

RESEARCH ARTICLE

Open Access



Reconfigurable versatile integrated photonic computing chip

Yufei Wang^{1,2†}, Kun Liao^{1*†}, Kuo Zhang^{3†}, Zhuochen Du^{1†}, Ze Wang^{1†}, Bo Ni^{1,2}, Tianyu Xu¹, Shuai Feng³, Yan Yang^{4*}, Qi-Fan Yang^{1,2,5,6*}, Quan Sun^{2*}, Xiaoyong Hu^{1,2,5,6*}  and Qihuang Gong^{1,2,5,6}

Abstract

With the rapid development of information technology, artificial intelligence and large-scale models have exhibited exceptional performance and widespread applications. Photonic hardware offers a promising solution to meet the growing demands for computational power and energy efficiency. Researchers have aimed to develop an efficient integrated photonic computing chip capable of supporting a wide range of application scenarios in both static and dynamic temporal domains. However, with several mainstream photonic components already well-developed, achieving fundamental breakthroughs at the level of basic computing units remains highly challenging. Here, we report a novel algorithm-hardware co-design strategy that enables in situ reconfigurability across diverse neural network models, all within a unified photonic configuration. We unlock the intrinsic capabilities of a compact cross-waveguide coupled microring component to natively support both static and dynamic temporal tasks. As a proof of concept, we experimentally integrated a turnkey soliton microcomb as the light source on the photonic computing platform, demonstrating the realization of fully connected, convolutional, and recurrent neural network models within a unified structure. The chip achieves area computing efficiency of up to 2.45 TOPS/mm² for 208 tunable components. We evaluate the performance of the proposed chip by implementing image classification tasks on the MNIST and CIFAR-10 datasets, achieving measured test accuracies of 92.93% and 56.57%, respectively. Sentiment analysis on the IMDB dataset achieves a measured test accuracy of 80.81%. Furthermore, speech recognition is implemented by combining three neural networks within a scaled-up architecture. This work addresses the challenges of performing versatile computations on integrated photonic platforms, offering a promising solution for chip-integrated multifunctional photonic information processing.

Keywords Reconfigurable photonic computing, Integrated optical neural networks, Optical frequency comb, Silicon photonics, Artificial intelligence

[†]Yufei Wang, Kun Liao, Kuo Zhang, Zhuochen Du and Ze Wang contributed equally to this work.

*Correspondence:

Kun Liao
kunliao@pku.edu.cn
Yan Yang
yyang10@ime.ac.cn
Qi-Fan Yang
leonardoyoung@pku.edu.cn
Quan Sun
sunquan@ydioe.pku.edu.cn
Xiaoyong Hu
xiaoyonghu@pku.edu.cn

¹ State Key Laboratory for Mesoscopic Physics & Department of Physics, Collaborative Innovation Center of Quantum Matter & Frontiers Science Center for Nano-Optoelectronics, Peking University, Beijing 100871, China

² Key Laboratory for Advanced Optoelectronic Integrated Chips of Jiangsu Province, Peking University Yangtze Delta Institute of Optoelectronics, Nantong, Jiangsu 226010, China

³ School of Science, Minzu University of China, Beijing 100081, China

⁴ Institute of Microelectronics, Chinese Academy of Sciences, Beijing 100029, China

⁵ Collaborative Innovation Center of Extreme Optics, Shanxi University, Taiyuan, Shanxi 030006, China

⁶ Hefei National Laboratory, Hefei 230088, China

1 Introduction

The rapid development of artificial intelligence (AI) and large-scale models has been driven by massive datasets and powerful computational resources [1–3]. To meet the growing demand for computation, photonic chips have emerged, leveraging the low power consumption, high speed, and wide bandwidth of photons as information carriers. Various approaches have been proposed to implement photonic chips using different photonic architectures and computational methods, either as fundamental computational acceleration units or to perform basic artificial intelligence tasks [4–10]. For AI tasks, the most common approach is using a single configuration for each individual function within a network model, such as photonic networks based on cascaded Mach–Zehnder interferometers (MZI) architectures, which are widely used for matrix–vector multiplications or convolutional layers in neural networks [11–17]; on-chip optical diffractive architectures that provide dense connections via diffraction are served as the hidden layers of diffractive neural networks [18–22]; and wavelength-division multiplexed (WDM) microring resonator (MRR) arrays, which are employed for efficient convolution operations [23, 24]. Beyond using a single configuration inherently tied to a specific single network model, hybrid configurations have been proposed to collaboratively execute the overall one model with multiple specialized blocks [25–31]. In 2022, Zhu et al. hybridized MZIs and diffraction cells on-chip for implementing parallel Fourier transforms, convolution operations and application-specific optical computing [29]. In 2024, Xu et al. integrated diffraction units and MZIs to passively perceive the high-dimensional inputs as compact representations and performed efficient task-specific feature embeddings with fully reconfigurable MZIs, combining the advantages of optical diffraction and interference [25].

As the works mentioned above, single configuration model primarily focuses on feed forward neural networks, which are designed for static tasks such as image classification and pattern recognition. Hybrid configurations are proposed to improve the network efficiency and its capacity to handle large-scale tasks, which may impose higher demands on micro-nano fabrication processes, making practical implementation more challenging (see Supplementary Note 1 and Supplementary Fig. 1 for more details). However, in practice, more flexible strategy and computational schemes can be employed to design a unified configuration that can be re-programmed in situ to enable seamless switching between different neural network models, including static and dynamic temporal ones, without the need for hybrid configurations. In 2024, Du et al. demonstrated both convolutional neural network (CNN) and recurrent neural network (RNN) on

the unified structure using a cross-waveguide coupled MRR unit [31]. However, this work achieved on-chip integration of only 8 MRRs, corresponding to 4 neurons, without discussing the scalability of the network. A versatile integrated photonic network architecture capable of supporting diverse network models is still being pursued, especially configurations that can handle dynamic temporal tasks, including RNN, long-short-term memory network (LSTM) [32], gated recurrent unit (GRU) [33], etc., which are particularly relevant for modern large-scale models and multimodal learning tasks. Although energy-efficient reconfigurable RNN accelerators have already been demonstrated on non-photonic platforms such as FPGAs and DSPs [34, 35], optical implementations remain comparatively modest. While some progress has been made in fiber-optic, optoacoustic and other optical systems [36–38], integrated photonic realizations have thus far been limited to theoretical proposals or proof-of-principle unit structures [39–41], leaving scalable, large-scale RNN models largely unexplored. Additionally, reservoir-computing (RC) networks, a subset of RNNs, have been investigated optically [42–44], yet their fixed and random reservoir weights constrain performance on complex tasks. Given the lack of a unified structure capable of addressing both static and dynamic temporal tasks, it is still a great challenge to realize reconfigurable and versatile integrated photonic computing chips.

Here, we report a scalable versatile integrated photonic chip that facilitates the implementation of diverse neural networks within a unified structure capable of handling both static and dynamic temporal tasks with high efficiency (Fig. 1a–f). Fully reconfigurable MRR and MZI arrays are utilized to handle diverse processing demands at different stages of large-scale tasks, including feature extraction, temporal reasoning, and output decision-making. Each MRR is based on a cross-waveguide coupling design, allowing each MRR unit to switch between handling static and temporal tasks by inputting information through one or both ports of the cross-waveguide, without requiring additional structures. As a proof of concept, we experimentally integrated a turnkey soliton microcomb with a free spectral range (FSR) of 100 GHz as the light source, fully leveraging the wavelength freedom to demonstrate the realization of fully connected neural network (FCNN), CNN, and photonic gated recurrent neural network (PGRNN) models with a tunable area efficiency of up to 2.45 TOPS/mm² at a typical frequency of 10 GHz (Supplementary Note 10). Table. S4 and Fig. S28 in the Supplementary Note 12 summarize the comparison between this work and several state-of-the-art photonic computing schemes. Three neural network tasks were demonstrated to validate the proposed

