On-chip Reconfigurable Optical Neural Networks

Xianmeng Zhao (zhaoxm@shanghaitech.edu.cn)
ShanghaiTech University
Haibin Lv
ShanghaiTech University
Cheng Chen
ShanghaiTech University
Shenjie Tang
ShanghaiTech University
Xiaoping Liu
ShanghaiTech University
Qin Qi
Shenzhen University

Article

Keywords: artificial neural networks, optical artificial intelligence hardware, artificial intelligence

Posted Date: February 15th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-155560/v1

On-chip Reconfigurable Optical Neural Networks

Xianmeng Zhao[1,2,3,†], Haibin Lv[1,†], Cheng Chen[1], Shenjie Tang[1], Xiaoping Liu[1*], Qin Qi[4,5*]

[1]School of Physical Science and Technology, ShanghaiTech University, Shanghai 201210, China
[2]Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai 201800, China
[3]University of Chinese Academy of Sciences, Beijing 100049, China
[4]College of Physics and Optoelectronic Engineering, Shenzhen University, Shenzhen, 100049, China
[5]Shenzhen Key Laboratory of Intelligent Optical Measurement and Detection, Shenzhen, 100049, China

[†]These authors contributed equally to this work
[*]E-mail: liuxp1@shanghaitech.edu.cn; qi.qin@szu.edu.cn

Implementing artificial neural networks on integrated platforms has generated significant interest in recent years. Several architectures for on-chip optical networks with basic functionalities have been successfully demonstrated, for example, optical spiking neurosynaptic, photonic convolution accelerator, and nanophotonic/electronic hybrid deep neuron networks. In this work, we propose a layered coherent silicon-on-insulator diffractive optical neural network, of which the inter-layer phase delay can be actively tuned. By forming a close-loop with control electronics, we further demonstrate that our fabricated on-chip neural network can be trained in-situ and consequently reconfigured to perform various tasks, including full adder operation and vowel recognition, while achieving almost the same accuracy as networks trained on conventional computers. Our results show that the proposed optical neural network could potentially pave the way for future optical artificial intelligence hardware.

The emergence of recent machine-learning technology, mostly artificial neural networks[1] (ANNs), has profoundly impacted many areas, including image and speech recognition[2,3], brain

circuit reconstruction[4], scientific research[5,6], etc. One of the main bottlenecks faced by the extensive applications of ANNs, especially in complex applications, is the speed of data processing. Traditional central processing units (CPUs) are not the best choice for implementing the ANNs because their computing architecture cannot run massive parallel calculation processes efficiently. Therefore, new electronic and optical architectures for high-speed data processing have received tremendous interest in recent years. Implementing ANNs using dedicated electronic platforms[7,8] has had significant signs of progress. In contrast, the optical domain's research attention just begins to build-up due to its natural capability of parallel data processing. Recent research findings in this direction have been encouraging with the demonstration of photonic convolution accelerator[9,10], hybrid electronic/nanophotonic processor[11], optical spiking neurosynaptic[12], and diffractive deep neural networks[13].

Training the ANNs is a critical step towards their successful deployment. Traditional electronic hardware settings use gradient descent algorithm by error back-propagation[14] to reduce the ANNs' training time and the required resources. In optics, several methods[15,16] are proposed to train the optical neural networks (ONNs) using the optical backpropagation algorithm. However, these techniques need a range of additional optical components on the networks and a set of optical measurements, making their implementation very complicated in integrated photonic platforms. Thus, the ONNs mentioned above are mostly pre-designed via training on the conventional computer using high-level simulators with optical component's transfer matrix models or scattering matrix models. The physical ONNs are then mapped out using process design kits (PDKs) and verified with system-level optical simulation tools. Although this kind of practice seems very standard in the microelectronic industry, it faces

significant challenges in optical networks because they operate as analog circuits, where errors propagate and accumulate. When the ONNs have very limited reconfigurability, PDKs with a small margin of error can ruin the performance of the as-fabricated ONNs because of very tight fabrication errors. For example, coherent ONNs' phase errors, arising from even nanometer-scale waveguide-size deviation, could quickly accumulate and become the most performance-degrading factor[17] (See Supplementary Information Section 1 for details). Thus, to achieve optimized ONNs, it is necessary to gain tuning capability and, most-preferably, in-situ training capability after fabrication.

In this work, we propose an integrated, coherent diffractive optical neural network (DONN) based on a series of cascaded multi-mode interference (MMI)[18] structures. Our architecture can allow for an almost arbitrary transfer matrix out of the continuous unitary space[19], which is crucial for in-situ training (See Supplementary Information Section 1 for details). This DONN encodes the input data in an optical phase domain. The interconnection between adjacent network stages is controlled by a tunable phase delay using phase modulators. Using a closed-loop control with external electronics, the proposed DONN can be trained in-situ through the stochastic gradient descent (SGD) algorithm[20]. Our proposed on-chip DONN's reconfigurability is demonstrated by first training it as a full adder and then retraining it to perform vowel recognition. Compared to the DONN numerically simulated on conventional computers, the on-chip DONN states of both configurations are trained in-situ and achieve similar performance, demonstrating that the in-situ training can ignore the errors from PDKs and fabrication.

