PHOTONICS Research

Backpropagation through nonlinear units for the all-optical training of neural networks

Xianxin Guo,[1,2,3,5,†] 🄍 Thomas D. Barrett,[2,6,†] 🄍 Zhiming M. Wang,[1,7] and A. I. Lvovsky[2,4,8]

[1]*Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China*
[2]*Clarendon Laboratory, University of Oxford, Oxford OX1 3PU, UK*
[3]*Institute for Quantum Science and Technology, University of Calgary, Calgary, Alberta T2N 1N4, Canada*
[4]*Russian Quantum Center, Skolkovo 143025, Moscow, Russia*
[5]*e-mail: xianxin.guo@physics.ox.ac.uk*
[6]*e-mail: thomas.barrett@physics.ox.ac.uk*
[7]*e-mail: zhmwang@uestc.edu.cn*
[8]*e-mail: alex.lvovsky@physics.ox.ac.uk*

We propose a practical scheme for end-to-end optical backpropagation in neural networks. Using saturable absorption for the nonlinear units, we find that the backward-propagating gradients required to train the network can be approximated in a surprisingly simple pump-probe scheme that requires only simple passive optical elements. Simulations show that, with readily obtainable optical depths, our approach can achieve equivalent performance to state-of-the-art computational networks on image classification benchmarks, even in deep networks with multiple sequential gradient approximation. With backpropagation through nonlinear units being an outstanding challenge to the field, this work provides a feasible path toward truly all-optical neural networks. © 2021 Chinese Laser Press

https://doi.org/10.1364/PRJ.411104

1. INTRODUCTION

Machine learning (ML) is changing the way in which we approach complex tasks, with applications ranging from natural language processing [1] and image recognition [2] to artificial intelligence [3] and fundamental science [4,5]. At the heart (or "brain") of this revolution are artificial neural networks (ANNs), which are universal function approximators [6,7] capable, in principle, of representing an arbitrary mapping of inputs to outputs. Remarkably, their function only requires two basic operations: matrix multiplication to communicate information between layers, and some nonlinear transformation of individual neuron states (activation function). The former accounts for most of the computational cost associated with ML. This operation can, however, be readily implemented by leveraging the coherence and superposition properties of linear optics [8]. Optics is therefore an attractive platform for realizing the next generation of neural networks, promising faster computation with low power consumption [9–13].

Proposals for optical neural networks (ONNs) have been around for over 30 years [14,15] and have been realized in both free-space [16–18] and integrated [9,10] settings. However, the true power of neural networks is not only that they can approximate arbitrary functions, but also that they can "learn" that approximation. The training of neural networks is, almost

universally, achieved by the backpropagation algorithm [19]. Implementing this algorithm optically is challenging because it requires the response of the network's nonlinear elements to be different for light propagating forward or backward. Confronted with these challenges, existing ONNs are actually trained with, or heavily aided by, digital computers [9,16,18,20]. As a result, the great advantages offered by optics remain largely unexploited. Developing an all-optically trained ONN to leverage these advantages remains an unsolved problem. Here, we address this challenge and present a practical training method capable of backpropagating the error signal through nonlinear neurons in a single optical pass.

The backpropagation algorithm aims to minimize a loss function that quantifies the divergence of the network's current performance from the ideal, via gradient descent [19]. To do so, the following steps are repeated until convergence: (1) forward propagation of information through the network; (2) evaluation of the loss function gradients with respect to the network parameters at the output layer; (3) backpropagation of these gradients to all previous layers; (4) parameter updates in the direction that maximally reduces the loss function. Forward propagation [step (1)] requires the aforementioned matrix multiplication, which maps information between layers, and a suitable nonlinear activation function, which is applied

individually to each neuron. While this nonlinearity has so far been mostly applied digitally in hybrid optical-electronic systems [9,20–22]—at the cost of repeatedly measuring and generating the optical state—recent work has also realized optical nonlinearities [18,23–25].

However, obtaining and backpropagating the loss-function gradients [(steps (2) and (3)] remains an outstanding problem in an optical setting. While backpropagating through the linear interconnection between layers is rather straightforward, as linear optical operations are naturally bidirectional, the nonlinearity of neurons is a challenge. This is because the backward-propagating signal must be modulated by the derivatives of the activation function of each neuron at its current input value, and these derivatives are not readily available in an ONN.

In 1987, Wagner *et al.* suggested that a feedforward ONN could be implemented and trained by using Fabry–Perot etalons to approximate the required forward and backward response of a sigmoid nonlinearity [26]. However, this backpropagation approach was never realized or even analyzed in detail, largely due to its inherent experimental complexity, with a subsequent ONN demonstration instead using digitally calculated errors [27]. A further approach to an optically trained feedforward network was proposed by Cruz-Cabrera *et al.* [28]. The authors used a highly nonstandard network architecture that transforms a "continuum of neurons" (a wavefront) as it passes through a nonlinear crystal using cross-phase modulation with a secondary "weight" beam. In a proof-of-concept experiment, the learning of two-bit logic was demonstrated.

An additional challenge is to map from the gradients with respect to the (platform-agnostic) weight matrices to the physical parameters that control these matrices in a specific ONN platform. In 2018, Hughes *et al.* [20] proposed a method to directly obtain the gradients of these control parameters by an additional forward-propagating step. However, this scheme assumes computing the derivatives of the activation functions digitally and applying them to the backpropagating signal electro-optically.

This work directly addresses the issue of optical backpropagation through nonlinear units in a manner that is both consistent with modern neural network architectures and compatible with leading ONN designs [9,20,29,30]. We consider an optical nonlinearity based on saturable absorption (SA) and show that, with the forward-propagating features and the backward-propagating errors taking the roles of pump and probe, respectively, backpropagation can be realized using only passive optical elements. Our method is effective and surprisingly simple, with the required optical operations for both forward and backward propagation realized using the same physical elements. Simulations with physically realistic parameters show that the proposed scheme can train networks to performance levels equivalent to state-of-the-art ANNs. When combined with optical calculation of the error term at the output layer via interference, this presents a path to the all-optical training of ONNs.

2. IMPLEMENTING OPTICAL BACKPROPAGATION

We begin by recapping the operation of a neural network before discussing optical implementations. Seeded with data at the

input layer ($a^{(0)}$), forward-propagation maps the neuron activations from layer $l - 1$ to the neuron inputs at layer l as

$$z_j^{(l)} = \sum_i w_{ji}^{(l)} a_i^{(l-1)}, \qquad (1)$$

via a weight matrix $w^{(l)}$, before applying a nonlinear activation function individually to each neuron, $a_j^{(l)} = g(z_j^{(l)})$ (with subscripts labelling individual neurons).

