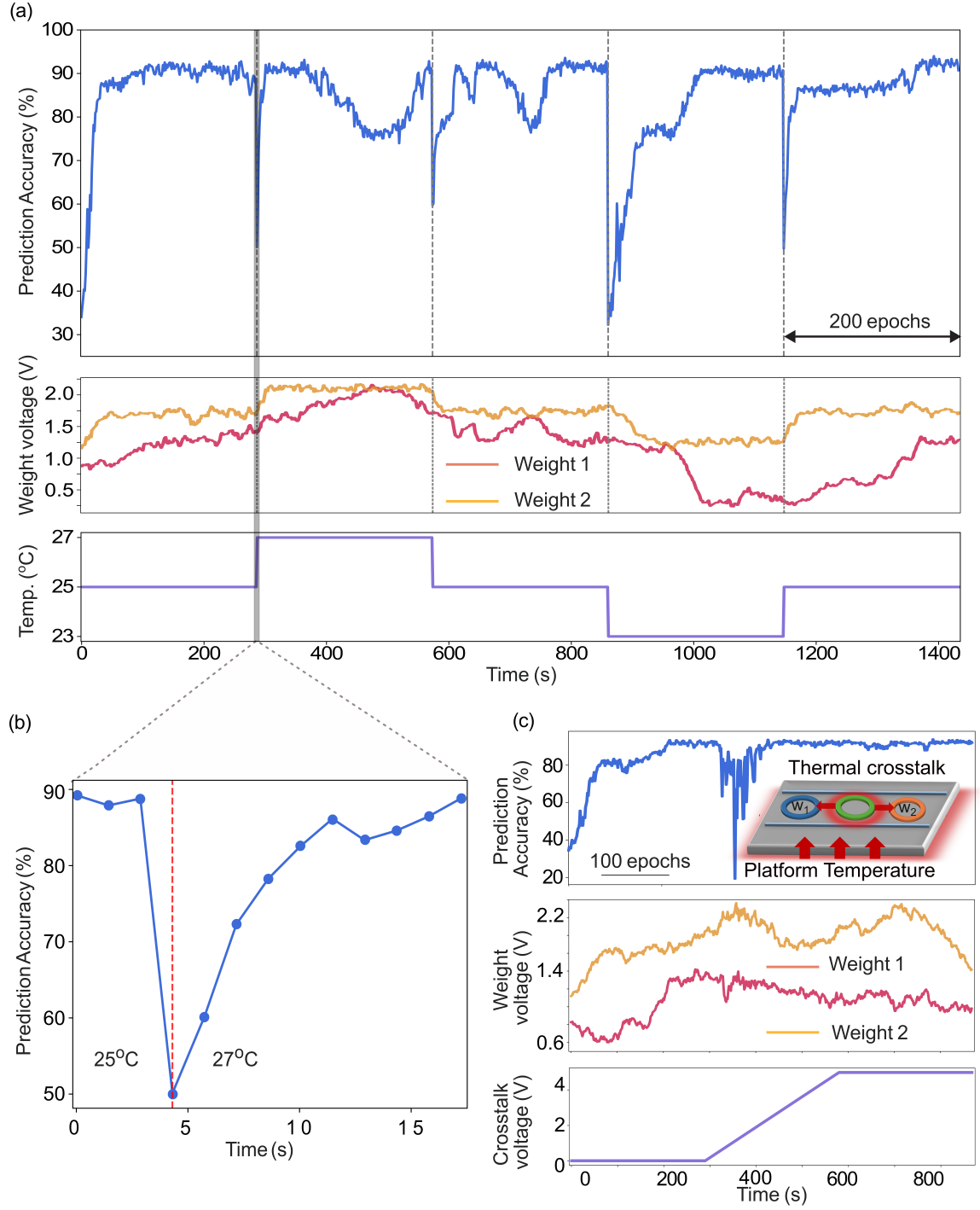


Fig. 4 Real-time adaptation results: we demonstrate the excellent stability of our system against two major environmental variations. a) we apply temperature changes to the test platform by varying the target temperature set by the temperature controller every 200 training epochs. The temperature starts at 25°C, and goes to 27°C, 25°C, 23°C, and 25°C subsequently. We record training accuracy after each epoch and provide a zoomed-in plot of the epochs near the first temperature transition from 25°C to 27°C. At 25°C, the actuated weight voltage range of [0, 2] V corresponds to a measured optical weight range of [-1, 1]. c) We introduce on-chip thermal crosstalk by injecting a steadily increasing bias voltage to the channels neighboring the ones under test. Starting from the 200th epoch, the bias voltages to the neighboring two channels are increased from 0 V to 5 V over a period of 200 epochs, with a step size of 25 mV per epoch. The thermal crosstalk persists after the bias voltages reach the maximum value for the rest of the trial. c) Weight voltage values recorded throughout the thermal crosstalk demonstration, together with the bias voltage supplied to the neighboring channels to generate thermal crosstalk.

more than 20% per minute can be compensated, but the rate could be further improved by varying the learning rate.

4 Discussion

To further support the scalability of the system and its potential to solve more complex computational tasks, we implemented a digital twin of our analogue end-to-end adaptive photonic system incorporating a multi-layer network and the MGD training algorithm. Digital twin is a digital replica of the system, enabling a connection between physical and digital environments. These virtual models of physical systems are typically used for simulation, analysis, and optimisation of their real-world counterparts, thus representing a close approximation to practical scenarios. Our digital twin model comprises a network of 30×30 microring resonators that can serve as a general-purpose architecture. Although the architecture is capable of solving various computational problems, we used it to classify the Yin-Yang dataset (see Fig. 5(a)) [34]. The network requires an input layer of 4 nodes, a hidden layer of 30 nodes, and an output layer of 3 nodes, totalling 37 fully connected feed-forward neural network units. The array of ring resonators incorporates weight values across wavelengths, sums the weighted inputs at the