Device architecture and experiment setup

The basic unit of an ANN, as illustrated in Fig. 1a, typically accepts *n* inputs and generates single output. A series of such units are hierarchically connected to form an ANN. It typical processes data that are represented by real numbers on traditional computational platforms. However, data can be advantageously encoded in multi-dimensions of optical waves in the optical domain, such as amplitude, phase, frequency, and polarization. Modern, sophisticated optical modulation schemes have demonstrated this advantage in fiber-optic communications. Similarly, the richful encoding schemes allow further neural network architecture innovation.

Intensity-modulated data are commonly used as the input of ONNs. However, when generating them using intensity modulators, e.g., Mach-Zehnder interferometers or ring modulators, are used, optical signals' phase changes accordingly with their intensity. In other words, data themselves have both intensity and phase modulation. Therefore, dealing with data flow inside a coherent DONN demands extra care. Here we resort to a phase modulation scheme, where the input data to our DONN are encoded in the optical phase domain. The data flow is manipulated via phase modulation throughout the network until the output layer, where the optical power is detected. A schematic of our proposed DONN's architecture is shown in Fig. 1b. Each layer of the DONN consists of an optical interference unit (OIU) and an optical phase modulator unit (OPMU). The OPMU introduces certain phase biases, of which the exact values are learned via in-situ training. The OIU acts as the dendrite that determines the amplitude and phase of the optical wave entering the next layer.

Based on the architecture shown in Fig.1b, an eight-layer (one input layer, six hidden layers, and one output layer) DONN is designed and fabricated on a silicon-on-insulator substrate using electron-beam lithography (See Supplementary Information Section 1 for details). A

micrographic photo of our fabricated DONN chip is shown in Fig. 1c. Coupling into and out of this chip is realized via a single cleaved fiber end and a fiber array in a vertical grating coupling scheme, respectively. The input light is split into eight channels by cascaded 1×2 MMIs. Due to the limited available in-house control electronics, the currently proposed DONN is designed to have five maximum input channels and five maximum output ports (see Fig. 1c). The actual number of input/output ports involved in computation can be less depending on the task. As shown in the schematics of our testing setup in Fig. 1d, the state of each thermo-modulator (5 for encoding input data and 30 for introducing phase biasing) is controlled independently via its corresponding digital-analog conversion (DAC) electronics. The optical signal from the output layer is converted to electric signals and then digitized using an analog-digital conversion (ADC) electronics. Using an in-house developed control software run on a PXI electronic instrumentation platform, closed-loop control of these thermo-modulators using feedback from the digitized output optical signal is established.

Neural networks are usually trained by minimizing cost function (CF), representing the discrepancy between target and actual outputs. Here an adaptive moment estimation[21] (Adam) gradient descent algorithm is used to minimize the CF. This method is known to have the ability to overcome the noise of gradients and damp oscillations across ravines. Successful implementation of this algorithm in our closed-loop control scheme, as shown above, requires the calculation of the gradient of the CF as accurately as possible, which itself is a function of the phase biasing of the 30 phase modulators. Here, it is realized by using a high-order finite difference method based on Lagrange interpolation[22]. A complete set of gradient information can be obtained by applying this method iteratively to all the 30 phase modulators. Besides, to

suppress system noises in the experiment and further improve the gradient measurement's accuracy, the measurement bandwidth is reduced by repeating the same measurement and averaging the obtained gradient values (More details are provided in Supplementary Information Section 2).

Experiment

One-bit full adder. An arithmetic logic unit (ALU) is the basic building block of digital computers[23]. As a vital component of the ALU, a one-bit full adder, as shown in Fig.2a, can be formulated by[24]:

$$p_n = a_n \oplus b_n$$

$$g_n = a_n \cdot b_n$$

$$C_n = p_n \cdot C_{n-1} + g_n$$

$$S_n = C_{n-1} \oplus p_n$$

Here a_n and b_n are addends, C_{n-1} is carry signal from the previous bit, p_n and g_n are propagate and generate signals that equal to the result of a half adder, C_n is output carry signal, and S_n is output sum. For demonstration purpose and simplicity, implementing a full adder can be realized in two consecutive steps as a proof-of-concept DONN. The proposed DONN is first trained to calculate p_n and g_n using input a_n and b_n, which is labeled as Step A (Fig.2a). Then, the DONN is trained and reconfigured to calculate C_n and S_n using p_n, g_n and C_{n-1} as the input, which is labeled as Step B (Fig.2a). The cost function for both steps is defined as the mean-squared error between normalized output and target. The input bits are encoded in the optical phase domain, where 0 phase represents input digit '0', and π phase

represents input digit '1'. The top two output ports are selected as the neural network's prediction. A full adder is successfully implemented using the two-step approach outlined above on the proposed DONN architecture in a closed-loop configuration (Fig. 1d). The complete set of output for all possible input combinations are plotted in Fig. 2b. Here, the output digit '0' (output digit '1') corresponds to a normalized low (high) output power. They can be easily discriminated from each other because there is a significant margin between the two output digits (the gray region in Fig. 2b). Due to the phase errors in the fabricated DONN, the as-fabrication DONN fails to work correctly and give false results (See Supplementary Information Section 3 for details). Therefore, closed-loop in-situ training is used to reconfigure the DONN to a correct state. The corresponding experimental results after training (red and blue dots in Fig. 2b) agree precisely with the truth tables of a full adder (red and blue crosses in Fig. 2b) (See Supplementary Information Section 3 for details). Notice that step B's input still uses 0 phase or π phase, instead of the direct output from Step A, to represent input digit '0' and '1'. Such substitution would be practically viable using an electronic buffer layer between these two steps if each of them is implemented on a separate on-chip DONN.