At the output layer, we evaluate the loss function, $\mathscr{L}$, and calculate its gradient with respect to the weights,

$$\frac{\partial \mathscr{L}}{\partial w_{ji}^{(l)}} = \frac{\partial \mathscr{L}}{\partial z_j^{(l)}} \frac{\partial z_j^{(l)}}{\partial w_{ji}^{(l)}} = \delta_j^{(l)} a_i^{(l-1)}, \qquad (2)$$

where $\delta_j^{(l)} \equiv \partial \mathscr{L}/\partial z_j^{(l)}$ is commonly referred to as the "error" at the jth neuron in the lth layer. From the chain rule, we have

$$\delta_j^{(l)} = \sum_k \frac{\partial \mathscr{L}}{\partial z_k^{(l+1)}} \frac{\partial z_k^{(l+1)}}{\partial z_j^{(l)}} = g'(z_j^{(l)})\rho_j^{(l+1)}, \qquad (3)$$

where $\rho_j^{(l+1)} = \sum_k w_{kj}^{(l+1)} \delta_k^{(l+1)}$. Given the error at the output layer, i.e., $\delta^{(L)}$, which is calculated directly from the loss function, the errors $\delta^{(L-1)}, \ldots, \delta^{(1)}$ for all preceding layers are sequentially found using Eq. (3). These errors, as well as the activations $a^{(l-1)}$ of all neurons, allow one to find the gradients [Eq. (2)] of the error function with respect to all the weights and hence apply the gradient descent.

The transformation [Eq. (1)] is readily implemented as a linear optical (interferometric) operation, with the neurons represented by real-valued field amplitudes in different spatial modes [8]. An experimental realization of this operation, with the matrix and vector dimension up to 56, has recently been reported by our group [31]. Remarkably, calculating $\rho^{(l+1)}$ in the right-hand side of the backpropagation in Eq. (3) involves the same weight matrix, meaning that it can be implemented by physical backward propagation of an optical signal through the same linear optical arrangement [27], as shown in Fig. 1(a). However, multiplying this signal by the derivative of the activation function, $g'(z^{(l)})$, is a challenge without invoking digital electronics.

To address this challenge, we require an optical implementation of the activation function with the following features: (i) nonlinear response for the forward input; (ii) linear response for the backward input; (iii) modulation of backward input with the derivative of the nonlinear function. While it is natural to use nonlinear optics for this purpose, it is difficult to satisfy the requirement that the unit must respond differently to forward- and backward-propagating light. Here, we show that this problem can be addressed using saturable absorption in the well-known pump-probe configuration.

Consider passing a strong pump, E_P, and a weak probe, E_{pr}, through a two-level medium (e.g., atomic vapor). The pump transmission is then a nonlinear function of the input,

$$E_{P,\text{out}} = g(E_{P,\text{in}}) = \exp\left(-\frac{\alpha_0/2}{1 + E_{P,\text{in}}^2}\right) E_{P,\text{in}}, \qquad (4)$$

where α_0 is the resonant optical depth and all fields are assumed to be normalized by the saturation threshold. Figure 2(a) plots the pump transmission $g(\cdot)$ at α_0 of 1 and 30. High optical depth induces strong nonlinearity in the unsaturated region,
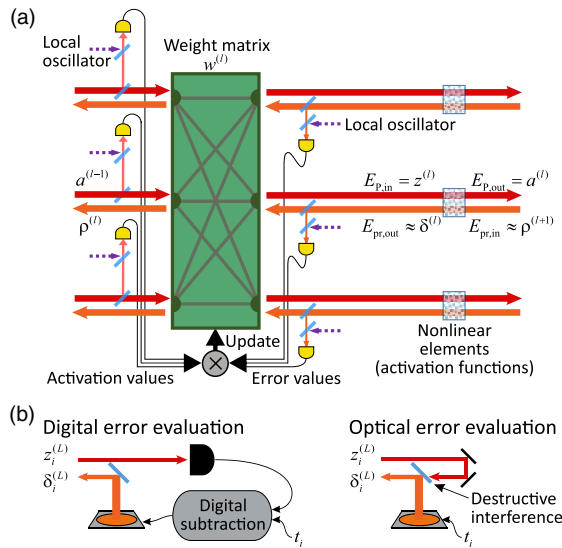
Fig. 1. ONN with all-optical forward- and backward-propagation. (a) A single ONN layer that consists of weighted interconnections and an SA nonlinear activation function. The forward- (red) and backward-propagating (orange) optical signals, whose amplitudes are proportional to the neuron activations, $a^{(l-1)}$, and errors, $\delta^{(l)}$, respectively, are tapped off by beam splitters, measured by heterodyne detection and multiplied to determine the weight matrix update in Eq. (2). This multiplication can also be implemented optically, as discussed in the text. The final update of the weights, as well as the preparation of network input, is implemented electronically. (b) Error calculation at the output layer performed optically or digitally, as described in the text.

Fig. 2. Saturable absorber response. (a) The transmission and (b) transmission derivative of an SA unit with optical depths of 1 (left) and 30 (right), as defined by Eqs. (4) and (6), respectively. Also shown in panel (b) are the actual probe transmissions given by Eq. (5), which approximate the derivatives, with and without the rescaling. The scaling factors are 1.2 (left) and 2.5 (right). In the amplitude region (i), the SA behaves as a linear absorber for weak input but then exhibits strong nonlinearity when the pump intensity approaches the saturation threshold. Region (ii) corresponds to strong saturation: the ground-state population is depleted, and the absorber is rendered transparent.

and a sufficiently strong pump renders the medium nearly transparent in the saturated region. A suitably weak probe, on the other hand, does not modify the transmissivity of the atomic media and hence experiences linear absorption with the absorption coefficient determined by the pump,
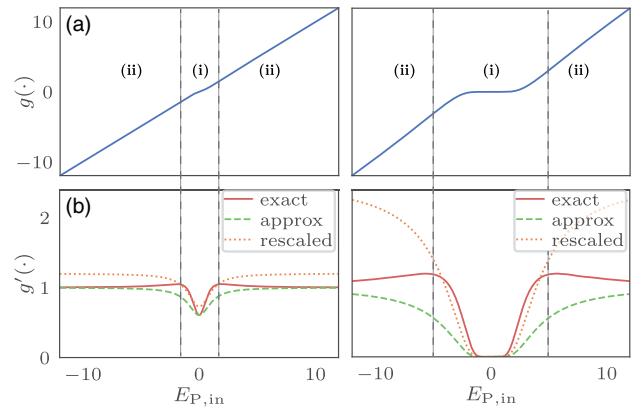
$$\frac{E_{\text{pr,out}}}{E_{\text{pr,in}}} = \exp\left(-\frac{\alpha_0/2}{1 + E_{P,\text{in}}^2}\right). \qquad (5)$$

Note that both beams are assumed to be resonant with the atomic transition and so, as the phase of the electric field is unchanged, we treat these as real-valued without loss of generality. Therefore, with the pump and probe taking the roles of the forward-propagating signal and backward-propagating error in an ONN, required features (i) and (ii) of our optical nonlinear unit are met.