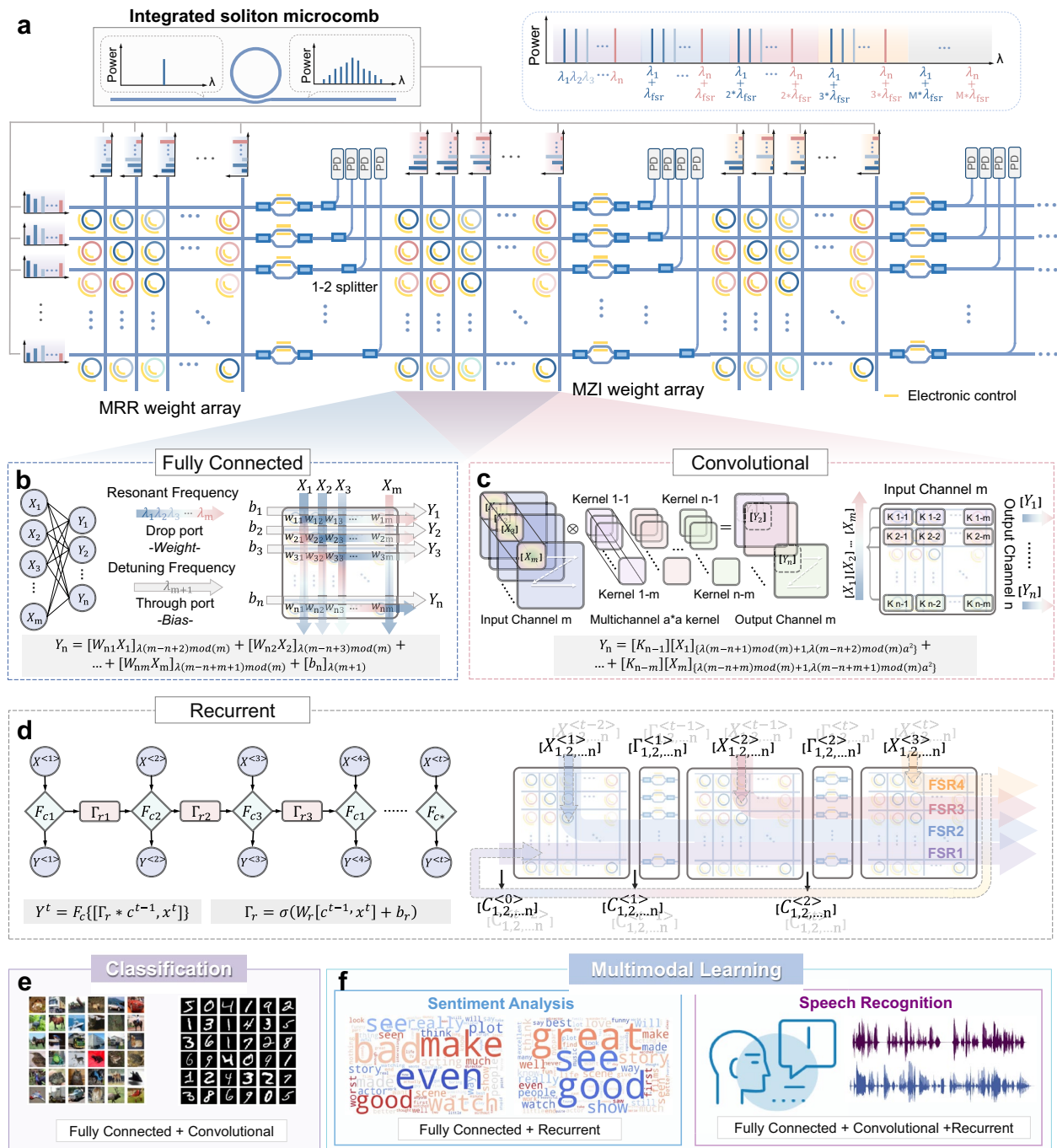


Fig. 1 Proposal of the reconfigurable versatile integrated photonic computing chip. **a** The architecture of a reconfigurable multifunctional integrated photonic chip. The main computational units of the chip consist of MRR arrays and MZI arrays, with a fully-integrated single soliton optical frequency comb used as the light source. **b** The mechanism for completing FCNN using a single-layer network. **c** The mechanism for completing CNN using a single-layer network. **d** The mechanism for completing PGRNN using a multi-layer network. **e** A demonstration on the chip using the combination of FCNN and CNN to complete an image classification task. **f** A demonstration on the chip using the combination of FCNN, CNN, and PGRNN (the combination of two or three) to complete multimodal tasks such as sentiment analysis and speech recognition

chip's adaptability and performance, particularly for multimodal tasks, which are in high demand but rarely

addressed by photonic neural networks (Fig. 1e, f). First, for image classification, a combination of CNN and

FCNN was used to construct an Inception model, achieving test accuracies of 92.93% and 56.57% on the MNIST and CIFAR-10 datasets, respectively. Second, PGRNN and FCNN were employed for sentiment analysis, achieving test accuracy of 80.81% on the IMDB dataset. Third, a combination of CNN, PGRNN, and FCNN with a scaled-up architecture was used for speech recognition. These tasks highlight the proposed photonic neural network's flexibility and scalability in accommodating diverse network models and its capability to solve multimodal tasks involving images, text, and audio, offering a promising path toward the development of next-generation photonic computing platforms (the scalability potential of our photonic computing chip is demonstrated in Supplementary Note 11).

2 Results and discussions

2.1 The versatile integrated photonic platform

We propose a scalable integrated photonic platform based on silicon-on-insulator (SOI) platform, capable of implementing FCNN, CNN, and PGRNN within a unified structure. To achieve a scalable and complete computational architecture, we employ fully reconfigurable MRR arrays as the fundamental building blocks of the network, enabling multi-layer cascaded processing, as shown in Fig. 1a. To further enable interlayer modulation, MZI arrays are introduced between adjacent layers. To ensure low crosstalk and high-precision computation, on-ring micro-heaters are integrated on the MRRs for fine-tuning of their resonance (see Supplementary Note 10 for the impact of thermal tuning speed and stability on computation), and a single-soliton optical frequency comb based on silicon nitride (Si_3N_4) microcavity is employed as the optical source. The comb teeth are modulated to enable precise amplitude modulation at specific wavelengths, allowing input encoding via the wavelength dimension. By altering the encoding scheme, we can demonstrate FCNN and CNN using a single-layer network, while the more challenging PGRNN can be demonstrated using the complete three-layer network.

We first introduce the operating principle of the single-layer network for implementing the FCNN (Fig. 1b). The core components of an FCNN—weights, inputs, and biases—are efficiently realized within a unified structure. Instead of separately computing weight multiplications and subsequently introducing biases through additional components or modulators, we leverage the two input ports of the MRR cross-waveguide, along with their corresponding drop ports, through ports, and the ingenious combination of different wavelengths of the optical frequency comb. This enables the direct multiplication of weights and inputs while seamlessly incorporating additive biases within the same MRR. The input information

vector is encoded according to the corresponding wavelengths and injected through the upper input ports of the MRR array. Each MRR in the array can independently tune its corresponding weight via electrical control, allowing the weighted summation to be directly obtained from the drop port outputs. To introduce biases, the left ports of the MRR array serve as additional inputs, carrying wavelengths that are detuned from all MRRs. Therefore, the through port (which corresponds to the drop port on the right side from the perspective of the input port) allows the bias term to be directly added to the weight term without further modulation. This design enables the photodetector to directly capture the combined result of weights and biases, ensuring efficient and complete computation of FCNN.

Having discussed the working principle of FCNN, we proceed to examine CNN (Fig. 1c). As one of the most widely used network models, CNN accounts for the most computationally intensive operations when processing large models. To address this challenge, photonic convolution kernels can typically be employed to accelerate the convolution operations within the network. The MRR-based convolution kernel we proposed can flexibly achieve convolution kernels of different sizes and multi-channel to multi-channel convolutions through different wavelength encodings and weight configurations of the MRRs. For instance, the multi-channel to multi-channel 1×1 convolution kernel plays a significant role in dimensionality reduction and cross-channel fusion, making it a fundamental component in classic network models such as Inception [45] and ResNet [46]. An MRR array of size $m \times n$ can efficiently realize a 1×1 convolutional network from m input channels to n output channels. During the convolution process, the weights of the MRRs remain unchanged, ensuring that the computational speed is not constrained by the thermal tuning response time. The principles for other convolution kernel sizes and different channel transformations are detailed in the Supplementary Note 3 and Supplementary Figs. 2–5.

Both the FCNN and CNN introduced above can be implemented using a single-layer MRR array. While the proposed PGRNN is based on a multi-layer MRR array with MZI interconnects between layers (Fig. 1d and Supplementary Fig. 8). A key characteristic of RNNs is their ability to capture temporal dependencies in sequential data, enabled by their distinctive network structure. Here, we employ a cross-waveguide MRR as a single neuron in the PGRNN (Supplementary Fig. 6). The two input ports of the cross-waveguide are designed to transmit information from the current time step and the preceding time step, respectively. The final output of the RNN is typically obtained by computing hidden states layer by layer, with the output representing either the entire time sequence

or the hidden state of the last time step, depending on the specific task requirements. To better address long-range dependencies in time sequences, PGRNN incorporates certain enhancement mechanisms such as reset gate, which regulates the proportion of information carried forward from the hidden state of the previous time step. In our design, MZI array is introduced between two layers of neurons as the reset gate to regulate the output proportion of each layer. A single MRR neuron is utilized to simultaneously modulate two sets of input signals by leveraging the different free spectral ranges (FSRs) of the MRR, input signals at wavelengths λ and $\lambda + \text{FSR}_{\text{mrr}}$ are injected from the two ports of the cross-waveguide respectively. The output waveguide on the right corresponds to both the drop port of the upper input port and the through port of the left input port. By detecting the signal from the right output waveguide, the superimposed result of both outputs can be obtained (Supplementary Note 4 and Supplementary Fig. 7). This design not only enables a single MRR to simultaneously modulate multiple optical signals from different input ports, but also ensures that the two signals do not interfere with each other, preventing signal crosstalk caused by interference.