Vowel recognition. To demonstrate the reconfigurability and the error tolerance, a more complex task is investigated, involving training the proposed DONN as a vowel classifier. Given that five maximumly inputs can be simultaneously driven in our testing setup, five vowel features are used to classify four different kinds of vowels, randomly selected from the complete set of eleven vowels. Our experiment uses a frequently used data set[25], which has data from 15 different people. Each people has six frames of speech from eleven vowels. Five input features of each vowel, corresponding to the five available data input channels, are randomly selected.

In our phase-encoding scheme, the input data's numerical value is normalized and rescaled between 0 to 2π phase. Our proposed DONN is first trained on a conventional computer (see Supplementary Information Section 4 for details). To investigate the impact of fabrication errors on the performance of the DONN, the transfer matrix of MMI W used in training is replaced with a parametrized matrix, $T = (1 - \alpha)W + \alpha R$, where R is a Haar-random perturbation, and α stands for the degree of error ($0 < \alpha < 1$)[19]. The test set's recognition rate as a function of α is calculated using this matrix T and shown in Fig. 3a. It indicates that the fabrication error has a substantial negative impact on the DONN's performance. This kind of impact is confirmed with the as-fabricated DONN's testing result (See Supplementary Information Section 4 for details), which gives very inferior vowel recognition accuracy. For this reason, practical DONN requires preferably in-situ training to compensate for the fabrication errors, which is performed again on our chip via in-situ closed-loop fashion to obtain the states of all the 30 tunable phase shifters in the hidden layers. The evolution of the cost function with the training epoch is shown in Fig. 3b. Our DONN's learning curve is steep for the first 50 epochs and then starts to saturate, approaching a maximum classification accuracy of 90.5% for the train set (164/180 are correctly identified, Fig.3c). The accuracy level is very close to that obtained via a conventional 64-bit computer (94.4%, Fig.3d). The trained DONN is then applied to the test set and achieves a classification accuracy of 88.3% (161/180 are correctly identified, Fig.3e). Its counterpart using a 64-bit computer achieves an accuracy of 93.3% (Fig.3f). These results suggest that our proposed on-chip DONN can be efficiently trained/reconfigured via in-situ closed-loop architecture and achieve similar accuracy with the DONN trained on traditional computers. The accuracy of DONN may be further improved by scaling up its size to deal with

more vowel features and deliberately engineered nonlinear functionalities in the hidden layers.

Conclusion

In this work, a cascaded-MMI-based DONN, of which the data are manipulated in the phase domain, is proposed and demonstrated experimentally. This DONN is capable of in-situ training and thus being reconfigured to perform different tasks. Our findings show that its performance is very promising, e.g., in both full-adder operation and vowel recognition, with training results comparable to its numerical counterpart trained on a computer. Our preliminary demonstration here strongly suggests DONN may have great potentials as a future computing architecture. With a rough estimation, when the DONN scale increases, the computational advantage of in-situ DONN training starts to emerge compared with its electronic counterpart (see Supplementary Information Section 5 for details). With the rapid development of high-speed control electronics, especially application-specific integrated circuit (ASIC), very high computation-speed and high energy-efficient DONN systems will be within reach soon.

Methods

Training procedure. We apply the following steps to train the DONN on chip:

1. For digital data, use 0 phase to denote digit '0' and π phase to denote digit '1'; for analog data, normalize the input data and rescale them to the range of $[0,\ 2\pi]$ and ;
2. Encode the input data according to their phase representation by applying corresponding voltages to phase shifters on the input channels;
3. Initialize the trainable variables, i.e., the phase of all the phase modulator in the hidden layer;

4. In each iteration, implement consecutively five forward propagations with the parameters $\boldsymbol{x}_0, \boldsymbol{x}_0 \mp h, \boldsymbol{x}_0 \mp 2h$, where $\boldsymbol{x}_0$ is the current phase state, h is a small phase variation, and calculate the gradient of the variables using the Lagrange interpolation method;
5. Update the trainable variables with the Adam method;
6. Repeat steps 4 to 5 until the CF converges to a preset value.

Reference

1 LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436-444 (2015).

2 Krizhevsky, A., Sutskever, I. & Hinton, G. E. in *Advances in neural information processing systems.* 1097-1105 (2012).

3 Graves, A., Mohamed, A.-r. & Hinton, G. in *2013 IEEE international conference on acoustics, speech and signal processing.* 6645-6649 (IEEE).

4 Helmstaedter, M. *et al.* Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature* **500**, 168 (2013).

5 Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O. & Walsh, A. Machine learning for molecular and materials science. *Nature* **559**, 547-555 (2018).

6 Raccuglia, P. *et al.* Machine-learning-assisted materials discovery using failed experiments. *Nature* **533**, 73 (2016).

7 Poon, C.-S. & Zhou, K. Neuromorphic silicon neurons and large-scale neural networks: challenges and opportunities. *Frontiers in neuroscience* **5**, 108, https://doi.org/10.3389/fnins.2011.00108 (2011).

8 Graves, A. *et al.* Hybrid computing using a neural network with dynamic external memory.

Nature **538**, 471 (2016).

9 Feldmann, J. *et al.* Parallel convolutional processing using an integrated photonic tensor core. *Nature* **589**, 52-58 (2021).

10 Xu, X. *et al.* 11 TOPS photonic convolutional accelerator for optical neural networks. *Nature* **589**, 44-51 (2021).