photodetectors, and applies a nonlinearity at the output, which emulates one photonic neuron in the model. To implement a replica of the experimental architecture, we have used experimentally extracted device parameters and measured lookup tables for MRRs, photodetectors, and the electro-optic nonlinearity of the MRR modulator, along with various perturbation signals, such as sinusoidal and binary random perturbations, to investigate training efficiency [35–37]. The MGD training uses sinusoidal weight perturbations generated by the 16-bit SMU DAC, with each of the 1800 heater currents dithered at a distinct, frequency-multiplexed tone and evaluated over the 500-sample batch. Fig. 5(b) shows the prediction accuracy of the physical model of the photonic core, which uses each epoch at 33 microseconds. Typically, the hardware introduces various system noise, losses and errors, including shot noise, TIA, and laser RIN noise at both detectors, inter-ring thermal crosstalk, quantization errors and optical insertion loss, which lowers the detector SNR through the link-budget TIA gain. Incorporating all these noises and errors in our digital twin deviates from an ideal system, yet the network still classifies Yin-Yang with 95.4%. The classification results and the corresponding confusion matrix at the end of training are shown in Fig. 5(c) and Fig. 5(d), respectively. For comparison, implementing the trained model using a GPU (NVIDIA A100) achieved 97% accuracy for the same Yin-Yang classification task. The deviation between the digital twin and the simulation accuracy is roughly 2%, arising from the physical hardware itself, mainly due to system-level noise, bounded resonance nonlinearity, and the limited dynamic range of the microring weights. The photonic core has an update time of $32.3 \mu\text{s}$, whereas the NVIDIA A100 has $54.4 \mu\text{s}$ and additional computational overhead. It is also worth noting that the software-based simulation uses 64-bit precision, whereas our digital twin uses only 16-bit precision and a reduced update time to achieve similar performance. Despite its much lower bit precision, the photonic system shows performance comparable to that of the digital processor due to the analog nature of our photonic processor. These findings validate the scalability and feasibility of perturbative training techniques for photonic circuits in larger-scale networks. The **Supplementary Section 4** provides further details on the digital twin and the training procedure.