Figure 2a–j illustrate experimental characteristics of our fully-integrated microcomb chip and the microcomb-driven versatile computing chip, respectively, along with the characterization of their key performance metrics. Figure 2a presents an optical image of the on-chip DFB laser directly coupled to the Si_3N_4 microresonator chip, which was fabricated by Qaleido Photonics [47], alongside a magnified optical image of the microresonator, which has dimensions of $5\text{ mm} \times 5\text{ mm}$. To enhance practicality, we fully packaged the turnkey microcomb module, integrating the driving current source and temperature control unit, resulting in a compact form factor comparable to that of a computer mouse [48] (Supplementary Note 6 and Supplementary Fig. 12). The fully-integrated microcomb chip is capable of generating a single-soliton mode-locked microcomb with a 100 GHz repetition rate, covering a spectral range exceeding 100 nm. After calibration, it delivers a stable and consistent microcomb source with reliable power output for computational use for up to 1 h (Fig. 2b, c).

The optical image of the packaged photonic chip and the details of the sample are presented in Fig. 2d, while Fig. 2e shows an optical micrograph of the MRR with the heater. Figure 2f displays the optical spectra of a single MRR device, obtained by splicing together measurements under different tuning voltages using a super-continuum light source as the input. As the voltage increases, the resonance wavelength of the MRR exhibits a redshift. When the voltage exceeds 7.5 V, the resonance

shift covers an entire FSR_{mrr} , ensuring full tuning of all wavelengths within one FSR_{mrr} . The tuning precision of the thermal control voltage is 0.001 V. By determining the voltage range corresponding to the minimum and maximum output at a specific wavelength within the MRR, precise weight control is achieved through fine voltage adjustments. To evaluate the weight precision of a single MRR and its susceptibility to thermal crosstalk, we applied weight values from 0 to 1 in increments of 0.1 to a single MRR while randomly applying voltages to other heating electrodes. The corresponding weight values of the single MRR were measured under 50 random voltage disturbances. As shown in Fig. 2g, the weight precision meets the computational requirements. Following the characterization of a single MRR, we further evaluated a single MZI. As the applied voltage increases, the MZI output follows a cosine variation curve (Fig. 2h). By recording the voltage range corresponding to the maximum and minimum output values, we achieved weight modulation from 0 to 1 within the MZI. For the PGRNN, which requires a multi-layer network, we characterized the MRR's weight tuning capabilities across multiple FSR_{mrr} . Figure 2i illustrates the electrical tuning bandwidth of the MRR, indicating that within the applied voltage range, the resonance wavelength can be shifted by an entire FSR_{mrr} . Moreover, the frequency shifts across different five FSR_{mrr} remain consistent with voltage variations, ensuring uniform and controllable wavelength modulation when utilizing input signal carriers located in different FSR_{mrr} . When the same MRR modulates input signals at wavelengths λ_1 and $\lambda_1 + \text{FSR}_{\text{mrr}}$, the through and drop outputs inherently exhibit opposite resonance trends in the spectral domain. Under thermo-optic voltage tuning, this spectral relationship naturally converts into opposite modulation trends in the voltage domain, causing the through output to decrease while the drop output increases. By detecting the same output port when it functions as the through port for λ_1 and as the drop port for $\lambda_1 + \text{FSR}_{\text{mrr}}$ (Fig. 2j), it is observed that within the same voltage tuning range, the weight modulation trends for these two wavelengths exhibit complementary Lorentzian-like variations, transitioning from 1 to 0 and from 0 to 1 respectively. This weight mapping relationship is utilized in the PGRNN computational process to achieve precise weight control.

2.2 Photonic neural networks for image classification

We first demonstrate the capability of our scalable integrated photonic chip by performing image classification tasks, a foundational static task that serves as a benchmark application for CNN and FCNN architectures. Our approach to image processing adopts the core concept of Inception, which extracts multi-scale image features

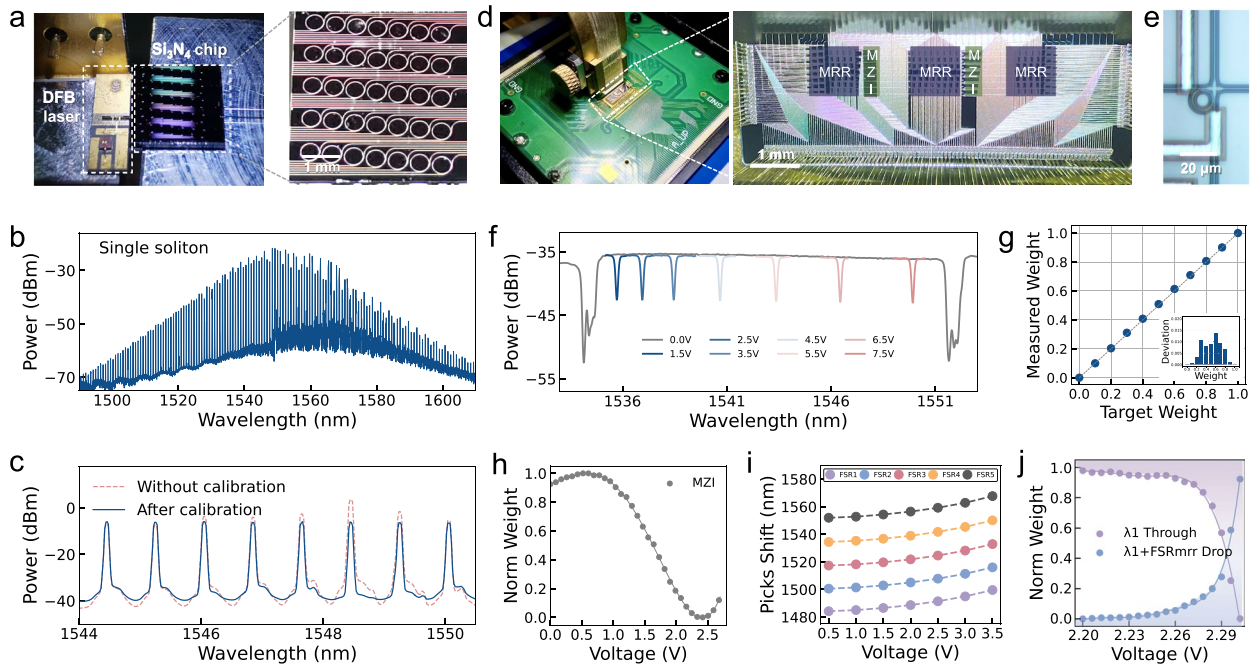


Fig. 2 Key characterization of the fully-integrated microcomb chip and microcomb-driven versatile computing chip. **a** The left picture is the optical image of the DFB laser coupled with Si_3N_4 microring resonator, while the right picture is the magnified of the Si_3N_4 microresonator. **b** Spectrum of the soliton microcomb. **c** Magnified view of eight comb teeth with and without calibration. **d** The left picture is the optical image of the packaged computing chip, while the right picture is a magnified detail of the chip. **e** A single cross-waveguide based MRR with heater details under an optical microscope. **f** The variation of the resonance wavelength of a single MRR under voltage control from 0.0 to 7.5 V. **g** The weight accuracy of a single MRR when 50 different random weights are applied to other MRRs. The inset shows the variance of the weights. **h** Normalized weight curve of a single MZI under different voltage control (the voltage axis is paired with a power scale ranging from 0 to 50 mW). **i** The resonance wavelength of a single MRR controlled by voltage across different FSRs_{mrr} (the voltage axis is paired with a power scale ranging from 0 to 40 mW), ensuring consistency and controllability when processing information using wavelengths from different FSRs_{mrr} simultaneously. **j** The purple curve represents the through port output curve of the MRR (tuning wavelength at $\lambda_1 = 1535.32$ nm). The normalized weight shows a Lorentzian-like decrease from 1 to 0. The blue curve represents the drop output curve of the MRR (tuning wavelength at $\lambda_1 + \text{FSR}_{\text{mrr}} = 1552.78$ nm). The normalized weight shows a Lorentzian-like increase from 0 to 1

through parallel convolution operations rather than relying on fixed-size convolutional kernels. For the CIFAR-10 dataset, which comprises of real-world RGB three-channel images, we employ convolution kernels of varying sizes to extract image features, which are then concatenated, flattened, and processed by fully connected layers to produce classification outputs across 10 categories. Figure 3a illustrates the working principle of the 1×1 convolutional kernel, where the MRR weights need to support real-value encoding (Supplementary Note 2). In the experiment, to ensure accurate input encoding and precise weight deployment on the MRRs and MZIs, preliminary calibration of both the input and the neurons is required (Supplementary Note 7 and Supplementary Figs. 13–19). To streamline this process, we developed an automated calibration pipeline comprising voltage scanning, extreme value detection, function fitting, and weight lookup table generation. This pipeline enables seamless and efficient weight calibration without excessive manual intervention.