11 Shen, Y. *et al.* Deep learning with coherent nanophotonic circuits. *Nature Photonics* **11**, 441-446 (2017).

12 Feldmann, J., Youngblood, N., Wright, C. D., Bhaskaran, H. & Pernice, W. H. P. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature* **569**, 208-214 (2019).

13 Lin, X. *et al.* All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004-1008 (2018).

14 Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning internal representations by error propagation. No. ICS-8506. (California Univ San Diego La Jolla Inst for Cognitive Science, 1985).

15 Wagner, K. & Psaltis, D. Multilayer optical learning networks. *Applied Optics* **26**, 5061-5076 (1987).

16 Hughes, T. W., Minkov, M., Shi, Y. & Fan, S. Training of photonic neural networks through in situ backpropagation and gradient measurement. *Optica* **5**, 864 (2018).

17 Fang, M. Y., Manipatruni, S., Wierzynski, C., Khosrowshahi, A. & DeWeese, M. R. Design of optical neural networks with component imprecisions. *Opt Express* **27**, 14009-14029 (2019).

18 Soldano, L. B. & Pennings, E. C. Optical multi-mode interference devices based on self-imaging: principles and applications. *Journal of lightwave technology* **13**, 615-627 (1995).

19 Saygin, M. Y. *et al.* Robust Architecture for Programmable Universal Unitaries. *Phys Rev Lett* **124**, 010501 (2020).

20 Bottou, L. in *Proceedings of COMPSTAT'2010*, 177-186 (Springer, 2010).

21 Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. Preprint at https://arxiv.org/abs/1412.6980 (2014).

22 Berrut, J.-P. & Trefethen, L. N. Barycentric lagrange interpolation. *SIAM review* **46**, 501-517 (2004).

23 Rabaey, J. M., Chandrakasan, A. P. & Nikolić, B. *Digital integrated circuits: a design perspective*. Vol. 7 (Pearson Education Upper Saddle River, NJ, 2003).

24 Ying, Z. *et al.* Electronic-photonic arithmetic logic unit for high-speed computing. *Nature Communications* **11**, 1-9 https://doi.org/10.1038/s41467-020-16057-3 (2020).

25 Deterding, D. H. *Speaker normalisation for automatic speech recognition*, University of Cambridge, (1990).

Acknowledgments

We acknowledge the financial support from Shanghai Municipal Science and Technology Major Project (Grant No. 2017SHZDZX03) and the start-up funding from ShanghaiTech University. Q.Q. acknowledges the financial support from Shenzhen Municipal Science and Technology (Grant No.860-000002081205, Grant No. and Grant No.ZDSYS20200107103001793) and the start-up funding from Shenzhen University.

Author contributions

X.Z., H.L., and X.L. conceived the idea and developed a theoretical model for on-chip training. X.Z. and

H.L. designed the photonic chip and built the experimental setup. X.Z. and C.C. conducted the experiment. X.Z., C.C., and S.T. prepared the data and developed the code for on-chip training. X.L. and Q.Q. supervised the research. All authors contributed to writing the paper.

Competing financial interests

The authors declare no competing financial interests.