Further, we have also analyzed the latency of our demonstrated analog photonic hardware platform. Our photonic processor has a total length of 1 mm, as a result, the propagation delay is 11.7 ps. When accounting for the response time of high-speed photodetectors (16GHz), the total end-to-end inference latency remains low at ~ 62.5 ps, which exclude the response time of high-speed ADCs and DACs. The total end-to-end latency is approximately 1.5 seconds for a larger learning rate and perturbation amplitude, and approximately 4 seconds when using smaller parameters (see the **Supplementary Section 5**). In the current configuration, the weight elements are being temperature-controlled, contributing to the observed convergence time in the order of a few seconds. However, our photonic chip also supports carrier-depletion PN junction for electro-optic weight tuning, which provides a typical bandwidth of ~ 16 GHz [38, 39]. This enables rapid weight adjustments and potentially reduces convergence times by three orders of magnitude.

We estimate that our monolithically integrated photonic chip has the total power consumption of 27.4 mW, including photonics, drivers, and readout electronics, resulting in an energy efficiency of 853 fJ/Op, equivalent to 1.2 TOPS/W when operating at a 16 GHz. See the **Supplementary Section 6** for a detailed breakdown of the power budget and energy efficiency of individual components. For the $4 \times 30 \times 3$ neural network architecture (see Fig. 5(a)), each neuron is implemented with four MRRs,

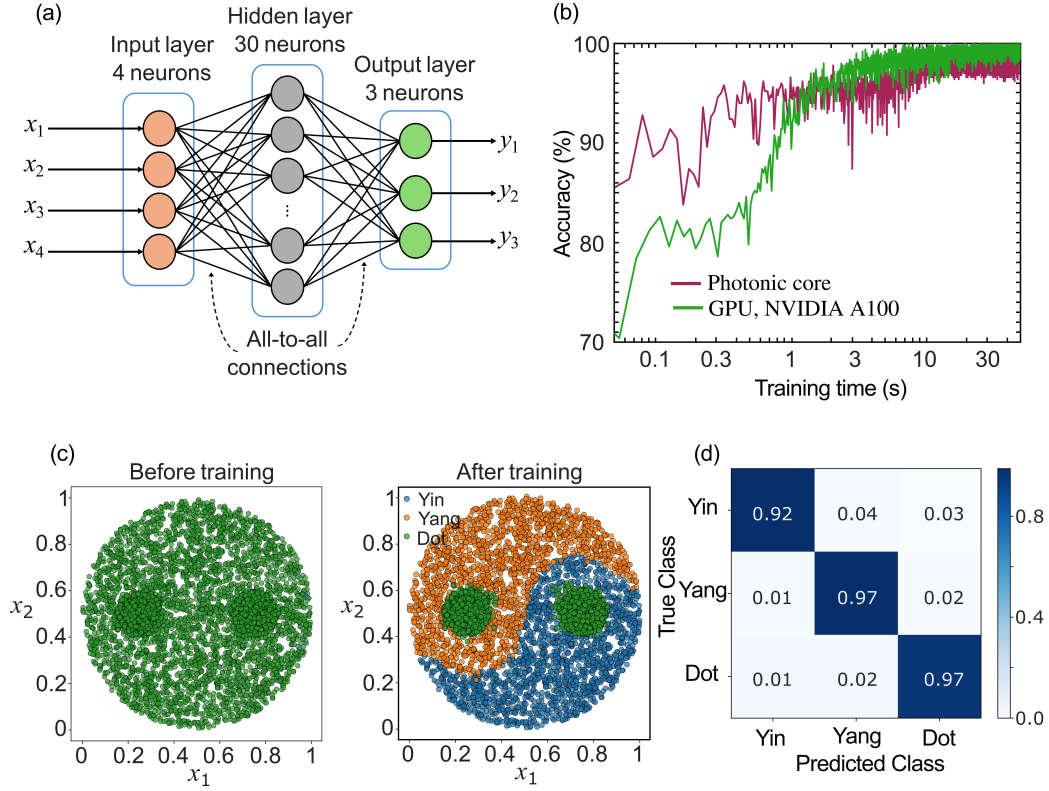


Fig. 5 Yin-Yang classification of the digital twin model with multi-layer networks: a) The digital twin of the analog end-to-end adaptive training system. It is a fully connected feedforward network contains one input layer with 4 neurons, one hidden layer with 30 neurons, and one output layer with 3 neurons. All neurons in both the hidden layer and the output layer use an electro-optic nonlinearity. b) Training accuracy results obtained from our digital twin model, reaching a maximum accuracy of 95.4 %. c) Snapshots of the classification outputs at the beginning of the training (0 seconds) and at the best accuracy value (16.2 seconds). d) Confusion matrix at the end of the training simulation.