Condition (iii), however, remains to be satisfied. The derivative of the pump transmission is

$$g'(E_{P,\text{in}}) = \left[1 + \frac{\alpha_0 E_{P,\text{in}}^2}{(1 + E_{P,\text{in}}^2)^2}\right] \exp\left(-\frac{\alpha_0/2}{1 + E_{P,\text{in}}^2}\right). \qquad (6)$$

The derivatives at α_0 of 1 and 30 are plotted in Fig. 2(b). Our key insight is that, in many instances, the square-bracketed factor in Eq. (6) can be considered constant, in which case the backpropagation transmission of Eq. (5) is a good approximation of the desired response in Eq. (6) up to a constant factor. Feature (iii) is then satisfied because a constant scaling of the

network gradients can be absorbed into the learning rate. This may appear as a coarse approximation, however; as we will see in the next section, it is only required to hold within the nonlinear region of the SA response, which is the case for our system [Fig. 2(b)].

The proposed scheme can be implemented on either integrated or free-space platforms. In the integrated setting, optical interference units that combine integrated phase-shifters and attenuators to realize intralayer weights have been demonstrated [9] as have, separately, on-chip SA through atomic vapor [32,33] and other nonlinear media [23,34]. A free-space implementation of the required matrix multiplication can be achieved using a spatial light modulator (SLM) [8,31] with the nonlinear unit provided by a standard atomic vapor cell. In the integrated case, an additional nontrivial step to map the weight gradients in Eq. (2) to suitable updates of the control parameters (i.e., phase-shifters and attenuator) is required; however, this challenge was recently addressed by Hughes *et al.* [20]. A free-space implementation, by contrast, has discrete blocks of SLM pixels directly controlling individual weights, so the update calculation is more straightforward.

Regardless of the chosen platform, passive optical elements can only implement weighted connections that satisfy conservation of energy. For networks with a single layer of nonlinear activations, this is not a practical limitation as the weight matrices can be effectively realized with normalized weights by correspondingly rescaling the neuron activations in the input layer. For deep networks with multiple layers, absorption through the vapor cell will reduce the field amplitude available to subsequent layers. This can be counteracted by interlayer amplification using, for example, semiconductor optical amplifiers [35].

In our proposed ONN, the only parts that require electronics are (a) real-valued homo- or heterodyne measurements of

the tapped-off neuron activations ($a^{(l)}$) and error terms ($\delta^{(l)}$) at each layer, (b) generating the network input and reference beams, and (c) updating the weights. In practice, the update (c) is calculated not for each individual training set element but as the average over multiple elements (a "mini-batch" or a training epoch); hence, the speed of this operation is not critical for the ONN performance. Generating the inputs, (b), and targets is decoupled from the calculation performed by the ONN and requires fast optical modulators, which are abundant on the market.

Finally, the measurements, (a), must be followed by calculating the product $\delta_j^{(l)} a_i^{(l-1)}$ and averaging over the minibatch. This operation can be implemented using electronic gate arrays. For a network with L layers of N neurons, this requires $2LN$ measurements and LN^2 offline multiplications. Alternatively, the multiplication can be realized by direct optical interference of the two signals with each other, followed by intensity measurement. The optical multiplication would require phase stability of the setup and the additional overhead of $2LN^2$ photodetectors but eliminate the need for reference beams and offline multiplications. Additionally, calculating the new weight matrices from the data acquired through these detectors will require $\sim LN^2$ operations, which would need to be performed once per epoch. Note that these operations may introduce a performance bottleneck due to the limited refresh rate of the modulators.

Although the activation and error signals are frequency-degenerate in our scheme, their counterpropagating configuration makes them easily distinguishable at the detection stage. Furthermore, the nonlinear unit operation is not affected by the relative phase of the two counterpropagating signals.

The primary latencies associated with the optical propagation of the signal in the ONN are due to the bandwidths of the SAs and intralayer amplifiers. Further processing speed limitations are present in the photodetection and multiplication of $\delta_j^{(l)} a_i^{(l-1)}$ as well as conversion of the computed weight matrix gradients to their actuators within the ONN [20]. This latter conversion, however, occurs once per training batch, so this limitation can be amortized by using large batches.

The remaining, not yet discussed, element of the ONN training is the calculation and reinjection of the error $\delta^{(L)}$ at the output layer, to initiate backpropagation. To implement this optically, we train the ONN with the mean-squared-error loss function,

$$\mathcal{L} = \sum_i \frac{1}{2}(z_i^{(L)} - t_i)^2, \tag{7}$$

where t_i is the target value for the ith output neuron. This loss function implies $\delta_i^{(L)} = \partial\mathcal{L}/\partial z_i^{(L)} = z_i^{(L)} - t_i$, which is calculable by interference of the network outputs with the target outputs on a balanced beam-splitter. This approach to error calculation is illustrated in the right panel of Fig. 1(b), whereas the left panel shows the standard approach in which the errors are calculated offline (electronically).

3. EXAMINING APPROXIMATION ERRORS

To investigate our proposed backpropagation scheme and, in particular, how our approximated derivatives affect network performance, we consider the canonical ML task of image classification. Our first set of numerical experiments classifies images of handwritten digits from 0 to 9. We use the MNIST [36] data set, which contains grey-scale bitmaps of size 28 × 28 fed into the input layer of the ONN. The output layer contains 10 neurons whose target values are 0 or 1 dependent on the digit encoded in the bitmap ("one-hot encoding"). In this section, we use a network architecture with a single 128-neuron hidden layer, as shown in Fig. 3(a). Further details of the networks, training, and calculation of the accuracy metric for all experiments presented in this work can be found in Appendix A.