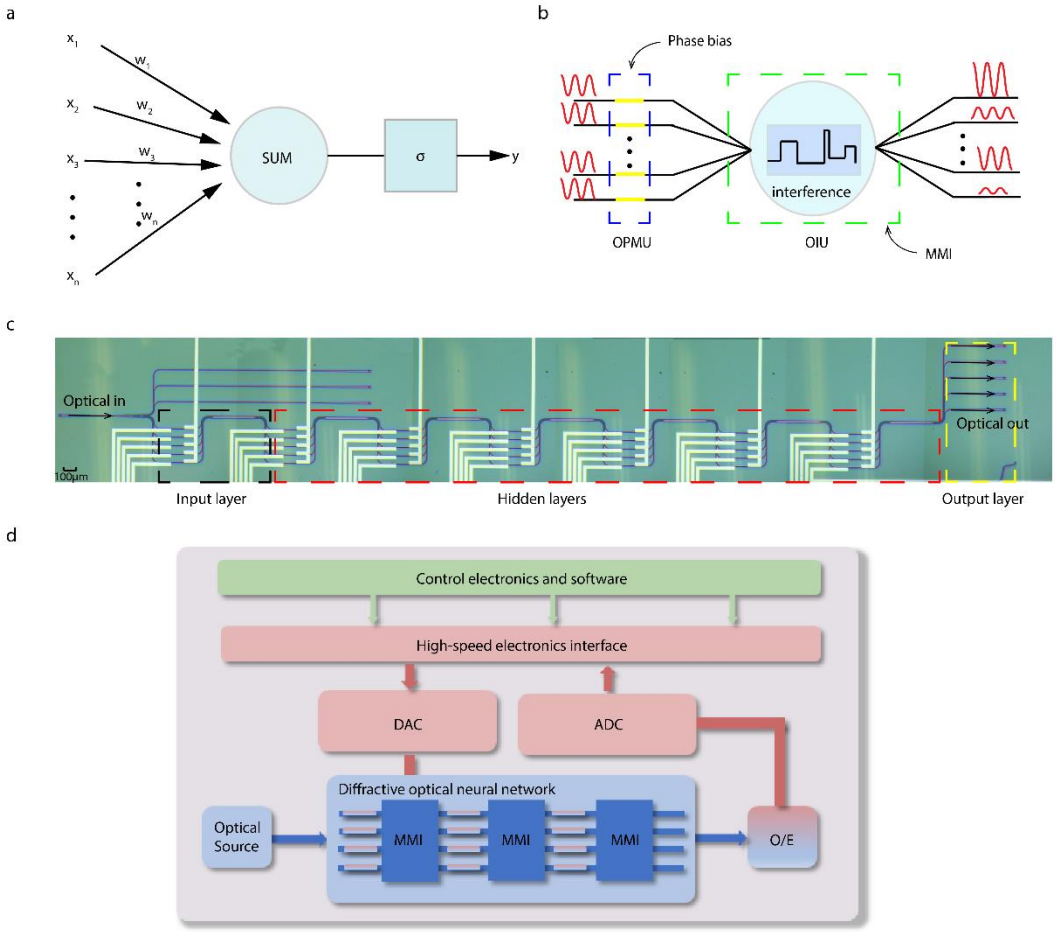

Figure 1 | Device architecture and experiment setup. a, Illustration of a basic unit of traditional fully connected neural networks, consisting of first a linear combination function accepting *n* inputs, SUM, and then a nonlinear activation function, σ, eventually generating an output. **b**, A schematic of the basic unit of our DONN, consisting of optical phase modulator unit (OPMU) (blue box) and optical interference unit (OIU) (green box). It accepts *n* inputs and generates *n* outputs. **c,** An optical micrograph of our fabricated DONN, which includes a 1×8 optical power splitting section, 7 basic units, and 35 thermo-optical modulators. The input layer (enclosed in the black box) encodes the input data. The hidden layers (enclosed in the red box) consist of the learnable parameters, i.e., the phase delay imposed on the optical wave. The yellow box denotes the output layer. **d**, Illustration of our closed-loop controlled testing setup.

O/E, photoelectric conversion; DAC, digital-to-analog conversion; ADC, analog-to-digital conversion.

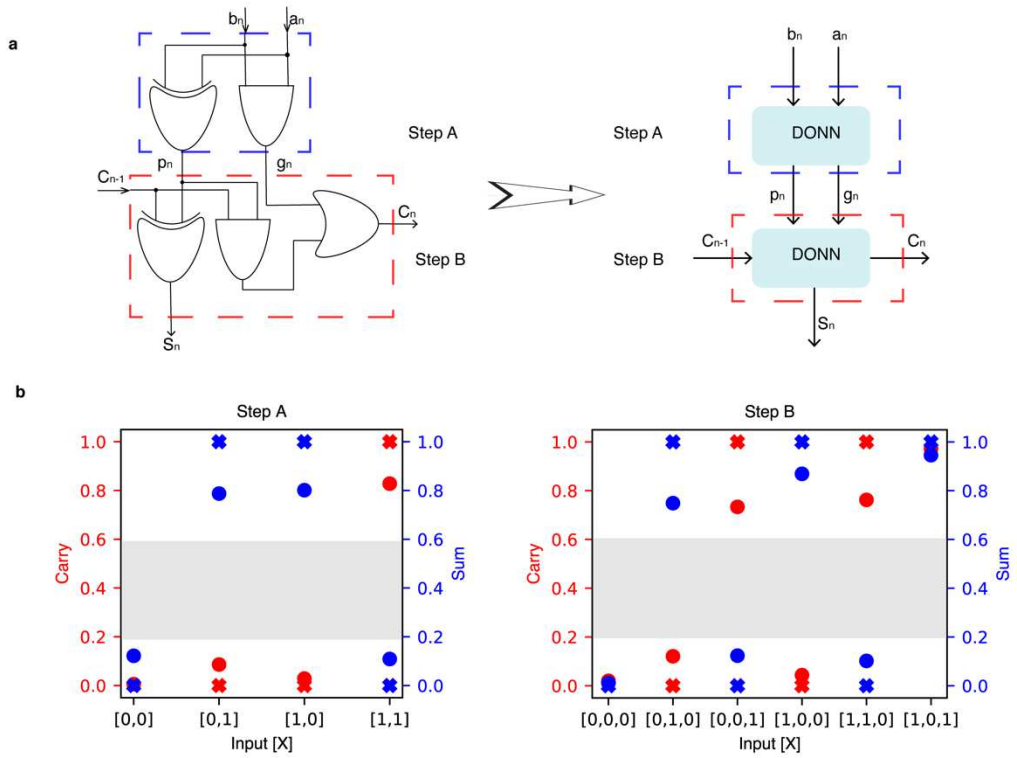

Fig 2 | DONN used as Full Adder. a, A schematic of the full adder with three inputs and two outputs, which is implemented optically into two-step operations using our reconfigurable DONN. **b**, Left panel: In Step A, the normalized predicted values of carry signal (red dot) and sum (blue dot) are plotted, while the targeted values of carry signal (red cross) and sum (blue cross) are also illustrated. The input X is denoted as $[a_n, b_n]$. Right panel: In Step B, the normalized predicted values of carry signal (red dot) and sum (blue dot) are plotted, while the targeted values of carry signal (red cross) and sum (blue cross) are also illustrated. The input X is labeled as $[C_{n-1}, p_n, g_n]$.