The theoretical output, experimental output, and their deviation for the convolution operation on the MNIST dataset (digit “9”) [49] and CIFAR-10 dataset (ship) [50] are shown in Fig. 3b, c, with the RMSE calculated to be 0.0444 and 0.0261, respectively. The results indicate minimal deviation and low RMSE values, confirming that our calibration method effectively ensures consistency between theoretical calculations and experimental outcomes, providing a robust foundation for further application demonstrations. Our network achieved a training accuracy of 98.95% on the MNIST dataset. Figure 3d presents the training accuracy and loss curves over five different initialization runs, while Fig. 3e displays the confusion matrix obtained experimentally on the MNIST dataset, achieving an accuracy of 92.93%. For the more challenging CIFAR-10 dataset, the training accuracy and loss curves across five different initialization runs are shown in Fig. 3f, with the highest training accuracy reaching 69.52%, and the confusion matrix of the experimental

results is presented in Fig. 3g, achieving a test accuracy of 56.57%. It is worth noting that, to match the experimental scale, training was conducted with a total of 81,920 trainable parameters, which is experimentally a relatively small training scale. Given the intrinsic difficulty of the CIFAR-10 dataset and the limited training data available in our experiment, the observed performance is already reasonably competitive. A more detailed discussion of experimental factors affecting accuracy, including thermal drift and modulation rate, is provided in the Supplementary Note 10 and Supplementary Fig. 27. Our network demonstrated highly efficient training performance with small training scale, experimentally validating both the advanced capability of our model and the potential of our chip for optical computing in computer vision tasks.

2.3 Photonic neural networks for sentiment analysis

Having validated the chip's capability in performing fundamental neural network computation, we next present sentiment analysis—a dynamic, sequential task that further demonstrates the network's ability to handle RNN-based computations. Sentiment analysis is a crucial task in natural language processing (NLP) that requires the network to extract and synthesize information from text paragraphs. Unlike traditional image processing and other static tasks, sentiment analysis evaluates the network's proficiency in processing sequential data, thereby imposing higher demands on its specialized architecture and model complexity. For our experiment, we utilize the classic IMDB dataset [51], which comprises movie reviews from the Internet Movie Database, each labeled with either a positive or negative sentiment. The dataset presents additional challenges due to its varying text lengths, diverse linguistic expressions, and even noise such as spelling errors, making data selection and model training more complex. The training architecture of our model is depicted in Fig. 4a. Each word in a sentence is assigned a numerical index, each word index is subsequently mapped to an embedding layer, encoding it into a feature vector with 8 features (Supplementary Fig. 20). These feature vectors are sequentially input from the top of each 8×8 MRR modulation array, while a pre-trained initial vector is fed from the left side of the MRR array. The right output array of each MRR layer contains both the current word's information and the contextual information accumulated from previous words. An MZI array is employed to regulate the proportion of prior contextual information retained for subsequent steps. This recursive process continues until all words in the sentence have been processed. Each hidden layer outputs 8 features per timestep, representing the transformation of features at each word step. Since sentiment analysis

requires summarizing and analyzing the entire sentence or paragraph, an average pooling operation is applied across all timesteps, yielding 8 averaged features. These features are then processed by an FCNN, reducing them to a single-channel output, with a classification threshold set at 0.5 (Fig. 4b). More training details are provided in Supplementary Note 8.

Considering the training time constraints and the scale of the experiment, we selected samples with less than 60 words from 25,000 training datasets (including 12,500 positive and 12,500 negative reviews) to obtain 1098 samples. These samples were processed through a PGRNN with a hidden layer dimension of 8, followed by an FCNN with 8 inputs and 1 output, totaling 80,464 trainable parameters. The final training accuracy reached 99.23%, with the training accuracy and loss curves shown in Fig. 4c, d. Given the limited number of training samples and model parameters, this result is highly promising. Representative sample cases are provided in Supplementary Table 2. The theoretical and experimental confusion matrices are presented in Fig. 4e, f, yielding corresponding test accuracies of 82.35% and 80.81%, respectively. By analyzing the final decision outputs of 30 test samples (Fig. 4g, h), we observed that, due to the binary nature of the classification task, cases exist where theoretical predictions are incorrect, yet experimental errors inadvertently lead to correct classifications. Furthermore, since binary classification exhibits a degree of tolerance to experimental errors, samples with more extreme decision outputs can still be correctly classified even in the presence of larger deviations. This task highlights the strong potential of our network architecture for handling dynamic temporal tasks, demonstrating its robustness in processing sequential data.

2.4 Photonic neural networks for speech recognition

The image classification and sentiment analysis tasks have respectively demonstrated and experimentally validated the capabilities of our chip in handling FCNN, CNN, and PGRNN (Fig. 5a). To further demonstrate the combined utilization of these network models on our designed chip, we tackle one of the most challenging yet widely used functions in large models—speech recognition (the physical architecture and the execution flow are detailed in Supplementary Note 9 and Supplementary Fig. 21–22). In this task, raw waveform information (Fig. 5b and Supplementary Fig. 25a) is first transformed into spectrograms using Short-Time Fourier Transform (STFT) in order to provide the neural network with rich feature representations while reducing computational costs (Supplementary Fig. 23). The spectrograms are initially processed by a CNN to transform their dimensionality, facilitating feature extraction and reducing data

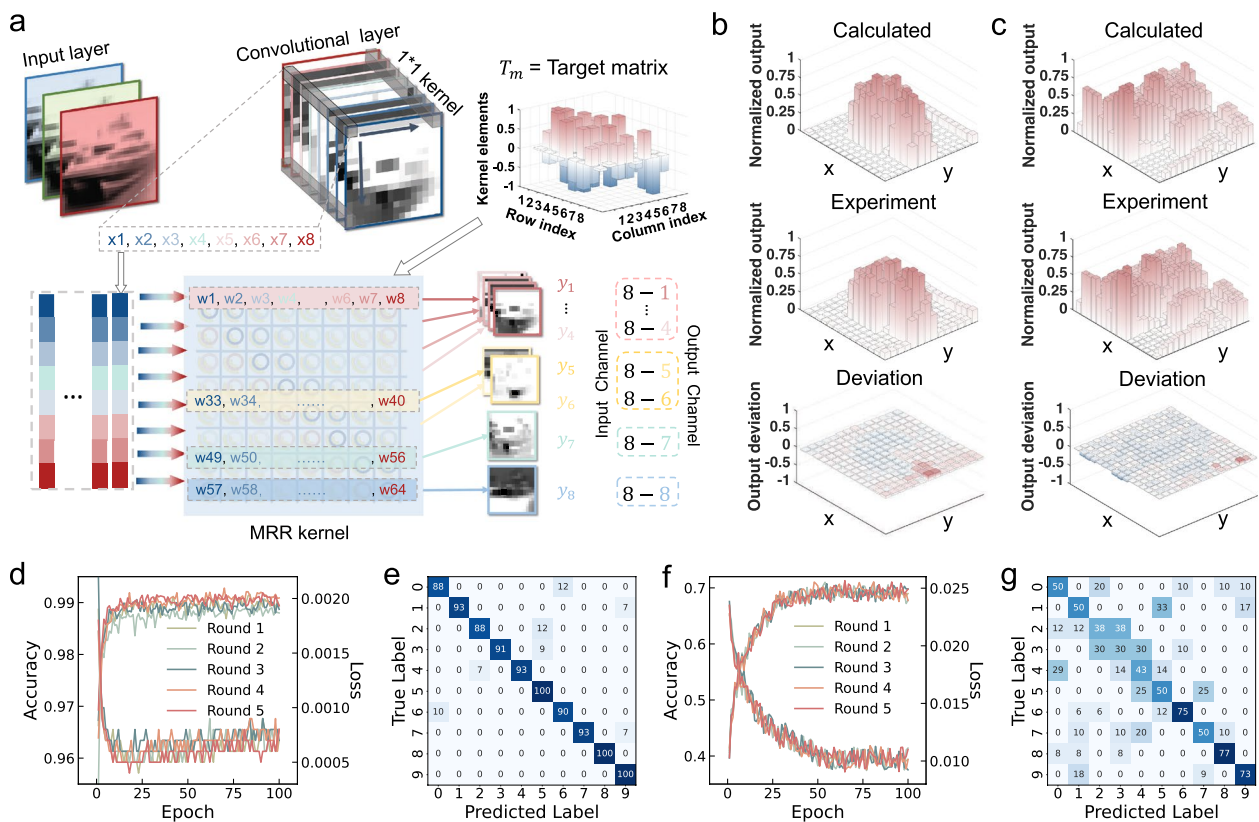


Fig. 3 Image classification tasks. **a** The input encoding and convolution principle when using a 1×1 convolution kernel in a single-layer network. **b** Theoretical output, experimental output, and the deviation between the theoretical and experimental results after a single convolution operation on the sample “9” (28×28 pixels) from the MNIST dataset. **c** Theoretical output, experimental output, and the deviation between the theoretical and experimental results after a single convolution operation on the “Ship” sample (32×32 pixels) from the CIFAR-10 dataset. **d** Training loss and accuracy on the MNIST dataset. **e** Confusion matrix from the MNIST dataset experiment. **f** Training loss and accuracy on the CIFAR-10 dataset. **g** Confusion matrix from the CIFAR-10 dataset experiment

dimensionality. Subsequently, the processed features are fed into a PGRNN layer with a hidden dimension of 416 for audio feature analysis and inference. Finally, an FCNN reduces the 416-dimensional output to a 32-dimensional representation, where each dimension corresponds to the probability of the 26 English letters (“a”–“z”) and commonly used punctuation marks included in the training character set at a given time step (Fig. 5d and Supplementary Fig. 25c). The character with the highest probability is selected as the final predicted output (Fig. 5f and Supplementary Fig. 25e).