Initially, we consider the activation function to be provided by SA with an optical depth of $\alpha_0 = 10$. For the chosen network architecture, this provides 97.3% ± 0.1% classification accuracy after training, with no difference in performance regardless of whether the true derivatives in Eq. (6) or the optically obtainable derivative approximations in Eq. (5) are used. From Fig. 3(b), we can see that, during training, the neurons are primarily distributed in the unsaturated region of the SA activation function. This is a consequence of the fact that the expressive capacity of neural networks arises from the nonlinearity of its neurons. Therefore, to train the network, the optically obtained derivatives need to approximate the exact derivatives (up to a fixed scaling as previously discussed) in only this nonlinear region.
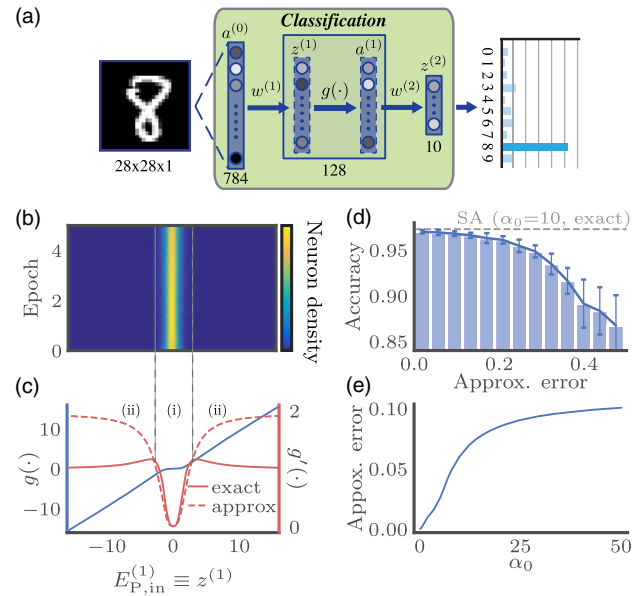
Fig. 3. Effects of imperfect approximation of the activation function derivative. (a) Feed-forward neural network architecture using a single hidden layer of 128 neurons. (b) Distribution of neuron inputs ($E_{P,\text{in}}^{(1)} \equiv z^{(1)}$), which is concentrated in the unsaturated region (1) of the SA activation function, $g(\cdot)$. As a result, the approximation error in the linear region (2) is less impactful on the training. (c) The transmission of an SA unit with $\alpha_0 = 10$, along with the exact and (rescaled for easier comparison) optically approximated transmission derivatives. (d) Performance loss associated with approximating activation function derivatives $g'(\cdot)$ with random functions, plotted as a function of the approximation error, for $\alpha_0 = 10$ (see Appendix B for details). (e) Average error of the derivative approximation in Eq. (5) as a function of the optical depth of an SA nonlinearity.

It is interesting to investigate how training is affected by imprecision in the derivatives used. To this end, we evaluate the network performance by replacing the derivative $g'(\cdot)$ with random functions of varying similarity to the true derivative within the nonlinear region (the quantitative measure, S, of the similarity is defined in Appendix B). From Fig. 3(d), we see that the performance appears robust to approximation errors, defined as $1-S$, of up to ~15%. We explain this potentially surprising observation by noting that gradient descent will converge even if the update vector deviates from the direction toward the exact minimum of the loss function, so long as this deviation is not too significant.

In the case of SA, i.e., when the approximate derivatives given by Eq. (5) are used, this error saturates at ~10% for increasing optical depth [see Fig. 3(e)] so no significant detrimental effect on the training accuracy can be expected. These results suggest that our scheme would still be effective in a noisy experimental setting, as further discussed in Appendix D, and that the approach studied here may function well for a broad range of optical nonlinearities.
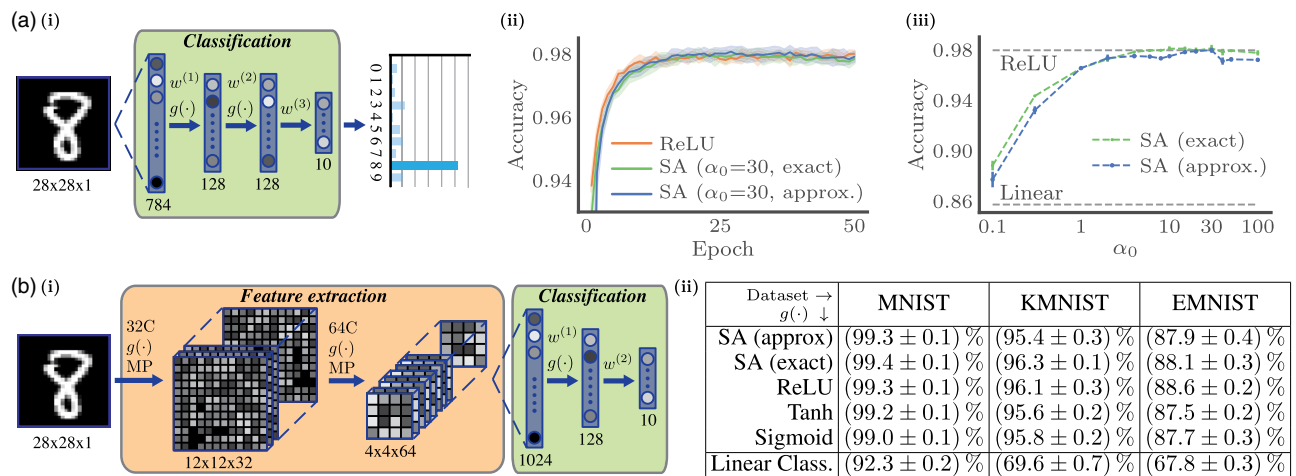
4. CASE STUDY: IMAGE CLASSIFICATION

Thus far, we have only used a simple network architecture to examine our derivative approximation; however, we now consider how ONNs with SA nonlinearities are compared with state-of-the-art ANNs. To do this, we use deeper network architectures for a range of image classification tasks. To obtain a comparison benchmark, we computationally train ANNs with equivalent architectures using standard best practices. Concretely, for ANNs we use ReLU (rectified linear unit) activation functions, defined as $g_{\mathrm{ReLU}}(z) = \max(0, z)$, and the categorical cross-entropy loss function, which is defined as

$\mathscr{L} = -\sum_i t_i \log(p_i)$, where $p_i = \exp(z_i^{(L)}) / \sum_k \exp(z_k^{(L)})$ is the softmax probability distribution of the network output (see Appendix A for a discussion of the different choices of loss function for ANNs and ONNs).

To begin, we use a network with two 128-neuron hidden layers, as shown in Fig. 4(a)(i), and once again consider the MNIST data set. Figure 4(a)(ii) compares the simulated performance of the optical and benchmark networks. The ReLU-based classifier achieves an accuracy of 98.0% ± 0.2%, which provides an approximate upper bound on the achievable performance of this network architecture for the chosen task [37]. An optical network with an optical depth of $\alpha_0 = 30$ [trained with approximate activation function derivatives in Eq. (5) obtained in optical backpropagation through the SA] exactly matches this level of performance with 98.0% ± 0.2% classification accuracy. As an additional benchmark, we train the optical network using the exact derivative in Eq. (6) of the activation function, obtaining a similar accuracy of 98.1% ± 0.3%. The convergence speed to near-optimum performance during training is unchanged across all of these networks.