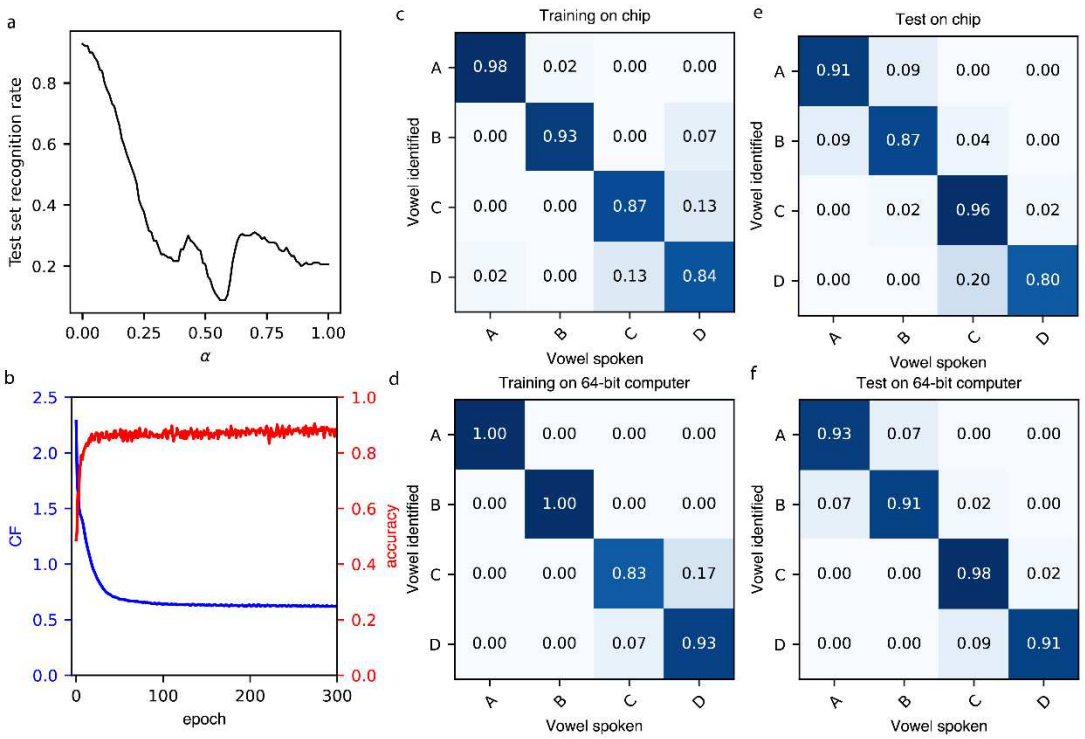

Fig 3 | Vowel recognition. a, Recognition rate of the test set versus α. **b,** Cost function and accuracy of train set versus epoch. **c**, **d**, Confusion matrix of the train set for our DONN trained on-chip (c) and on a 64-bit computer (d). The diagonal elements indicate the correct prediction, and the off-diagonal elements are the wrong prediction. **e**, **f**, Similar to c, d, but the test set is used as input data.