A critical challenge in this task is distinguishing between prolonged sounds of the same letter and naturally occurring consecutive letters within words. This challenge is addressed using the Connectionist Temporal Classification (CTC) method. Figure 5e and Supplementary Fig. 25d zoom in on 30 timesteps to compare the ground truth with the predicted output at each step. To evaluate the model’s training performance, we quantify the frequency of these errors using the word error rate

(WER), defined as the ratio of the number of substitutions, deletions, and insertion errors to the total number of words. The model is trained and evaluated on the LJ-Speech dataset [52], which comprises 13,100 audio-text pairs sampled at 16 kHz. Within our scaled-up architecture, the model achieves a final training WER of 0.41 and a test WER of 0.55 (Supplementary Fig. 24). Figure 5c and Supplementary Fig. 25b compares the ground truth and predicted texts for two test sentences (more samples are detailed in Supplementary Fig. 26). The WER for the first sample shown in Fig. 5b is 0.688, while the second sample shown in Supplementary Fig. 25a exhibits a higher WER of 0.167. Despite the significant difference in WER between the two samples, the predicted words from the second sample still allow for meaningful interpretation of the sentence, whereas the raw audio alone provides no direct semantic information without model inference. This demonstrates that even for samples with relatively low inference accuracy from our network’s results, meaningful and interpretable outcomes can still be obtained.

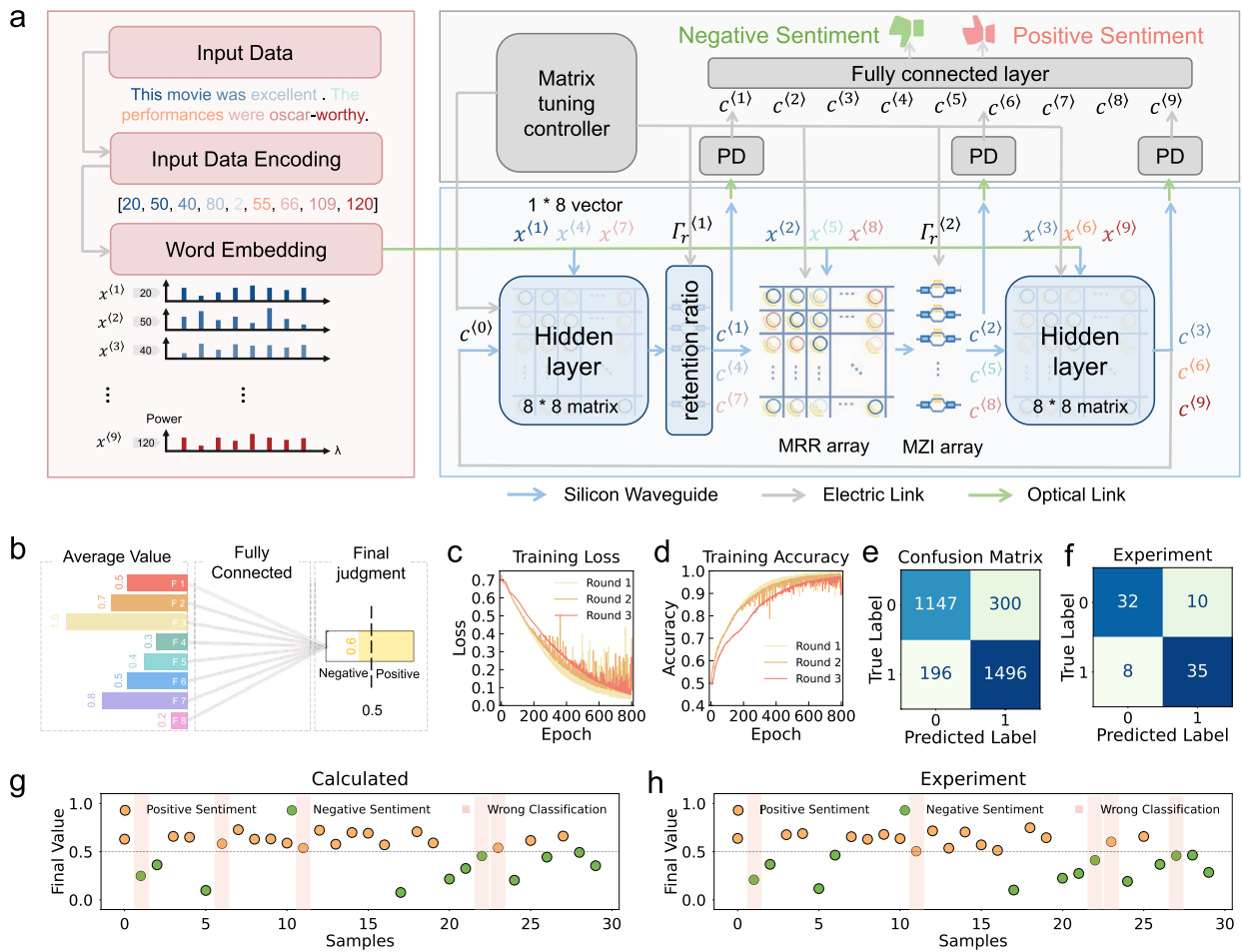


Fig. 4 Sentiment analysis tasks. **a** The input encoding process and neural network inference flow for the sentiment analysis task. **b** The 8 feature vector outputs after averaging along the word step dimension following PGRNN, which then pass through a fully connected layer to generate the final decision output. **c** Training loss on the IMDB dataset. **d** Training accuracy on the IMDB dataset. **e, f** Confusion matrix for the IMDB dataset from the theoretical computation (**e**) and the experiment (**f**). **g, h** The final decision output distribution for 30 selected samples from the IMDB dataset: theoretical computation (**g**) and experiment (**h**)

This task, which integrates FCNN, CNN, and PGRNN, not only evaluates our chip's capacity to handle dynamic temporal data but also showcases its potential for performing highly complex computations in real-world applications.

3 Conclusions

In this work, we report a strategy for realizing a reconfigurable and versatile integrated photonic computing chip that facilitates the implementation of FCNN, CNN and PGRNN within a unified structure capable of handling both static and dynamic temporal tasks with high efficiency. Our platform leverages a fully-integrated soliton microcomb as the light source and integrates MRR and MZI to enable flexible and reconfigurable photonic computing with an area efficiency of up to 2.45 TOPS/mm² for 208 tunable components. To validate the adaptability

and performance of our proposed chip, we conducted three neural network tasks consist of image classification, sentiment analysis, and speech recognition, each demonstrating exceptional inference capabilities. By overcoming the challenges of achieving versatile computation on integrated photonic platforms, our work lays a strong foundation for advancing integrated photonic systems, offering a promising path toward the development of next-generation photonic computing platforms capable of tackling both static and dynamic large-scale model tasks, and potentially transforming the landscape of photonic-based artificial intelligence applications.

4 Methods

4.1 Device fabrication

The devices were fabricated using the CMOS pilot process line, with 248 nm- λ deep ultraviolet (DUV)