Table 1 Comparisons with other contemporary works.

	Input size	Architecture	Analog inference latency	Analog training latency	Energy efficiency	Online tracking ¹	Real-time adaptation ²
This work	2	IP	62.5 ps	22 ms	70 GOPS/W	Yes	Yes
Ref. [40]	430k	FP	192 ps	Not given	0.12 GOPS/W	No	Yes
Ref. [11]	6	IP	410 ps	Not given	13 GOPS/W	No	No
Ref. [14]	4	IP	1 ns	Offline training	5,000 GOPS/W	No	No
Ref. [41]	16k	FP	Not given	64 ms	5.4×10^9 GOPS/W	No	No
Ref. [42]	30	IP	570 ps	Digital training	71.4 GOPS/W	No	No

¹Online tracking of updates to the training problem or learning rule.

²Real-time adaptation to external and internal parameter changes, including environmental temperature variations and neighboring channel thermal crosstalk. IP- Integrated photonic platform, FS- Free-space architecture.

performing 243 operations ($5 \times 30 + 31 \times 3$) per inference cycle, yielding an energy efficiency of 3.4 fJ/Ops, corresponding to 291 TOPS/W. A comparison of on-chip and free-space optical computing architectures is presented in Table 1. Moreover, we highlight that integrating phase-change materials (PCMs) in our architecture could push energy efficiency to the PetaOps/W range while significantly enhancing information density [43]. This capability paves the way for ultrafast, power-efficient, and low-cost neuromorphic computing platforms, addressing both scalability and real-time adaptability in photonic machine learning.

5 Conclusion

We have experimentally demonstrated an analog end-to-end adaptive neuromorphic hardware system that physically measures and locally updates the gradient of the dynamical system directly in the optical domain. Unlike conventional digital implementations, which are constrained by the limitations of digital matrix multiplications, our platform achieves fully analog gradient measurement through multiplexing and perturbative updates at the output. This approach effectively bypasses digital precision bottlenecks and enables high-accuracy inference while maintaining robust real-time adaptability. Our demonstration addresses critical, longstanding challenges that have limited the practicality of neuromorphic hardware architectures, particularly the inability to perform on-the-fly training with dynamic data and the lack of real-time adaptation to crosstalk and environmental fluctuations. Implemented on an integrated photonic platform, our system achieves favourable performance in online tracking tasks, reaching 86.5% accuracy while dynamically adapting to learning rules. Notably, our system ensures stability even when training continues past convergence, eliminating the need for manual calibration or active stabilization. This intrinsic stability allows our photonic processor to self-adapt to changing conditions with fast convergence, effectively mitigating the impact of system noise. We further evaluated the performance of the system in terms of scalability, latency, and energy efficiency. Our photonic system achieves an ultra-low latency of 62.5 ps and an energy efficiency of 853 fJ/Op, highlighting its potential for large-scale neuromorphic networks. Our fully analog processor significantly reduces the complexity associated with training neuromorphic hardware, paving the way for scalable implementations in integrated photonics. The demonstrated online adaptive training approach is versatile and broadly applicable across a spectrum of machine learning platforms. Crucially, it lays the groundwork for real-time learning in analog hardware, with transformative implications for applications in sensing, autonomous driving, telecommunications, and beyond.