Figure 4(a)(iii) shows the trained performance of optical networks as a function of the optical depth, which essentially determines the degree of nonlinearity of the transmission function. As $\alpha_0 \to 0$, our network can only learn linear functions of the input, which restricts the classification accuracy to 85.7% ± 0.4%. For larger optical depths, the performance of the network improves, with the strong performance observed at $\alpha_0 = 1$ increasing to near optimal levels once $\alpha_0 \geq 10$, which is readily obtainable experimentally. Eventually, for $\alpha_0 \geq 30$, we start to see the performance of the approximated derivatives reduced, although high accuracy is still obtained. This can be attributed to the increasing approximation errors associated with high optical depths [see Fig. 3(e)], which, as

Fig. 4. Performance on image classification. (a) (i) The fully connected network architecture. (ii) Learning curves for the SA [with either exact derivatives in Eq. (6) of the activation function or their approximation in Eq. (5)] and benchmark ReLU networks. (iii) The final classification accuracy achieved as a function of the optical depth, α_0, of the SA cell. (b) (i) The convolutional network architecture. Sequential convolution layers of 32 and 64 channels convert a 28 × 28 pixel image into a 1024-dimensional feature vector, which is then classified (into $N_C = 10$ classes for MNIST and KMNIST, and $N_C = 47$ classes for EMNIST) by fully connected layers. Pooling layers are not shown for simplicity. (ii) Classification accuracy of convolutional networks when using various activation functions. The same deep network architecture is applied to all data sets, but the SA networks use mean-pooling, while the benchmark networks use max-pooling. The last row shows the performance of a simple linear classifier as a baseline.

Dataset → $g(\cdot)$ ↓	MNIST	KMNIST	EMNIST
SA (approx)	(99.3 ± 0.1) %	(95.4 ± 0.3) %	(87.9 ± 0.4) %
SA (exact)	(99.4 ± 0.1) %	(96.3 ± 0.1) %	(88.1 ± 0.3) %
ReLU	(99.3 ± 0.1) %	(96.1 ± 0.3) %	(88.6 ± 0.2) %
Tanh	(99.2 ± 0.1) %	(95.6 ± 0.2) %	(87.5 ± 0.2) %
Sigmoid	(99.0 ± 0.1) %	(95.8 ± 0.2) %	(87.7 ± 0.3) %
Linear Class.	(92.3 ± 0.2) %	(69.6 ± 0.7) %	(67.8 ± 0.3) %

previously discussed, accumulate in the deeper network architecture. In free-space implementations with saturated atomic vapor cells, the optical depth can be dynamically tuned via cell temperature. For semiconductor absorbers in integrated settings, the optical depth is related to the material thickness and/or the density of dopants.

To probe the limits of the achievable performance using SA nonlinearities and optical backpropagation, we also consider the more challenging Kuzushiji-MNIST [38] (KMNIST) and extended-MNIST [39] (EMNIST) data sets. For these applications, we use a deep network architecture with convolutional layers (see Appendix A for details), as illustrated in Fig. 4(b)(i), which significantly increases the achievable classification accuracy to a level approaching the state-of-the-art. While not the focus of this work, we emphasize that convolutional operations are readily achievable with optics. Current research into convolutional ONNs either directly leverages imaging systems [40] or decomposes the required convolution into optical matrix multiplication [41–43].

In addition to convolutional layers, convolutional neural networks also contain pooling layers, which locally aggregate neuron activations. The common implementation of these is max-pooling; however, this operation does not readily translate to an optical setting. Therefore, for ONNs, we deploy mean-pooling, where the activation of neurons is locally averaged, which is a straightforward linear optical operation. In contrast, our benchmark ANNs utilize max-pooling.

Figure 4(b)(ii) compares the obtained performance with SA nonlinearities (with $\alpha_0 = 10$) to that achieved with benchmark ANNs that use various standard activation functions. We see an equivalent level of performance, despite the approximation in the backpropagation phase. This result suggests that all-optical backpropagation can be utilized to train sophisticated networks to state-of-the-art levels of performance.

5. DISCUSSION

This work presents an effective and surprisingly simple approach to achieving optical backpropagation through nonlinear units in a neural network, an outstanding challenge in the pursuit of truly all-optical networks. With our scheme, the information propagates through the network in both directions without interconversion between optical and electronic form. The role of digital electronics is reduced to the preparation of the network input, photodetection, and updating the network parameters. In these elements of the network, the conversion speed is not critical, particularly for large batches of training data. A detailed estimate of the energy efficiency and computation speed of the optically trained neural network is presented in Appendix C.

As compared with offline training, optical training is more robust against experimental imperfections, since such imperfections are automatically included and counteracted during the training process. As an illustration, numerical simulation results of optical training in noisy experimental settings are presented in Appendix D.

The scheme is compatible with a variety of ONN platforms. We also anticipate that a broader class of nonlinear optical phenomena can be used to implement the activation function.

For example, one could consider directly using saturation of intralayer amplifiers for this role, circumventing the need for SA units entirely. A preliminary numerical experiment to this effect is discussed in Appendix E. Our scheme may also be applicable to the optical Kerr nonlinearity as proposed in Ref. [30] for training diffractive neural networks, albeit with the added complexity of operating with complex-valued field amplitudes and weights. Therefore, as well as presenting a path toward the end-to-end optical training of neural networks, this work sets out an important consideration for nonlinearities in the design of analog neural networks of any nature.

APPENDIX A. NETWORK DETAILS

1. Image Data Sets

We consider three different data sets, all containing 28×28 pixel grey-scale images: MNIST [36], Kuzushiji-MNIST (KMNIST) [38], and extended-MNIST (EMNIST) [39]. MNIST corresponds to handwritten digits from 0 to 9; KMNIST contains 10 classes of handwritten Japanese cursive characters; and we use the EMNIST balanced data set, which contains 47 classes of handwritten digits and letters. MNIST and KMNIST have 70,000 images in total, split into 60,000 training and 10,000 test instances. EMNIST has 131,600 images, with 112,800 (18,800) training (test) instances. For all data sets, the training and testing sets have all classes equally represented.

2. Network Architectures

The fully connected network we train to classify MNIST [corresponding to the results in Fig. 4(a)] first unrolls each image into a 784-dimensional input vector, before two 128-neuron hidden layers and a 10-neuron output layer.

The convolutional network depicted in Fig. 4(b)(i) has two convolutional layers of 32 and 64 channels, respectively. Each layer convolves the input with 5×5 filters (with a stride of 1 and no padding), followed by a nonlinear activation function and finally a pooling operation (with both kernel size and stride of 2). After the convolutional network, classification is carried out by a fully connected network with a single 128-neuron hidden layer and N_C neuron output layer, where N_C is the number of classes in the target data set.

Multilayer ONNs are assumed to have the same optical depth of their saturable absorbers in all layers.