Figures

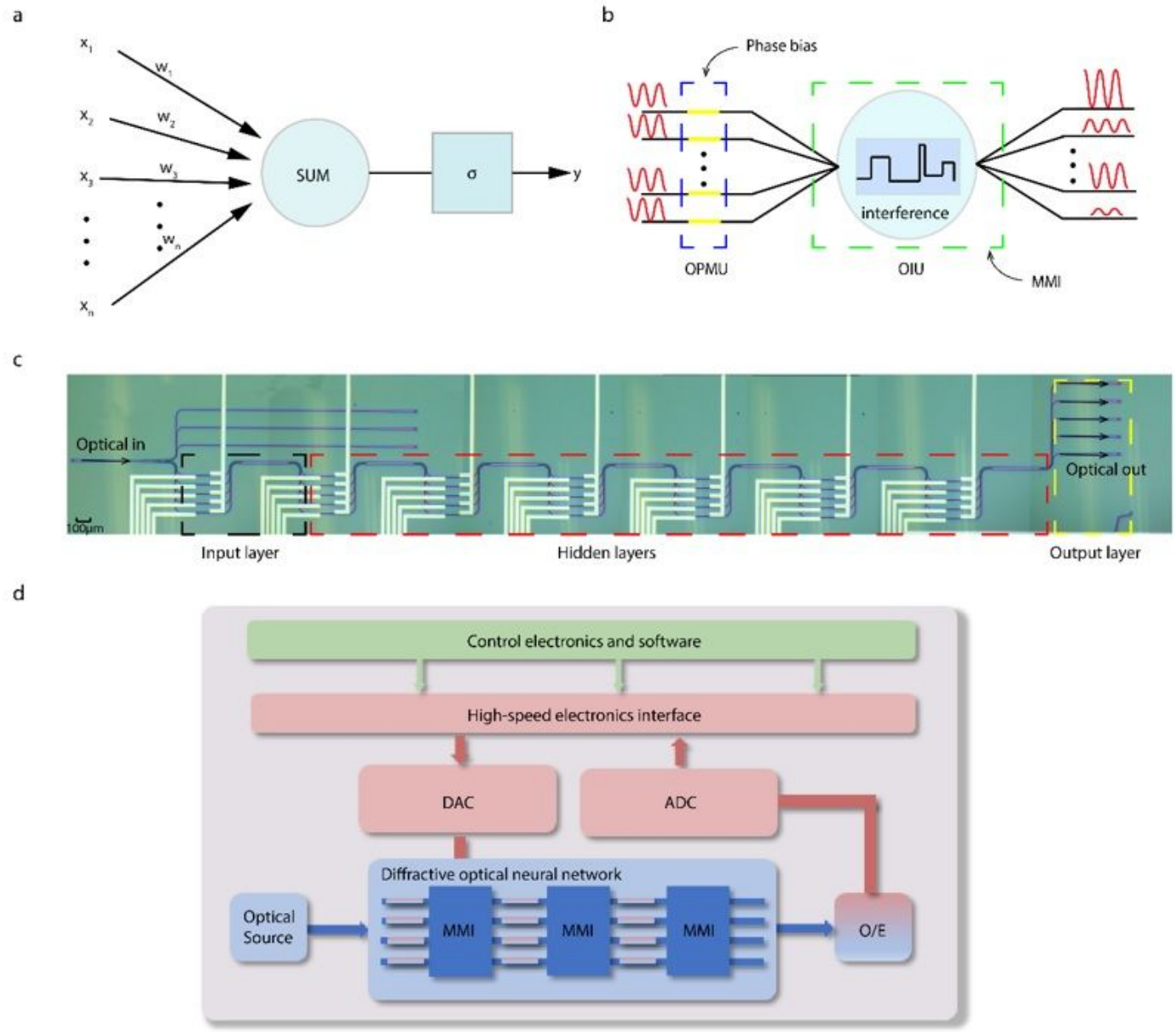

Figure 1

Device architecture and experiment setup. a, Illustration of a basic unit of traditional fully connected neural networks, consisting of first a linear combination function accepting n inputs, SUM, and then a nonlinear activation function, s, eventually generating an output. b, A schematic of the basic unit of our DONN, consisting of optical phase modulator unit (OPMU) (blue box) and optical interference unit (OIU) (green box). It accepts n inputs and generates n outputs. c, An optical micrograph of our fabricated DONN, which includes a 1×8 optical power splitting section, 7 basic units, and 35 thermo-optical modulators. The input layer (enclosed in the black box) encodes the input data. The hidden layers (enclosed in the red box) consist of the learnable parameters, i.e., the phase delay imposed on the optical

wave. The yellow box denotes the output layer. d, Illustration of our closed-loop controlled testing setup. O/E, photoelectric conversion; DAC, digital-to-analog conversion; ADC, analog-to-digital conversion.