6 Methods

Device fabrication

The monolithic integrated photonic processor is fabricated at the AMF foundry on a silicon-on-insulator (SOI) wafer with a silicon thickness of 220 nm device layer and a buried oxide thickness of 3 μm . The bus waveguides have a width of 500 nm. The MRRs have radii of 8.0 μm , 8.01213 μm , 8.02426 μm , 8.03639 μm , 8.04852 μm to prevent resonance overlap. Each resonator shows a quality factor of ~ 6000 and a bandwidth of 200 pm, with a physical separation of 92 μm between neighbouring MRRs. In the experiment, we have used MRR1 and MRR2 for all the classification tasks and MRR1-MRR3 for the demonstration of thermal crosstalk. The MRRs have in-resonator photoconductive heaters [29] that can actuate the weight by thermally tuning the MRR resonance. The system has an end-to-end optical loss of 12 dB, and each MRR has an estimated insertion loss of < 3 dB. Our system requires ~ 0.8 mA of heating current to change the transmission of the MRRs by 15 dB. To implement the N-doped heater, each MRR consists of a circular waveguide etched to a 90 nm thick pedestal that hosts the phosphorous dopants. A 10 μm wide N doping section is patterned to follow the MRR, outside of which heavy N++ doping is used to make ohmic contacts. The phosphorous dopant concentrations are N: $5 \times 10^{17} \text{ cm}^{-3}$ and N++: $5 \times 10^{20} \text{ cm}^{-3}$. Metal vias and traces are deposited to connect the heater contacts of the MRR weight bank to electrical metal pads. An integrated balanced photodetector sums the difference between the drop and through ports without the need for a differential amplifier, providing weighted addition of all input signals. On-chip detection also removes round-trip coupling loss by half. The BPD comprises two germanium-based photodetectors connected in series, with the n-contact of one PD connected to the p-contact of the other [44], and has an operational bandwidth of ~ 16 GHz.

Acknowledgements

This research was funded by Queen’s University (<https://ror.org/02y72wh86>), University of Pittsburgh (<https://ror.org/01an3r305>), and NIST (<https://ror.org/05xpvk416>). Thanks to Nathanael Eddy for his assistance in taking the micrographs using a modified Nikon-E microscope (Centre for Nanophotonics, Department of Physics, Engineering Physics, and Astronomy, Queen’s University).

Author contribution

BJS, ZG, and ANT conceived the idea of the analog end-to-end training system. SMB and ANM developed the MGD algorithm. ZG devised the experiments with inputs from BJS, ANT, SMB, and ANM. ZG performed the experiments and numerical simulations. AA studied the energy budget of the analog end-to-end training system. All authors analyzed the data. AA led the writing with contributions from all authors. BJS and SMB supervised the study.

Conflicts of Interest

The authors declare no conflict of interest.

Data Availability

Available upon request from the corresponding authors.

Code Availability

Available upon request from the corresponding authors.

Additional information

Supplementary information. The online version contains supplementary material available at <https://doi.org/xxx/xxx>.

Correspondence and requests for materials

Corresponding author: Bhavin J. Shastri.

References

- [1] Bywater, R.: Solving the protein folding problems. *Nature Precedings* (2010) <https://doi.org/10.1038/npre.2010.4730.1>
- [2] Zhou, L., Schellaert, W., Martínez-Plumed, F., Moros-Daval, Y., Ferri, C., Hernández-Orallo, J.: Larger and more instructable language models become less reliable. *Nature* **634**(8032), 61–68 (2024) <https://doi.org/10.1038/s41586-024-07930-y>
- [3] Chen, Y., Nazhamaiti, M., Xu, H., Meng, Y., Zhou, T., Li, G., Fan, J., Wei, Q., Wu, J., Qiao, F., Fang, L., Dai, Q.: All-analog photoelectronic chip for high-speed vision tasks. *Nature* **623**(7985), 48–57 (2023) <https://doi.org/10.1038/s41586-023-06558-8>
- [4] Bishop, C.M.: Neural networks and their applications. *Review of Scientific Instruments* **65**(6), 1803–1832 (1994) <https://doi.org/10.1063/1.1144830> https://pubs.aip.org/aip/rsi/article-pdf/65/6/1803/19141715/1803.1_online.pdf
- [5] Al-Kayed, N., St-Arnault, C., Morison, H., Aadhi, A., Huang, C., Tait, A.N., Plant, D.V., Shastri, B.J.: Programmable 200 gops hopfield-inspired photonic ising machine. *Nature* **648**(8094), 576–584 (2025)
- [6] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015) <https://doi.org/10.1038/nature14539>
- [7] Bengio, Y., Boulanger-Lewandowski, N., Pascanu, R.: Advances in optimizing recurrent networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8624–8628 (2013). <https://doi.org/10.1109/ICASSP.2013.6639349>
- [8] Krogh, A.: What are artificial neural networks? *Nature Biotechnology* **26**(2), 195–197 (2008) <https://doi.org/10.1038/nbt1386>