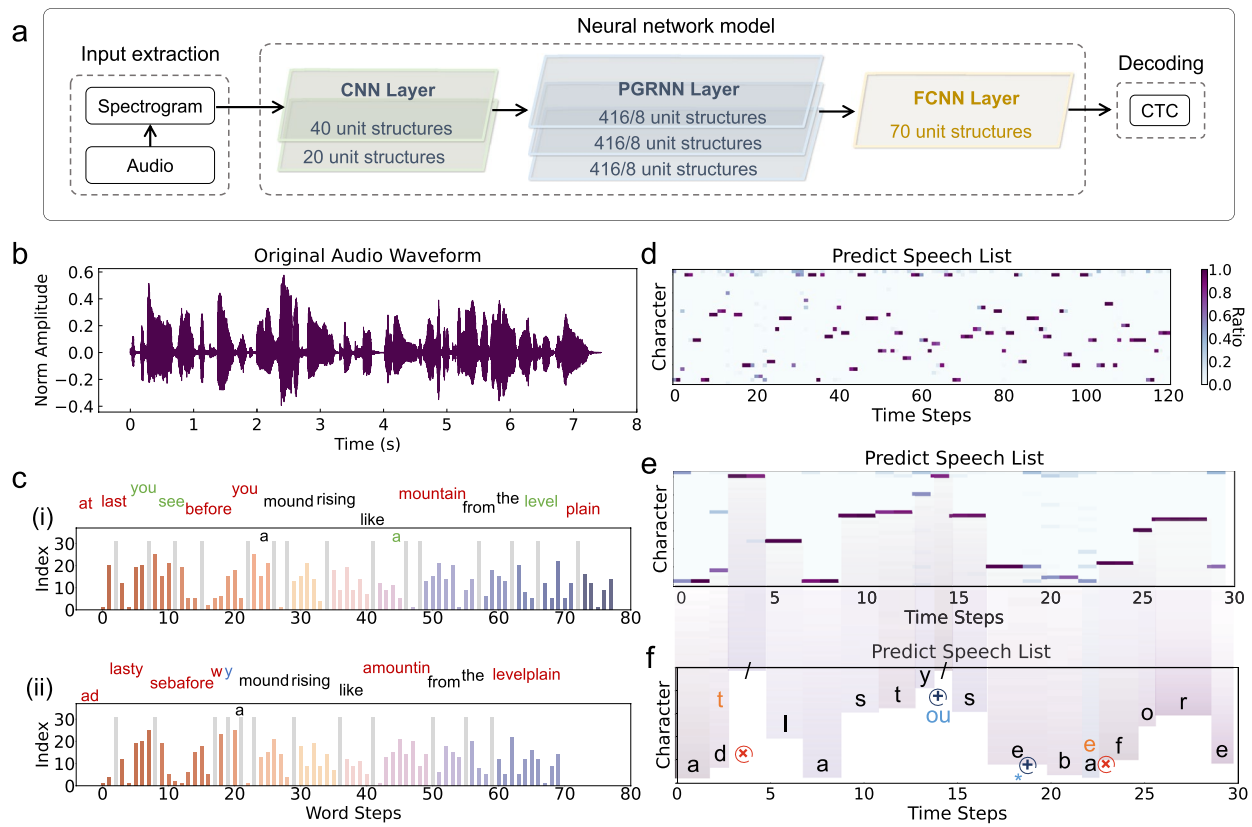


Fig. 5 Speech recognition tasks. **a** The input encoding process and neural network inference flow for the speech recognition task. **b** Raw audio waveform from the LJ-Speech dataset. **c** The corresponding true text (i) and predicted text (ii). The errors in (ii) include 6 substitutions [replacing "at" with "ad", replacing "last" with "lasty", etc., marked in red in (i) and (ii)], 4 deletions [deleting "you" and "see", etc., marked in green in (i)], and 1 insertion [inserting "y" after "w", marked in blue in (ii)]. **d** The character probabilities at each time step, with the y-axis representing the 26 English letters and common punctuation marks, totaling 32 characters. **e** Zoomed-in view of the character probabilities for the first 30-time steps and their corresponding final predicted output characters (**f**)

photolithography employed for mass production. The fabrication took place on a 200 mm-diameter silicon-on-insulator (SOI) wafer, featuring a 220 nm-thick top silicon layer and a 3 μm -thick buried oxide (BOX) layer. The wafer was coated with positive photoresist and patterned using DUV lithography, followed by a double inductively coupled plasma (ICP) etching process for the top silicon. The device's waveguides were formed by a 220 nm-depth full etch, and the grating couplers were created with a 70 nm-depth shallow etch. A high-temperature annealing process, specifically a hydrogen (H_2) annealing step, was then employed to reduce the optical loss of the waveguides, decreasing the typical propagation loss of the silicon waveguides from approximately 3 dB/cm before annealing to around 1.5–2 dB/cm after treatment. Following this, a 1 μm -thick cladding of silicon dioxide was deposited using plasma-enhanced chemical vapor deposition (PECVD), preparing the surface for the

heater and metallization processes. A 55 nm-thick TiN layer was deposited by PECVD and dry etched to form the heater for the resistive thermal-optic phase shifters. Next, an 800 nm-thick AlCu layer was deposited and wet etched to define the metallization pattern. Finally, another 1 μm -thick silicon dioxide cladding was deposited by PECVD, followed by pad opening etching to form the wire bonding pads.

4.2 Integrated turnkey soliton microcomb generation

We demonstrated an integrated turnkey soliton microcomb by directly coupling a commercial Distributed Feedback (DFB) laser to a Si_3N_4 chip [53]. Since no isolator was used in the experimental setup, the backscattered light inside the microresonator was re-injected into the laser, thereby altering its frequency and achieving relative locking between the laser frequency and the microresonator mode. This phenomenon, known as self-injection

locking, can reduce the laser linewidth and facilitate the generation of mode-locked microcomb when the feedback phase is appropriately adjusted. In our experiment, the Si_3N_4 microresonator had an FSR_{comb} of 100 GHz and a loaded Q factor of 5 M. The optical power injected into the chip waveguide from the DFB was approximately 30 mW, and the laser frequency was tuned to match the microresonator resonance. The feedback phase was controlled via a piezoelectric displacement stage, and after optimizing the feedback phase, the formation of the microcomb was observed. Notably, a low-noise step appeared in the comb power spectrum, corresponding to the formation of a single mode-locked pulse inside the microresonator. This mode-locked state remained stable within a laser current range of 0.3 mA, enabling reliable generation of the mode-locked microcomb. Ultimately, we observed a single soliton mode-locked microcomb with a 100 GHz repetition rate and a spectral range exceeding 100 nm, with over 30 comb teeth having a power greater than -30 dBm in the direct output. A detailed comparative analysis of other candidates for multi-wavelength light sources and microcomb is provided in the Supplementary Table 1.

4.3 Experimental setup

In the experiment, we used a supercontinuum light source (YSL PHOTONICS SC-5) and a spectrometer (YOKOGAWA AQ6370D) to characterize the performance of the MRR, including the change in resonant frequency under thermal tuning. During the initial calibration process, the supercontinuum source was also used to conveniently observe the resonant frequency of the MRR. For the experimental computational tasks, we used our home-made integrated turnkey soliton microcomb as the light source. The optical frequency comb's power was amplified using an erbium-doped fiber amplifier (Amonics AEDFA-C-EX1-DWDM-22-B-FA and Thorlabs EDFA 100P) to compensate for power losses from subsequent input modulation equipment. The amplified optical frequency comb was then independently power-controlled for different frequencies using a dense wavelength division multiplexer (FS FMU-D402160M) and a variable optical attenuator (Si Chuan Zi Guan Photonics Technology VOA module). The VOA module adjusted its attenuation value via the RS232 communication protocol to encode the input data. Based on computational requirements, a 50:50 split was made through a single-input multiple-output optical fiber splitter. After encoding, the signal was input through a 56-channel optical fiber array with a $127\text{ }\mu\text{m}$ pitch (32 channels for input and 24 channels for output). The output ports were monitored using a single-channel optical

power meter (Thorlabs PM100USB and S154C) and a multi-channel optical power meter (Si Chuan Zi Guan Photonics Technology OPM-24 module). More details of experimental setup can be found in Supplementary Note 5 and Supplementary Figs. 10–11. The computational efficiency can be measured by the computational speed per unit area. The MRR array and MZI array on our chip form a complete single-layer network with an area of $1.12\text{ mm} \times 0.96\text{ mm}$, while the computing speed is $264f_{\text{clk}} \times 10^{-12}$ (Supplementary Note 10 and Supplementary Table 3 presents the detailed calculation of TOPS and power consumption). Currently, our modulation speed is mainly limited by the VOA, since there have already been demonstrations of lithium niobate modulators achieving modulation rates of tens of GHz [54, 55], we can significantly enhance our system by replacing the low-speed VOA with a high-speed lithium niobate electro-optic modulator. This upgrade would allow the area efficiency to reach $2.45\text{ TOPS}/\text{mm}^2$ at a typical frequency of 10 GHz. Additionally, high-speed InGaAs photon detectors operating at over 220 GHz and analog-to-digital converters can be used. The efficient optical configuration, along with high-speed modulation and detection methods, enables our designed chip to achieve exceptional area efficiency.

4.4 Chip design

The overall size of the chip is $3\text{ mm} \times 8\text{ mm}$, composed of three layers of 8×8 MRR arrays, two interlayer 1×8 MZI arrays, 248 cross-coupled waveguides, 16 1-input 2-output splitters, and 56 coupling gratings, totaling 528 optical components. It includes 208 tunable units, 32 input ports, and 24 output ports. The radius of the MRRs is designed to be $5\text{ }\mu\text{m}$, with coupling gaps of $0.23\text{ }\mu\text{m}$ for both ends of the cross-coupled waveguides, and an FSR_{mrr} of 17.35 nm .

4.5 Data processing

4.5.1 FCNN

As shown in the Fig. 1b, each column of the input ports is encoded as input information X_1-X_m , and each row of the MRR corresponds to $W_{n1}-W_{nm}$, which are multiplied with X_1-X_m . The drop output ports in each row correspond to Y_1-Y_n . It is important to note that since the MRR array has waveguides connected both horizontally and vertically, not only do the modulation wavelengths of the MRRs in each row need to be different, but the modulation wavelengths of the MRRs in each column must also be staggered to ensure that the optical signals in the vertical and horizontal waveguides of the cross-waveguide do not interfere with each other. The introduction of biases requires using the left ports of the MRR array as

inputs, with each row corresponding to b_1-b_n . Therefore, the formula for the output port can be written as:

$$Y_n = [W_{n1}X_1]_{\lambda(m-n+2)\bmod(m)} + [W_{n2}X_2]_{\lambda(m-n+3)\bmod(m)} + \dots + [W_{nm}X_m]_{\lambda(m-n+m+1)\bmod(m)} + [b_n]_{\lambda(m+1)}, \quad (1)$$

(Supplementary Note 2).