3. Network Loss Function

As stated in the main text, we train ONNs using the mean-squared-error (MSE) loss function, whereas the ANN baselines use categorical cross-entropy (CCE). This choice was made as the gradients of MSE loss are readily calculable in an optical setting, whereas the softmax operation in CCE would require offline calculation. However, our ANNs use CCE, as this is the standard choice for classification problems in the deep learning community. For completeness, we retrained our ANN baselines for MNIST classification using MSE. The fully connected classifier [Fig. 4(a)(i)] provided a classification accuracy of $98.0\% \pm 0.2\%$, while the convolutional classifier [Fig. 4(a)(ii)], using ReLU nonlinearities, scored $99.5\% \pm 0.1\%$. In both cases, the performance of MSE is essentially equivalent to that of CCE.

4. Network Training

All networks are trained with a minibatch size of 64. We used the Adam optimizer with a learning rate of 5×10^{-4}, independent of the optical depth of the SA. For each network, the test images of the target data set are split evenly into a "validation" set and a "test" set. After every epoch, the performance of the network is evaluated on the held-out "validation" images. The best ONN parameters found over training are then used to verify the performance on the "test" set. Therefore, learning curves showing the performance during training [i.e., Fig. 4 (a)(ii)] are plotted with respect to the "validation" set, with all other reported results corresponding to the "test" set. The fully connected networks were trained on MNIST for 50 epochs. The convolutional networks are trained for 20 epochs when using ReLU, Tanh, or Sigmoid nonlinearities and 40 epochs when using SA nonlinearities.

Training performance is empirically observed to be sensitive to the initialization of the weights, which we ascribe to the small derivatives away from the nonlinear region of the SA response curve. For low optical depths, $\alpha_0 \leq 30$, all layers are initialized as a normal distribution of width 0.1 centered around 0. For higher optical depths, the weights of the fully connected ONN shown in Fig. 4(a) are initialized to a double-peaked distribution comprised of two normal distributions of width 0.15 centered at ± 0.15. We do not constrain our weight matrices during training because, as discussed in the main text, conservation of energy can always be satisfied by rescaling the input power or output threshold for the first and last linear transformation and using intralayer amplifiers in deeper architectures.

For all images, the input is rescaled to be between 0 and 1 (which practically would correspond to $0 \leq E_{P,\text{in}}^{(0)} \leq 1$) when passed to a network with computational nonlinearities (i.e., ReLU, Sigmoid, or Tanh). Due to "absorption" in networks with SA nonlinearities, we empirically observe that rescaling the input data to higher values results in faster convergence when training convolutional networks with multiple hidden layers. Therefore, the fully connected networks in Fig. 4(a) use inputs between 0 and 1, and the convolutional networks in Fig. 4(b) use inputs normalized between 0 and 5 (15) for $\alpha_0 \leq 10$ ($\alpha_0 > 10$).

APPENDIX B. CALCULATION OF THE DERIVATIVE APPROXIMATION ERROR

As discussed in the main text, we approximate the true derivatives $g'(\cdot)$ of the activation functions by random functions $f(\cdot)$ to test the effect of the approximation error on training. Here, we discuss how these functions are generated and how the similarity measure is defined.

The response of a saturable absorption nonlinearity can be considered in two regimes, i.e., nonlinear (unsaturated) and linear (saturated), which are labeled (i) and (ii) in Fig. 2, respectively. During the network training, the neuron input values $(z_j^{(l)})$ are primarily distributed in the nonlinear region, as seen in Fig. 3(b) and discussed in the main text. Therefore, we model the neuron input as a Gaussian distribution within this region:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{z^2}{2\sigma^2}\right), \tag{B1}$$

where 2σ is the width of region (1). We then define the similarity as the reweighted normalized scalar product between the accurate and approximate derivatives,

$$S = \frac{|\int f(z)g'(z)p(z)\mathrm{d}z|^2}{\int [f(z)]^2 p(z)\mathrm{d}z \cdot \int [g'(z)]^2 p(z)\mathrm{d}z}. \tag{B2}$$

According to the Cauchy–Schwarz inequality, S is bounded by 1 and therefore so is the average approximation error, $1-S$.

To obtain the results in Figs. 3(d) and 3(e), we generate 200 random functions for f, with different approximation errors. We first generate an array of pseudo-random numbers ranging from 0 to 1, concatenate it with the flipped array to make them symmetric like the derivative $g'(\cdot)$, and then use shape-preserving interpolation to obtain a smooth and symmetric random function. The network is then trained once with each of the generated f.

APPENDIX C. OPTICAL POWER CONSUMPTION AND COMPUTATION SPEED

The optical power consumption in an ONN depends on the network architecture and implementation details. For concreteness, we consider a fully connected network with $N = 1000$ units per layer, with SA optical nonlinearities implemented on the ^{87}Rb D_2 line. Recalling Fig. 3(b) from the main text, we note that, during training, the input power to each neuron is typically restricted to the unsaturated region, (i), of the nonlinearity response. For the SA nonlinearities we consider, the saturation intensity is given by [44]

$$I_{\text{sat}} = \frac{\hbar\omega\Gamma}{2\sigma_0} = 16.6 \ \mu\text{W}\cdot\text{mm}^{-2}, \tag{C1}$$

where $\Gamma = 2\pi \times 6$ MHz is the natural linewidth, and $\sigma_0 = 3\lambda^2/(2\pi)$ is the resonant absorption cross section. For beams with a waist of $w_0 = 100 \ \mu$m, this corresponds to a saturation power of $P_{\text{sat}} \approx 500$ nW per neuron and total SA input power for 1000 units on the order of 500 μW.

To saturate the SA, the optical pulse needs to be longer than the excited state lifetime $\Gamma^{-1} = 26$ ns. The energy cost of a single forward pass through the network is then ~ 10 pJ, and the backpropagation energy cost is negligible. Since a single interlayer transition involves a VMM with N^2 multiplications, one can estimate the energy cost per multiply-accumulate operation to be ~ 10 aJ. In an integrated setting, the saturation powers are higher, but the pulse durations are proportionally shorter, so similar energy costs can be expected. This is, of course, an idealized estimate, which does not include peripheral energy costs in powering and sustaining the instruments and stabilizing the system. Hence, the actual power consumption can be expected to be at least an order of magnitude higher. For comparison, today's electronic processors like CPUs and GPUs have energy costs on a scale of 0.1 to 1 nJ per operation.

The computation speed of the ONN is determined by the response time of the SA units and amplifiers as well as the speed of optical modulators. The response time of atomic-based SA is tens of nanoseconds and that of semiconductor SA is on the order of picoseconds. Response time of optical amplifiers is

on a similar time scale. The speed of the optical modulator preparing the network input is more likely to be the bottleneck in the near future. The bandwidths of SLM, thermo-optic modulator, and electro-optic modulator are on the order of 10 kHz [31], 100 kHz [9], and 10 GHz [45], respectively. Adoption of an ultrafast electro-optic modulator in an ONN with 1000 neurons per layer would perform 10^{16} operations per second. Assuming the energy consumption of 1 mW as estimated above, the energy efficiency would be 10^{19} operations per second per watt. For comparison, a higher-end consumer GPU has a computation speed of $\sim 10^{12}$ operations per second, with the energy efficiency of $\sim 10^{10}$ operations per second per watt, and neuromorphic electronics has an energy efficiency below 10^{14} operations per second per watt [13].