- [9] Zhang, W., Tait, A., Huang, C., Lima, T., Bilodeau, S., Blow, E.C., Jha, A., Shastri, B.J., Prucnal, P.: Broadband physical layer cognitive radio with an integrated photonic processor for blind source separation. *Nature Communications* **14**(1), 1107 (2023) <https://doi.org/10.1038/s41467-023-36814-4>
- [10] Mourgias-Alexandris, G., Moralis-Pegios, M., Tsakyridis, A., Simos, S., Dabos, G., Totovic, A., Passalis, N., Kirtas, M., Rutirawut, T., Gardes, F.Y., Tefas, A., Pleros, N.: Noise-resilient and high-speed deep learning with coherent silicon photonics. *Nature Communications* **13**(1), 5572 (2022) <https://doi.org/10.1038/s41467-022-33259-z>
- [11] Bandyopadhyay, S., Sludds, A., Krastanov, S., Hamerly, R., Harris, N., Bunandar, D., Streshinsky, M., Hochberg, M., Englund, D.: Single-chip photonic deep neural network with forward-only training. *Nature Photonics* (2024) <https://doi.org/10.1038/s41566-024-01567-z>
- [12] Skalli, A., Sunada, S., Goldmann, M., Gebiski, M., Reitzenstein, S., Lott, J.A., Czyszanowski, T., Brunner, D.: Model-free front-to-end training of a large high performance laser neural network (2025). <https://arxiv.org/abs/2503.16943>
- [13] Wagner, K., Psaltis, D.: Multilayer optical learning networks. *Appl. Opt.* **26**(23), 5061–5076 (1987) <https://doi.org/10.1364/AO.26.005061>
- [14] Pai, S., Sun, Z., Hughes, T.W., Park, T., Bartlett, B., Williamson, I.A.D., Minkov, M., Milanizadeh, M., Abebe, N., Morichetti, F., Melloni, A., Fan, S., Solgaard, O., Miller, D.A.B.: Experimentally realized in situ backpropagation for deep learning in photonic neural networks. *Science* **380**(6643), 398–404 (2023) <https://doi.org/10.1126/science.ade8450> <https://www.science.org/doi/pdf/10.1126/science.ade8450>
- [15] Ambrogio, S., Narayanan, P., Tsai, H., Shelby, R.M., Boybat, I., Nolfo, C., Sidler, S., Giordano, M., Bodini, M., Farinha, N.C.P., Killeen, B., Cheng, C., Jaoudi, Y., Burr, G.W.: Equivalent-accuracy accelerated neural-network training using analogue memory. *Nature* **558**(7708), 60–67 (2018) <https://doi.org/10.1038/s41586-018-0180-5>
- [16] Wright, L.G., Onodera, T., Stein, M.M., Wang, T., Schachter, D.T., Hu, Z., McMahon, P.L.: Deep physical neural networks trained with backpropagation. *Nature* **601**(7894), 549–555 (2022) <https://doi.org/10.1038/s41586-021-04223-6>
- [17] Scellier, B., Bengio, Y.: Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in Computational Neuroscience* **11** (2017) <https://doi.org/10.3389/fncom.2017.00024>
- [18] Xu, T., Zhang, W., Zhang, J., Luo, Z., Xiao, Q., Wang, B., Luo, M., Xu, X., Shastri, B.J., Prucnal, P.R., Huang, C.: Control-free and efficient integrated photonic neural networks via hardware-aware training and pruning. *Optica* **11**(8), 1039–1049 (2024) <https://doi.org/10.1364/OPTICA.523225>
- [19] Zhao, G., Shu, X., Zhou, R.: High-performance real-world optical computing trained by in situ gradient-based model-free optimization. *IEEE transactions on pattern analysis and machine intelligence* **PP** (2024) <https://doi.org/10.1109/TPAMI.2024.3466853>
- [20] Lin, H., Ou, J., Fan, Z., Yan, X., Hu, W., Cui, B., Xu, J., Li, W., Chen, Z., Yang, B., Liu, K., Mo, L., Li, M., Lu, X., Zhou, G., Gao, X., Liu, J.-M.: In situ training of an in-sensor artificial neural network based on ferroelectric photosensors. *Nature Communications* **16**(1), 421 (2025) <https://doi.org/10.1038/s41467-024-55508-z>
- [21] McCaughan, A.N., Oripov, B.G., Ganesh, N., Nam, S.W., Dienstfrey, A., Buckley, S.M.: Multiplexed gradient descent: Fast online training of modern datasets on hardware neural networks without backpropagation. *APL Machine Learning* **1**(2), 026118 (2023) <https://doi.org/10.1063/5.0157645> https://pubs.aip.org/aip/aml/article-pdf/doi/10.1063/5.0157645/18017061/026118.1_5.0157645.pdf