4.5.2 CNN

As shown in Fig. 1c, it is a schematic diagram of a 1×1 convolutional kernel applied to m input channels and n output channels. Each row of inputs corresponds to the vectors X_1-X_m , representing the first pixel of the images from the m input channels starting from the top-left corner (which needs to be scanned over all pixels in time). The eight MRRs in the first row correspond to the convolutional kernel values of these eight input channels, respectively. The output port of the first row is the sum of eight convolution operation, which is the value of output channel 1. Similarly, the n th row represents the value of output channel n . For other convolutional kernel sizes, see the Supplementary Note 3 and Supplementary Figs. 2–5. Therefore, the formula for the output port can be written as:

$$Y_n = [K_{n-1}][X_1]_{\{\lambda(m-n+1)\bmod(m)+1, \lambda(m-n+2)\bmod(m)a2\}} + \dots + [K_{n-m}][X_m]_{\{\lambda(m-n+m)\bmod(m)+1, \lambda(m-n+m+1)\bmod(m)a2\}}, \quad (2)$$

where K represent convolution kernels of different sizes. Additionally, for the 1×1 convolution kernel, the wavelength encoding of each row of input information remains the same, reducing modulation complexity, as the rows represent the same input processed by different convolutional kernels to yield different output channel results. For the CIFAR-10 dataset, which consists of real-world RGB three-channel images, we first expand the original $32 \times 32 \times 3$ image dimension to $32 \times 32 \times 8$. This

dimensional expansion enhances the expressive capability of the image while also aligning with our chip architecture. The $32 \times 32 \times 8$ image is then divided into four groups and fed into the 1×1 convolutional kernel layer, where the 8-channel image is split into four branches: 1-channel, 1-channel, 2-channels, and 4-channels. These branches undergo pooling, 8×8 convolution, 4×4 convolution, and no operation, respectively. The features extracted from different convolution kernels are subsequently concatenated and flattened before being processed by a fully connected layer to produce the 10-class classification result.

4.5.3 PGRNN

The basic matrix–vector multiplication operation and wavelength pairing principle have been demonstrated in FCNN and CNN. Below, the principle of the PGRNN algorithm will be mainly presented, and its relationship with wavelength pairing can be derived from the aforementioned work. We define the weights of the three MRR array groups as matrices **A**, **B**, and **C**, respectively:

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & A_{n3} & A_{nn} \end{bmatrix}, \quad (3)$$

(**B** and **C** are in the same form as **A**). The control vectors for the two layers of MZI arrays are Γ_r^1, Γ_r^2 :

$$\Gamma_r^1 = \begin{bmatrix} \Gamma_{r1}^1 \\ \Gamma_{r2}^1 \\ \vdots \\ \Gamma_{rn}^1 \end{bmatrix}, \quad (4)$$

(Γ_r^2 are in the same form as Γ_r^1). After the first layer of network modulation including MRRs and MZIs, the formula for the output port can be written as:

$$\begin{bmatrix} c_1^t \\ c_2^t \\ \vdots \\ c_n^t \end{bmatrix} = \begin{bmatrix} A_{11}x_1^t + A_{12}x_2^t + \dots + A_{1n}x_n^t \\ A_{21}x_1^t + A_{22}x_2^t + \dots + A_{2n}x_n^t \\ \vdots \\ A_{n1}x_1^t + A_{n2}x_2^t + \dots + A_{nn}x_n^t \end{bmatrix} + \begin{bmatrix} A_{11}'c_1^{t-1} + A_{12}'c_2^{t-1} + \dots + A_{1n}'c_n^{t-1} \\ A_{21}'c_1^{t-1} + A_{22}'c_2^{t-1} + \dots + A_{2n}'c_n^{t-1} \\ \vdots \\ A_{n1}'c_1^{t-1} + A_{n2}'c_2^{t-1} + \dots + A_{nn}'c_n^{t-1} \end{bmatrix} \begin{bmatrix} \Gamma_{r1}^1 \\ \Gamma_{r2}^1 \\ \vdots \\ \Gamma_{rn}^1 \end{bmatrix}, \quad (5)$$

where c^t is the hidden output of current layer, and x^t is the input of the current layer, c^{t-1} is the hidden output of the previous layer. After the second layer of modulation, the formula for the output port can be written as (Supplementary Fig. 9):

$$\begin{aligned}
 \begin{bmatrix} c_1^{t+1} \\ c_2^{t+1} \\ \vdots \\ c_n^{t+1} \end{bmatrix} &= \begin{bmatrix} B_{11}x_1^{t+1} + B_{12}x_2^{t+1} + \cdots + B_{1n}x_n^{t+1} \\ B_{21}x_1^{t+1} + B_{22}x_2^{t+1} + \cdots + B_{2n}x_n^{t+1} \\ \vdots \\ B_{n1}x_1^{t+1} + B_{n2}x_2^{t+1} + \cdots + B_{nn}x_n^{t+1} \end{bmatrix} \\
 &+ \begin{bmatrix} B_{11}'(A_{11}x_1^t + \Gamma_{r1}^1 A_{11}'c_1^{t-1}) + B_{12}'(A_{12}x_2^t + \Gamma_{r1}^1 A_{12}'c_2^{t-1}) + \cdots + B_{1n}'(A_{1n}x_n^t + \Gamma_{r1}^1 A_{1n}'c_n^{t-1}) \\ B_{21}'(A_{21}x_1^t + \Gamma_{r2}^1 A_{21}'c_1^{t-1}) + B_{22}'(A_{22}x_2^t + \Gamma_{r2}^1 A_{22}'c_2^{t-1}) + \cdots + B_{2n}'(A_{2n}x_n^t + \Gamma_{r2}^1 A_{2n}'c_n^{t-1}) \\ \vdots \\ B_{n1}'(A_{n1}x_1^t + \Gamma_{rn}^1 A_{n1}'c_1^{t-1}) + B_{n2}'(A_{n2}x_2^t + \Gamma_{rn}^1 A_{n2}'c_2^{t-1}) + \cdots + B_{nn}'(A_{nn}x_n^t + \Gamma_{rn}^1 A_{nn}'c_n^{t-1}) \end{bmatrix} \begin{bmatrix} \Gamma_{r1}^2 \\ \Gamma_{r2}^2 \\ \vdots \\ \Gamma_{rn}^2 \end{bmatrix}, \quad (6)
 \end{aligned}$$

The modulation method of the third layer is consistent with the aforementioned formulas. Based on this, we experimentally implemented a three-layer network and cyclic operations in time to process information for all time steps, obtaining the output vector $\mathbf{c}^{(t)}$ for each time step. Finally, the output decision was made through a FCNN. The weight matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{\Gamma}_r$ were optimized during the neural network training and deployed onto the MRR and MZI arrays during the experiment.

4.5.4 Photonic neural network training

In the demonstrations of classification tasks and sentiment analysis tasks, photonic neural networks were implemented using the PyTorch deep learning framework. The network parameters were trained using the Adam optimizer. We set the neurons corresponding to each MRR and MZI unit as trainable parameters for gradient descent. In the demonstration of speech recognition task training, we implemented photonic neural networks using the TensorFlow deep learning framework. The network parameters were also trained by the Adam optimizer, and the encoding and decoding of language were completed using the Keras interface.

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s43593-025-00098-6>.

Supplementary Material 1.

Acknowledgements

This work was supported by the National Key Research and Development Program of China under Grant No. 2024YFA1209204, the National Natural Science Foundation of China under Grant No. 12474322, No. 62274179 and No. 62235001, the Innovation Program for Quantum Science and Technology under Grant No. 2021ZD0301500, Beijing Natural Science Foundation under Grant No. Z210004, the International Partnership Program of Chinese Academy of Sciences under Grant No. 02GJHZ2023026FN and the 173 Technical Field Fund under Grant No. 2023-JCJQ-JJ-1017, No.2022-JCJQ-JJ-0446.

Author contributions

XH conceived the research and initiated the project. YW conducted the design. YW, KL, KZ, and ZD developed the theoretical model and performed the numerical simulations. YW, ZW, BN, and TX performed the experiments and data processing. YW and KL prepared the manuscript. SF contributed to the discussions. KL, YY, QY, QS, XH, and QG supervised the project. All authors read and approved the final manuscript.

Funding

This study was funded by National Key Research and Development Program of China (2024YFA1209204); National Natural Science Foundation of China (12474322, 62274179, 62235001); Innovation Program for Quantum Science and Technology (2021ZD0301500); Beijing Natural Science Foundation (Z210004); International Partnership Program of Chinese Academy of Sciences (02GJHZ2023026FN); 173 Technical Field Fund (2023-JCJQ-JJ-1017, 2022-JCJQ-JJ-0446).