APPENDIX D. OPTICAL TRAINING WITH EXPERIMENTAL IMPERFECTIONS

A practical ONN will exhibit noise and errors arising from background scattering, nonideal optical multiplication or interference, and digitization error of electronic signals. Here, we investigate the robustness of our scheme against these imperfections. We adopt the network structure of Fig. 3 and perform simulation on the MNIST data set.

In the first series of tests, we add a certain amount of random Gaussian noise to the activation outputs $a_i^{(l)}$ and error fields $\delta_i^{(l)}$. We define the noise level as the ratio between the standard deviation of noise and signal: $\sigma_{\text{noise}}/\sigma_{\text{signal}}$. The top three rows in Table 1 show the training result. The classification accuracy decreases mildly from 97.3% to 95.8% with 10% noise.

Our second series of tests consists in randomly scaling the activation output and error beams: each mode has a fixed scaling factor, and all the scaling factors are sampled from a Gaussian distribution $N(1, \sigma)$. This models the nonuniform losses of different spatial modes. From Table 1 (fourth to sixth rows), we see that such imperfection causes no performance degradation because the weights are automatically rescaled during training to counteract such deviation.

We further consider the digitization error of weights, activation, and error beams, since they are usually electronically controlled or read out with limited-bitwidth analog-to-digital or digital-to-analog converters. The seventh to ninth rows of Table 1 show that the training is sensitive to the bitwidth limitations, and the accuracy drops to 96.0% with 8-bit precision and 93.5% with 6-bit precision. Therefore, in an actual system, one should use at least 8-bit controls to preserve high accuracy

of the network, which can be readily achieved. Note that high-performance ONNs with bitwidths as low as 2 to 4 bits have been proposed [46].

Finally, the last row of Table 1 shows that classification accuracy as high as 96.1% can still be achieved for a practical 8-bit system with the combined effect of 5% random noise and up to 10% deviation.
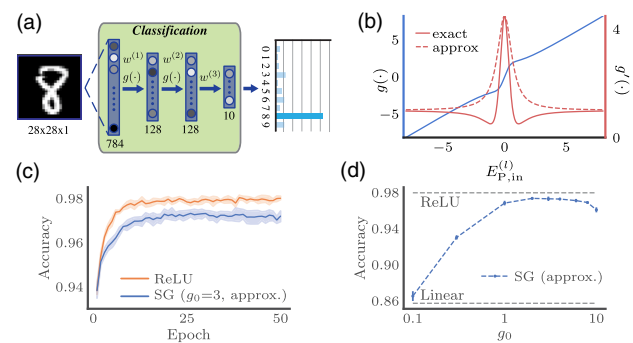
APPENDIX E. OPTICAL BACKPROPAGATION WITH SATURABLE GAIN

In optical amplifiers, saturable gain (SG) takes place when a sufficiently high input power depletes the excited state of the gain medium. In a two-level system, this process can be described similarly to saturable absorption by simply replacing the optical depth term $-\alpha_0$ in Eq. (4) with a positive gain factor g_0. The transmission in the forward direction is plotted and compared with both the theoretically exact and optically obtained transmission derivatives in Fig. 5(b) with $g_0 = 3$. The derivative curves have the inverted shapes of the SA derivative curves.

The optically obtained derivatives also appear to be a reasonable approximation of the exact gradient. To examine this, we replace the SA nonlinearity with SG nonlinearity in the fully connected network, as shown in Fig. 4(a), and repeat the optical training simulation. The MNIST image classification performance is shown in Figs. 5(c) and 5(d). High accuracy can be achieved with a gain factor as small as 1, and the best result scores 97.3% ± 0.1% at $g_0 = 3$, slightly lower than that of the benchmark ReLU network and SA-based ONN. Since the derivative approximation error of the SG nonlinearity is the same as that of the SA nonlinearity, the performance degradation is mainly attributed to the nonlinearity itself; however, higher performance may be achievable through careful hyperparameter tuning.

APPENDIX F. CODE AVAILABILITY

Source code is available in Code 1, Ref. [47].

Fig. 5. Optical backpropagation through saturable gain (SG) nonlinearity. (a) Fully connected network architecture, which is the same as Fig. 4(a) except for the nonlinearity. (b) Transmission and transmission derivatives of the SG unit with gain factor $g_0 = 3$. (c) Learning curves for the SG-based ONN and benchmark ReLU networks. (d) The final classification accuracy achieved as a function of the gain.

Table 1. ONN Training with Experimental Imperfections

Noise Level	Deviation Level	Bitwidth	Accuracy
0%	0%	32	97.3% ± 0.1%
5%	0%	32	96.7% ± 0.2%
10%	0%	32	95.8% ± 0.3%
0%	5%	32	97.1% ± 0.3%
0%	10%	32	97.1% ± 0.1%
0%	20%	32	97.1% ± 0.3%
0%	0%	8	96.0% ± 0.4%
0%	0%	6	93.5% ± 0.8%
5%	10%	8	96.1% ± 0.2%

†These authors contributed equally to this work.