- [22] Xu, X., Tan, M., Corcoran, B., Wu, J., Boes, A., Nguyen, T.G., Chu, S.T., Little, B.E., Hicks, D.G., Morandotti, R., Mitchell, A., Moss, D.J.: 11 tops photonic convolutional accelerator for optical neural networks. *Nature* **589**(7840), 44–51 (2021) <https://doi.org/10.1038/s41586-020-03063-0>
- [23] Feldmann, J., Youngblood, N., Karpov, M., Gehring, H., Li, X., Stappers, M., Le Gallo, M., Fu, X., Lukashchuk, A., Raja, A.S., Liu, J., Wright, C.D., Sebastian, A., Kippenberg, T.J., Pernice, W.H.P., Bhaskaran, H.: Parallel convolutional processing using an integrated photonic tensor core. *Nature* **589**(7840), 52–58 (2021) <https://doi.org/10.1038/s41586-020-03070-1>
- [24] Huang, C., Fujisawa, S., Lima, T.F., Tait, A.N., Blow, E.C., Tian, Y., Bilodeau, S., Jha, A., Yaman, F., Peng, H.-T., Batshon, H.G., Shastri, B.J., Inada, Y., Wang, T., Prucnal, P.R.: A silicon photonic–electronic neural network for fibre nonlinearity compensation. *Nature Electronics* **4**(11), 837–844 (2021) <https://doi.org/10.1038/s41928-021-00661-2>
- [25] Lederman, J.C., Zhang, W., Lima, T.F., Blow, E.C., Bilodeau, S., Shastri, B.J., Prucnal, P.R.: Real-time photonic blind interference cancellation. *Nature Communications* **14**(1), 8197 (2023) <https://doi.org/10.1038/s41467-023-43982-w>
- [26] Zhang, W., Lederman, J.C., Lima, T., Zhang, J., Bilodeau, S., Hudson, L., Tait, A., Shastri, B.J., Prucnal, P.R.: A system-on-chip microwave photonic processor solves dynamic rf interference in real time with picosecond latency. *Light: Science & Applications* **13**(1), 14 (2024) <https://doi.org/10.1038/s41377-023-01362-5>
- [27] Buckley, S.M., Tait, A.N., McCaughan, A.N., Shastri, B.J.: Photonic online learning: a perspective. *Nanophotonics* **12**(5), 833–845 (2023) <https://doi.org/10.1515/nanoph-2022-0553>
- [28] Guo, Z., Tait, A.N., Marquez, B.A., Filipovich, M., Morison, H., Prucnal, P.R., Chrostowski, L., Shekhar, S., Shastri, B.J.: Multi-level encoding and decoding in a scalable photonic tensor processor with a photonic general matrix multiply (gemm) compiler. *IEEE Journal of Selected Topics in Quantum Electronics* **28**(6: High Density Integr. Multipurpose Photon. Circ.), 1–14 (2022) <https://doi.org/10.1109/JSTQE.2022.3196884>
- [29] Jayatilaka, H., Murray, K., Guillén-Torres, M., Caverley, M., Hu, R., Jaeger, N.A.F., Chrostowski, L., Shekhar, S.: Wavelength tuning and stabilization of microring-based filters using silicon in-resonator photoconductive heaters. *Opt. Express* **23**(19), 25084–25097 (2015) <https://doi.org/10.1364/OE.23.025084>
- [30] Zhang, W., Huang, C., Peng, H.-T., Bilodeau, S., Jha, A., Blow, E., Lima, T.F., Shastri, B.J., Prucnal, P.: Silicon microring synapses enable photonic deep learning beyond 9-bit precision. *Optica* **9**(5), 579–584 (2022) <https://doi.org/10.1364/OPTICA.446100>
- [31] Yang, C., Li, Y., Zhou, S., Guo, Y., Jia, C., Liu, Z., Houk, K.N., Dubi, Y., Guo, X.: Author correction: Real-time monitoring of reaction stereochemistry through single-molecule observations of chirality-induced spin selectivity. *Nature Chemistry* (2024) <https://doi.org/10.1038/s41557-024-01670-2>
- [32] Blaser, E., Pylyshyn, Z.W., Holcombe, A.O.: Tracking an object through feature space. *Nature* **408**(6809), 196–199 (2000) <https://doi.org/10.1038/35041567>
- [33] Lale, S., Renn, P.I., Azizzadenesheli, K., Hassibi, B., Gharib, M., Anandkumar, A.: Falcon: Fourier adaptive learning and control for disturbance rejection under extreme turbulence. *npj Robotics* **2**(1), 6 (2024) <https://doi.org/10.1038/s44182-024-00013-0>
- [34] Kriener, L., Göltz, J., Petrovici, M.A.: The Yin-Yang dataset (2022). <https://arxiv.org/abs/2102.08211>
- [35] Tait, A.N., Lima, T., Nahmias, M.A., Miller, H.B., Peng, H.-T., Shastri, B.J., Prucnal, P.R.: Silicon photonic modulator neuron. *Phys. Rev. Appl.* **11**, 064043 (2019) <https://doi.org/10.1103/PhysRevApplied.11.064043>