Data availability

All data in this study are included in this published article and its supplementary information file.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 8 April 2025 Revised: 7 July 2025 Accepted: 10 July 2025

Published online: 18 August 2025

References

1. T.B. Brown et al., Language models are few-shot learners. *Adv. Neural. Inf. Process. Syst.* **33**, 1877–1901 (2020)
2. E. Strubell, A. Ganesh, A. McCallum, Energy and policy considerations for modern deep learning research. *AAAI* **34**, 13693 (2020)
3. G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Honolulu, HI, 2017), pp. 2261–2269.
4. R. Li, Y. Gong, H. Huang, Y. Zhou, S. Mao, Z. Wei, Z. Zhang, Photonics for neuromorphic computing: fundamentals, devices, and opportunities. *Adv. Mater.* **37**, 2312825 (2025)
5. H. Zhou et al., Photonic matrix multiplication lights up photonic accelerator and beyond. *Light Sci. Appl.* **11**, 30 (2022)
6. P.L. McMahon, The physics of optical computing. *Nat. Rev. Phys.* **5**, 717 (2023)
7. S. Hua et al., An integrated large-scale photonic accelerator with ultralow latency. *Nature* **640**, 361 (2025)
8. S.R. Ahmed et al., Universal photonic artificial intelligence acceleration. *Nature* **640**, 368 (2025)

9. A. Najjar Amiri, A.D. Vit, K. Gorgulu, E.S. Magden, Deep photonic network platform enabling arbitrary and broadband optical functionality. *Nat. Commun.* **15**, 1432 (2024)
10. Y. Huang, W. Liu, R. Sun, T. Fu, Y. Wang, Z. Huang, S. Yang, H. Chen, Diffraction-driven parallel convolution processing with integrated photonics. *Laser Photonics Rev.* **19**, 2400972 (2025)
11. Y. Shen et al., Deep learning with coherent nanophotonic circuits. *Nat. Photon* **11**, 441 (2017)
12. Z. Du et al., Ultracompact and multifunctional integrated photonic platform. *Sci. Adv.* **10**, eadm7569 (2024)
13. K. Liao et al., Hetero-integrated perovskite/Si₃N₄ on-chip photonic system. *Nat. Photon.* **19**, 358–368 (2025)
14. K. Liao, C. Li, T. Dai, C. Zhong, H. Lin, X. Hu, Q. Gong, Matrix eigenvalue solver based on reconfigurable photonic neural network. *Nanophotonics* **11**, 4089 (2022)
15. K. Liao et al., All-optical computing based on convolutional neural networks. *OEA* **4**, 200060 (2021)
16. S. Pai et al., Experimentally realized in situ backpropagation for deep learning in photonic neural networks. *Science* **380**, 398 (2023)
17. S. Bandyopadhyay, A. Sludds, S. Krastanov, R. Hamerly, N. Harris, D. Bunandar, M. Streshinsky, M. Hochberg, D. Englund, Single-chip photonic deep neural network with forward-only training. *Nat. Photon.* **18**, 1335 (2024)
18. T. Fu, Y. Zang, Y. Huang, Z. Du, H. Huang, C. Hu, M. Chen, S. Yang, H. Chen, Photonic machine learning with on-chip diffractive optics. *Nat. Commun.* **14**, 70 (2023)
19. X. Wang, B. Redding, N. Karl, C. Long, Z. Zhu, J. Skowronek, S. Pang, D. Brady, R. Sarma, Integrated photonic encoder for low power and high-speed image processing. *Nat. Commun.* **15**, 4510 (2024)
20. V. Nikkhah, A. Pirmoradi, F. Ashtiani, B. Edwards, F. Aflatouni, N. Engheta, Inverse-designed low-index-contrast structures on a silicon photonics platform for vector-matrix multiplication. *Nat. Photon.* **18**, 501 (2024)
21. T. Wu, M. Menarini, Z. Gao, L. Feng, Lithography-free reconfigurable integrated photonic processor. *Nat. Photon.* **17**, 710 (2023)
22. J. Cheng et al., Multimodal deep learning using on-chip diffractive optics with in situ training capability. *Nat. Commun.* **15**, 6189 (2024)
23. S. Ohno, R. Tang, K. Toprasertpong, S. Takagi, M. Takenaka, Si microring resonator crossbar array for on-chip inference and training of the optical neural network. *ACS Photon.* **9**, 2614 (2022)
24. J. Cheng et al., Human emotion recognition with a microcomb-enabled integrated optical neural network. *Nanophotonics* **12**, 3883 (2023)
25. Z. Xu, T. Zhou, M. Ma, C. Deng, Q. Dai, L. Fang, Large-scale photonic chip-let Taichi empowers 160-TOPS/W artificial general intelligence. *Science* **384**, 202 (2024)
26. S. Hong et al., Versatile parallel signal processing with a scalable silicon photonic chip. *Nat. Commun.* **16**, 288 (2025)
27. D. Wang, Y. Nie, G. Hu, H.K. Tsang, C. Huang, Ultrafast silicon photonic reservoir computing engine delivering over 200 TOPS. *Nat. Commun.* **15**, 10841 (2024)
28. B. Bai et al., Microcomb-based integrated photonic processing unit. *Nat. Commun.* **14**, 66 (2023)
29. H.H. Zhu et al., Space-efficient optical computing with an integrated chip diffractive neural network. *Nat. Commun.* **13**, 1044 (2022)
30. S. Xu, J. Wang, S. Yi, W. Zou, High-order tensor flow processing using integrated photonic circuits. *Nat. Commun.* **13**, 7970 (2022)
31. S. Du, J. Zhang, H. Ouyang, Z. Tao, Q. Yan, H. Hao, J. Zhou, J. You, Y. Tang, T. Jiang, Scalable parallel photonic processing unit for various neural network accelerations. *Photon. Res.* **12**, 2691 (2024)
32. A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM networks, In: *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*, Vol. 4 (IEEE, Montreal, Que., Canada, 2005), pp. 2047–2052.
33. J. Chung, C. Gulcehre, K. Cho, Y. Bengio, *Empirical evaluation of gated recurrent neural networks on sequence modeling*, [arXiv:1412.3555](https://arxiv.org/abs/1412.3555).
34. J. Chen, S.-W. Jun, S. Hong, W. He, J. Moon, Ecton: very low-power recurrent neural network accelerator for real-time inference at the edge. *ACM Trans. Reconfig. Technol. Syst.* **17**, 1 (2024)
35. X. Zhang, Y. Zhu, X. Lou, Reconfigurable and energy-efficient architecture for deploying multi-layer RNNs on FPGA. *IEEE Trans. Circ. Syst. I* **71**, 5969 (2024)
36. G. Mourgas-Alexandris, G. Dabos, N. Passalis, A. Totovic, A. Tefas, N. Pleros, All-optical WDM recurrent neural networks with gating. *IEEE J. Select. Topics Quantum Electron.* **26**, 1 (2020)
37. S. Becker, D. Englund, B. Stiller, An optoacoustic field-programmable perceptron for recurrent neural networks. *Nat. Commun.* **15**, 3020 (2024)
38. J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, D. Brunner, Reinforcement learning in a large-scale photonic recurrent neural network. *Optica* **5**, 756 (2018)
39. T.W. Hughes, I.A.D. Williamson, M. Minkov, S. Fan, Wave physics as an analog recurrent neural network. *Sci. Adv.* **5**, eaay6946 (2019)
40. B. Wang, T.F. De Lima, B.J. Shastri, P.R. Prucnal, C. Huang, Multi-wavelength photonic neuromorphic computing for intra and inter-channel distortion compensations in WDM optical communication systems. *IEEE J. Select. Topics Quantum Electron.* **29**, 1 (2023)
41. A.N. Tait, T.F. De Lima, E. Zhou, A.X. Wu, M.A. Nahmias, B.J. Shastri, P.R. Prucnal, Neuromorphic photonic networks using silicon photonic weight banks. *Sci. Rep.* **7**, 7430 (2017)
42. K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, P. Bienstman, Experimental demonstration of reservoir computing on a silicon photonics chip. *Nat. Commun.* **5**, 3541 (2014)
43. I. Bauwens, K. Harkhoe, E. Gooskens, P. Bienstman, G. Verschaffelt, G. Van Der Sande, Combining a passive spatial photonic reservoir computer with a semiconductor laser increases its nonlinear computational capacity. *Opt. Express* **32**, 24328 (2024)
44. D. Zhong, P. Hou, J. Zhang, W. Deng, T. Wang, Y. Chen, Q. Wu, Excellent predictive-performances of photonic reservoir computers for chaotic time-series using the fusion-prediction approach. *Opt. Express* **31**, 24453 (2023)
45. C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, Going deeper with convolutions, In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Boston, MA, USA, 2015), pp. 1–9.
46. K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Las Vegas, NV, USA, 2016), pp. 770–778.
47. Z. Ye et al., Foundry manufacturing of tight-confinement, dispersion-engineered, ultralow-loss silicon nitride photonic integrated circuits. *Photon. Res.* **11**, 558 (2023)
48. Y. Wang et al., *Compact turnkey soliton microcombs at microwave rates via wafer-scale fabrication*, [arXiv:2502.10941](https://arxiv.org/abs/2502.10941).
49. Li. Deng, The MNIST database of handwritten digit images for machine learning research [Best of the Web]. *IEEE Signal Process. Mag.* **29**, 141 (2012)
50. A. Krizhevsky, *Learning multiple layers of features from tiny images* (2009).
51. A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, C. Potts, *learning word vectors for sentiment analysis, human language technologies*, pp. 142–150 (2011).
52. K. Ito, L. Johnson, *The LJ Speech Dataset*, <https://keithito.com/LJ-Speech-Dataset/>.
53. B. Shen et al., Integrated turnkey soliton microcombs. *Nature* **582**, 365 (2020)
54. H. Feng et al., Integrated lithium niobate microwave photonic processing engine. *Nature* **627**, 80 (2024)
55. Y. Hu et al., *Integrated lithium niobate photonic computing circuit based on efficient and high-speed electro-optic conversion*, [arXiv:2411.02734](https://arxiv.org/abs/2411.02734).