REFERENCES

1. E. Cambria and B. White, "Jumping NLP curves: a review of natural language processing research," IEEE Comput. Intell. Mag. **9**, 48–57 (2014).
2. W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: a comprehensive review," Neural Comput. **29**, 2352–2449 (2017).
3. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, and S. Dieleman, "Mastering the game of go with deep neural networks and tree search," Nature **529**, 484–489 (2016).
4. J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proceedings of the 34th International Conference on Machine Learning* (2017), Vol. 70, pp. 1263–1272.
5. G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, "Neural-network quantum state tomography," Nat. Phys. **14**, 447–450 (2018).
6. K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Neural Netw. **2**, 359–366 (1989).
7. G. Cybenko, "Approximation by superpositions of a sigmoidal function," Math. Control Signals Syst. **2**, 303–314 (1989).
8. P. N. Tamura and J. C. Wyant, "Two-dimensional matrix multiplication using coherent optical techniques," Opt. Eng. **18**, 182198 (1979).
9. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljačić, "Deep learning with coherent nanophotonic circuits," Nat. Photonics **11**, 441–446 (2017).
10. J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. P. Pernice, "All-optical spiking neurosynaptic networks with self-learning capabilities," Nature **569**, 208–214 (2019).
11. L. De Marinis, M. Cococcioni, P. Castoldi, and N. Andriolli, "Photonic neural networks: a survey," IEEE Access **7**, 175827 (2019).
12. A. N. Tait, T. F. De Lima, E. Zhou, A. X. Wu, M. A. Nahmias, B. J. Shastri, and P. R. Prucnal, "Neuromorphic photonic networks using silicon photonic weight banks," Sci. Rep. **7**, 7430 (2017).
13. B. J. Shastri, A. N. Tait, T. Ferreira de Lima, M. A. Nahmias, H. T. Peng, and P. R. Prucnal, "Principles of neuromorphic photonics," in *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, ed. (Springer, 2018).
14. Y. Abu-Mostafa and D. Psaltis, "Optical neural computers," Sci. Am. **256**, 88–95 (1987).
15. S. Jutamulia and F. Yu, "Overview of hybrid optical neural networks," Opt. Laser Technol. **28**, 59–72 (1996).
16. J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, and D. Brunner, "Reinforcement learning in a large-scale photonic recurrent neural network," Optica **5**, 756–760 (2018).
17. X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "All-optical machine learning using diffractive deep neural networks," Science **361**, 1004–1008 (2018).
18. Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, "All-optical neural network with nonlinear activation functions," Optica **6**, 1132–1137 (2019).
19. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436–444 (2015).
20. T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, "Training of photonic neural networks through *in situ* backpropagation and gradient measurement," Optica **5**, 864–871 (2018).
21. I. A. D. Williamson, T. W. Hughes, M. Minkov, B. Bartlett, S. Pai, and S. Fan, "Reprogrammable electro-optic nonlinear activation functions for optical neural networks," IEEE J. Sel. Top. Quantum Electron. **26**, 7700412 (2020).
22. M. M. P. Fard, I. A. Williamson, M. Edwards, K. Liu, S. Pai, B. Bartlett, M. Minkov, T. W. Hughes, S. Fan, and T. A. Nguyen, "Experimental realization of arbitrary activation functions for optical neural networks," Opt. Express **28**, 12138–12148 (2020).
23. Z. Cheng, H. K. Tsang, X. Wang, K. Xu, and J.-B. Xu, "In-plane optical absorption and free carrier absorption in graphene-on-silicon waveguides," IEEE J. Sel. Top. Quantum Electron. **20**, 4400106 (2013).
24. A. Jha, C. Huang, and P. R. Prucnal, "Reconfigurable all-optical nonlinear activation functions for neuromorphic photonics," Opt. Lett. **45**, 4819–4822 (2020).
25. M. Miscuglio, A. Mehrabian, Z. Hu, S. I. Azzam, J. George, A. V. Kildishev, M. Pelton, and V. J. Sorger, "All-optical nonlinear activation function for photonic neural networks," Opt. Mater. Express **8**, 3851–3863 (2018).
26. K. Wagner and D. Psaltis, "Multilayer optical learning networks," Appl. Opt. **26**, 5061–5076 (1987).
27. D. Psaltis, D. Brady, and K. Wagner, "Adaptive optical networks using photorefractive crystals," Appl. Opt. **27**, 1752–1759 (1988).
28. A. A. Cruz-Cabrera, M. Yang, G. Cui, E. C. Behrman, J. E. Steck, and S. R. Skinner, "Reinforcement and backpropagation training for an optical neural network using self-lensing effects," IEEE Trans. Neural Netw. **11**, 1450–1457 (2000).
29. T. Zhou, X. Lin, J. Wu, Y. Chen, H. Xie, Y. Li, J. Fan, H. Wu, L. Fang, and Q. Dai, "Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit," arXiv:2008.11659 (2020).
30. T. Zhou, L. Fang, T. Yan, J. Wu, Y. Li, J. Fan, H. Wu, X. Lin, and Q. Dai, "*In situ* optical backpropagation training of diffractive optical neural networks," Photon. Res. **8**, 940–953 (2020).
31. J. Spall, X. Guo, T. D. Barrett, and A. I. Lvovsky, "Fully reconfigurable coherent optical vector-matrix multiplication," Opt. Lett. **45**, 5752–5755 (2020).
32. W. Yang, D. B. Conkey, B. Wu, D. Yin, A. R. Hawkins, and H. Schmidt, "Atomic spectroscopy on a chip," Nat. Photonics **1**, 331–335 (2007).
33. R. Ritter, N. Gruhler, W. Pernice, H. Kübler, T. Pfau, and R. Löw, "Atomic vapor spectroscopy in integrated photonic structures," Appl. Phys. Lett. **107**, 041101 (2015).
34. Q. Bao, H. Zhang, Y. Wang, Z. Ni, Y. Yan, Z. X. Shen, K. P. Loh, and D. Y. Tang, "Atomic-layer graphene as a saturable absorber for ultrafast pulsed lasers," Adv. Funct. Mater. **19**, 3077–3083 (2009).
35. M. J. Connelly, *Semiconductor Optical Amplifiers* (Springer, 2007).
36. Y. LeCun, C. Cortes, and C. Burges, and ATT Labs, "MNIST handwritten digit database," http://yann.lecun.com/exdb/mnist (2010).
37. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffer, "Gradient-based learning applied to document recognition," Proc. IEEE **86**, 2278–2324 (1998).
38. T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, "Deep learning for classical Japanese literature," arXiv:1812.01718 (2018).
39. G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "EMNIST: an extension of MNIST to handwritten letters," arXiv:1702.05373 (2017).
40. J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein, "Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification," Sci. Rep. **8**, 12324 (2018).
41. H. Bagherian, S. Skirlo, Y. Shen, H. Meng, V. Ceperic, and M. Soljačić, "On-chip optical convolutional neural networks," arXiv:1808.03303 (2018).
42. S. Xu, J. Wang, R. Wang, J. Chen, and W. Zou, "High-accuracy optical convolution unit architecture for convolutional neural networks by cascaded acousto-optical modulator arrays," Opt. Express **27**, 19778–19787 (2019).
43. R. Hamerly, L. Bernstein, A. Sludds, M. Soljačić, and D. Englund, "Large-scale optical neural networks based on photoelectric multiplication," Phys. Rev. X **9**, 021032 (2019).
44. D. A. Steck, "Rubidium 87 D line data," http://steck.us/alkalidata/rubidium87numbers.pdf.

45. C. Wang, M. Zhang, X. Chen, M. Bertrand, A. Shams-Ansari, S. Chandrasekhar, P. Winzer, and M. Lonoar, "Integrated lithium niobate electro-optic modulators operating at CMOS-compatible voltages," Nature **562**, 101–104 (2018).

46. M. Miscuglio and V. J. Sorger, "Photonic tensor cores for machine learning," Appl. Phys. Rev. **7**, 031404 (2020).

47. T. D. Barrett, "Tomdbar/all-optical-neural-networks: publication code (Version v1.0.0)," https://doi.org/10.5281/zenodo.4059672 (2020).