- [36] Morison, H., Singh, J., Al Kayed, N., Aadhi, A., Moridsadat, M., Tamura, M., Tait, A.N., Shastri, B.J.: Nonlinear dynamics in neuromorphic photonic networks: Physical simulation in verilog-a. *Physical Review Applied* **21**(3), 034013 (2024)
- [37] Singh, J., Morison, H., Guo, Z., Marquez, B.A., Esmaeeli, O., Prucnal, P.R., Chrostowski, L., Shekhar, S., Shastri, B.J.: Neuromorphic photonic circuit modeling in verilog-a. *APL Photonics* **7**(4) (2022)
- [38] Hagan, D.E., Ye, M., Wang, P., Cartledge, J.C., Knights, A.P.: High-speed performance of a tdfa-band micro-ring resonator modulator and detector. *Optics express* **28**(11), 16845–16856 (2020)
- [39] Khaled, A., Aadhi, A., Huang, C., Tait, A.N., Shastri, B.J.: Fully integrated hybrid multimode-multiwavelength photonic processor with picosecond latency. *Nature Communications* (2025)
- [40] Momeni, A., Rahmani, B., Malléjac, M., Hougne, P., Fleury, R.: Backpropagation-free training of deep physical neural networks. *Science* **382**(6676), 1297–1303 (2023) <https://doi.org/10.1126/science.adi8474> <https://www.science.org/doi/pdf/10.1126/science.adi8474>
- [41] Xue, Z., Zhou, T., Xu, Z., Yu, S., Dai, Q., Fang, L.: Fully forward mode training for optical neural networks. *Nature* **632**(8024), 280–286 (2024) <https://doi.org/10.1038/s41586-024-07687-4>
- [42] Ashtiani, F., Geers, A.J., Aflatouni, F.: An on-chip photonic deep neural network for image classification. *Nature* **606**(7914), 501–506 (2022) <https://doi.org/10.1038/s41586-022-04714-0>
- [43] Zhang, X., Rooney, N., R. Kari, S., Nobile, N.A., Tamura, M., Guo, Z., Lian, C., Ríos Ocampo, C.A., Shastri, B.J., Xiong, F., *et al.*: Non-volatile trimming of photonic micro-rings using integrated and electrically programmable phase-change elements. *Optical Materials Express* **16**(3), 721–732 (2026)
- [44] Hai, M.S., Sakib, M.N., Liboiron-Ladouceur, O.: A 16 GHz silicon-based monolithic balanced photodetector with on-chip capacitors for 25 GBaud front-end receivers. *Opt. Express* **21**(26), 32680–32689 (2013) <https://doi.org/10.1364/OE.21.032680>

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Supplimentary.pdf](#)
- [SupplementaryAA.pdf](#)