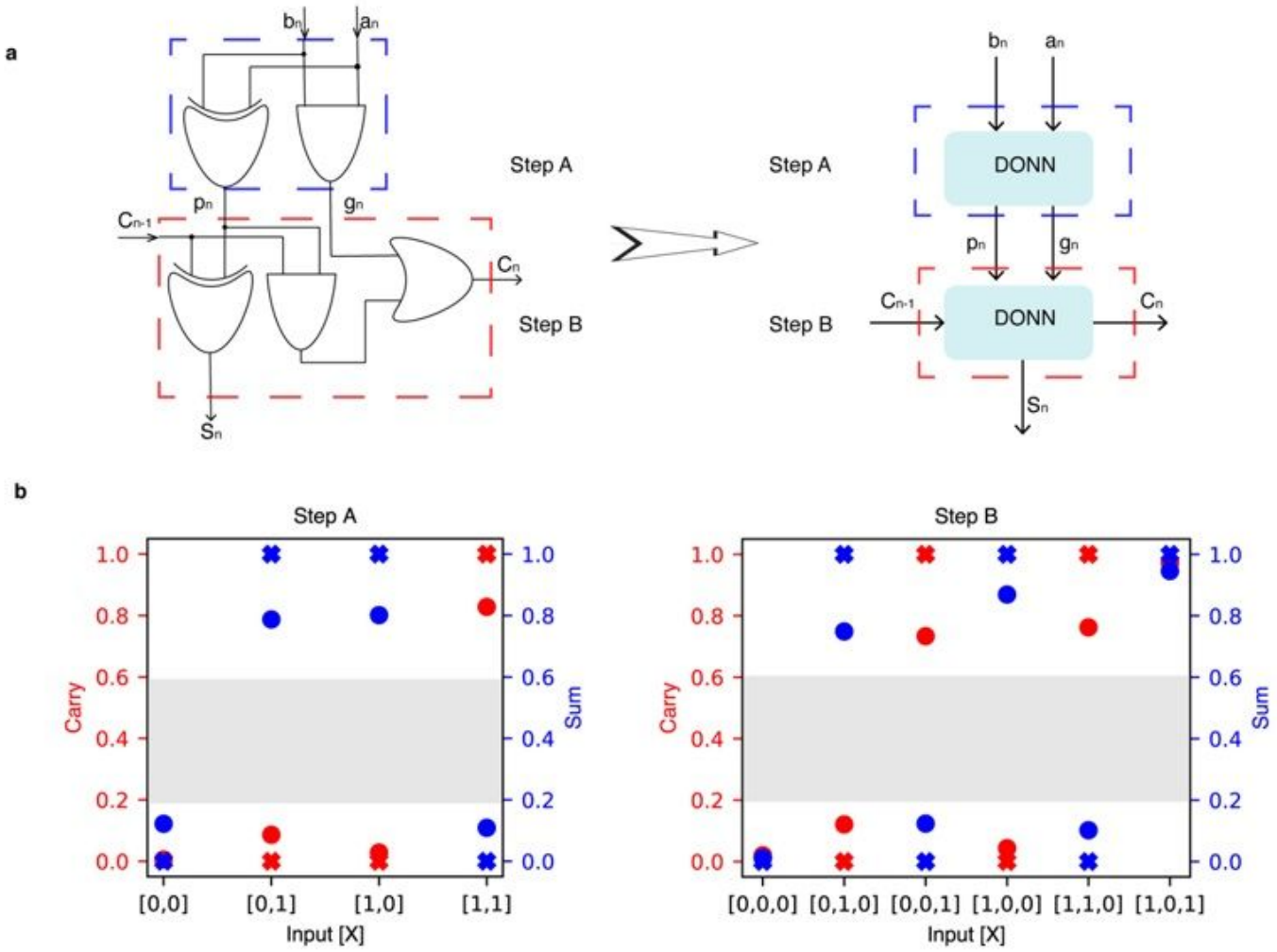

Figure 2

DONN used as Full Adder. a, A schematic of the full adder with three inputs and two outputs, which is implemented optically into two-step operations using our reconfigurable DONN. b, Left panel: In Step A, the normalized predicted values of carry signal (red dot) and sum (blue dot) are plotted, while the targeted values of carry signal (red cross) and sum (blue cross) are also illustrated. The input X is denoted as . Right panel: In Step B, the normalized predicted values of carry signal (red dot) and sum (blue dot) are plotted, while the targeted values of carry signal (red cross) and sum (blue cross) are also illustrated. The input X is labeled as [Cn-1, pn, gn]

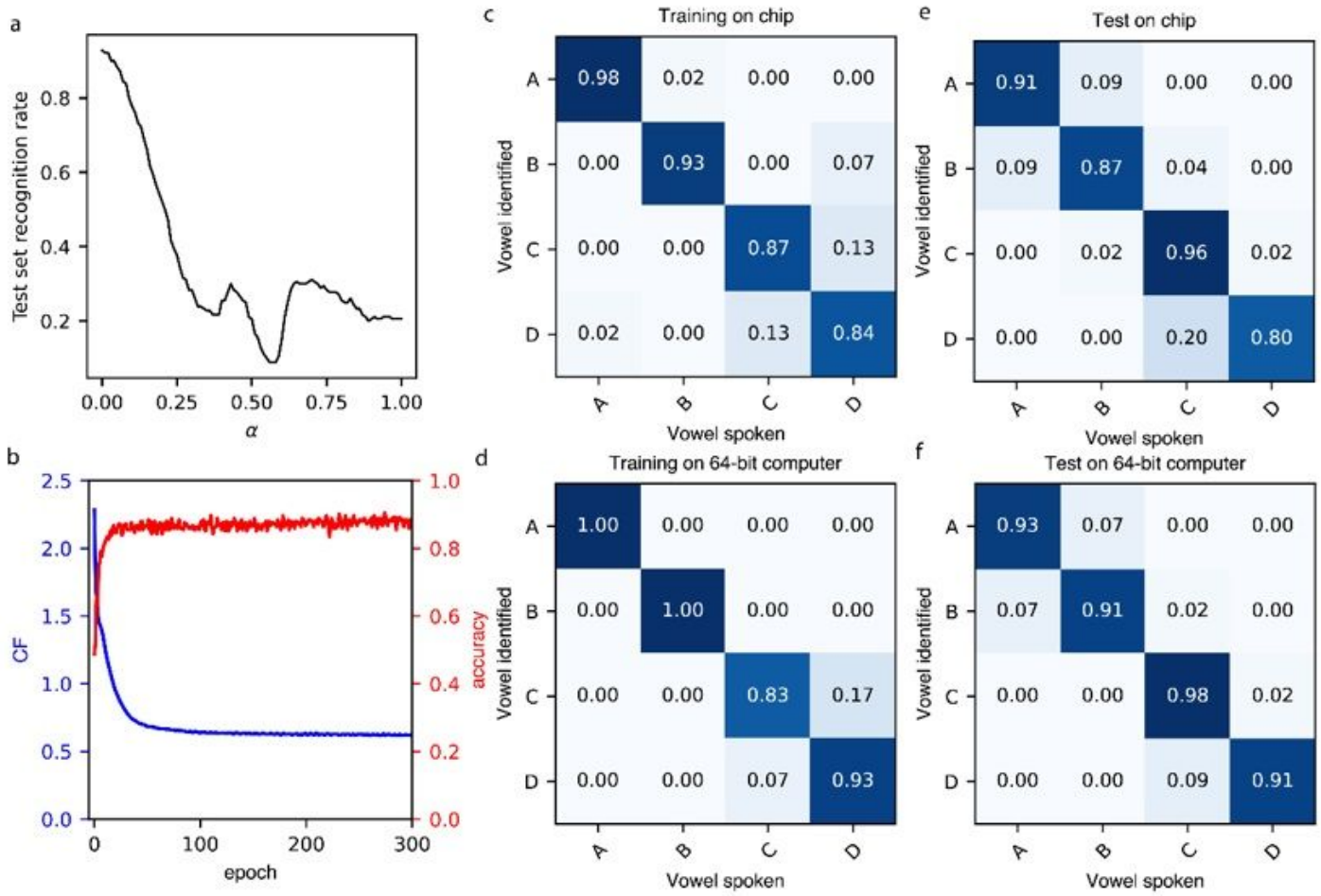

Figure 3

Vowel recognition. a, Recognition rate of the test set versus . b, Cost function and accuracy of train set versus epoch. c, d, Confusion matrix of the train set for our DONN trained on-chip (c) and on a 64-bit computer (d). The diagonal elements indicate the correct prediction, and the off-diagonal elements are the wrong prediction. e, f, Similar to c, d, but the test set is used as input data.

Supplementary Files

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

- SMOnchipreconfigurableopticalneuralnetworks.docx