SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

February 2021, Vol. 64 122401:1–122401:12
https://doi.org/10.1007/s11432-020-2998-1

Optoelectronic convolutional neural networks based on time-stretch method

Yubin ZANG, Minghua CHEN, Sigang YANG & Hongwei CHEN*

Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing 100084, *China*

Received 14 May 2020/Revised 9 July 2020/Accepted 17 July 2020/Published online 20 January 2021

Abstract In this paper, a new architecture of optoelectronic convolutional neural networks (CNNs) based on time-stretch method is proposed. In this loop-shaped structure mainly composed of fiber optical and electronic devices, computations of data from each layer of CNN which are carried by light pulses with high repetition rate can be accomplished in a serial way. Therefore, a 5-layer CNN with two convolution layers, two mean pooling layers and one fully-connected layer are implemented. Under the test of handwriting digit recognition, its accuracy can reach up to 95% under ideal circumstances. Tests under different relative noise levels have been conducted and analyzed as well.

Keywords convolutional neural networks, time-stretch method, artificial intelligence

Citation Zang Y B, Chen M H, Yang S G, et al. Optoelectronic convolutional neural networks based on time-stretch method. Sci China Inf Sci, 2021, 64(2): 122401, https://doi.org/10.1007/s11432-020-2998-1

1 Introduction

Convolutional neural networks (CNNs) have been widely adopted in fields of computer vision [1], speech recognition [2], and autonomous vehicles [3–5]. By adopting convolution layers and pooling layers as their core layers instead of only fully-connected layers like traditional feed-forward neural networks, convolutional neural networks have advantages in pattern extraction and generalization. Under most circumstances, they have better recognition and prediction performances compared with traditional feed-forward neural networks [6].

Electronic integrated chips (ICs) are the most widely used platform for implementing convolutional neural networks. Although convolutional neural networks perform better in machine learning tasks compared with feed-forward neural networks, more basic universal linear computing units are required in order to accomplish hundreds or even thousands of convolution operations between each layer. Transistors are manufactured smaller and smaller to pursue not only higher computing speed but also lower energy cost. However, problems will become thorny sooner or later since today's scale of transistors is so tiny that Moore's law will be ineffective in the near future [7]. It means that researchers cannot improve performances of the chip by rigidly shrinking the scale of transistors like they have always done before. Although convolutional neural networks which adopt the technique of parameter reuse in convolution and pooling layers can reduce memory requirements for storage to some extend compared with feed-forward neural networks with the same scale, it cannot completely address the data exchange problem which will lower both power efficiency and computing speed.

While researchers have paid much their attention on further improving performances of electronic implemented CNNs, the technology of optoelectronics may provide another scheme to breakthrough today's bottleneck of implementing CNNs and other photonic based AI structures with higher computing speed and lower energy consumption [8–16]. Optical devices are not able to record data, so electronic counterparts are adopted to store feedback results and better control the whole system. Therefore, both higher computing speed and lower energy cost can be obtained once adopting the up-to-date technology

* Corresponding author (email: chenhw@tsinghua.edu.cn)

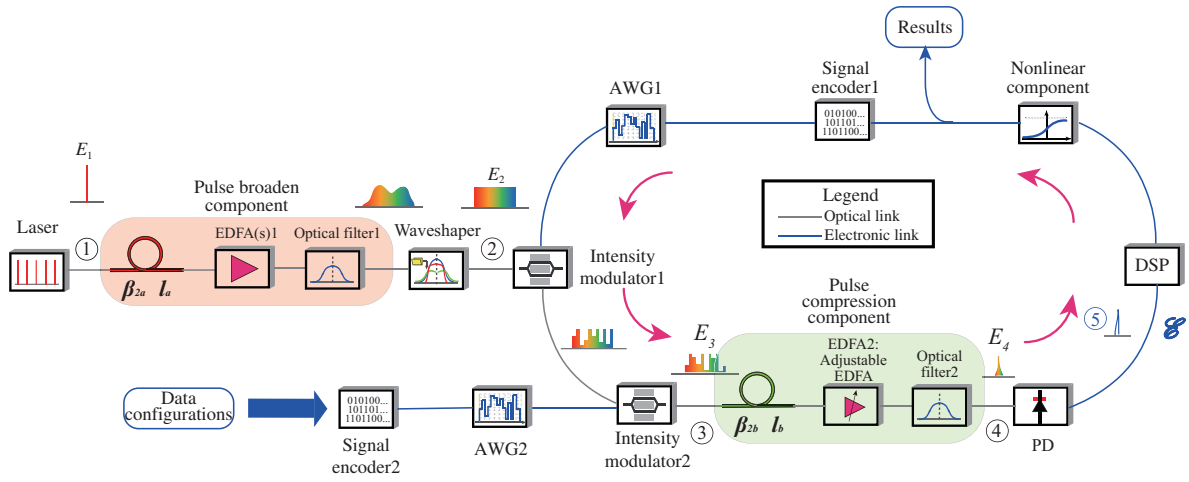

Figure 1 (Color online) Structure of TS-CNN system. Light pulses propagate through the loop-shaped structure to implement linear computations in CNN. Index in circles represents important nodes whose signals will be shown in Subsection 3.2. Note that EDFA(s)1 represents one kind of untunable EDFA(s).

of optoelectronics. Time-stretch method [17], which is often used in fields of signal processing [18–21] and imaging [22–25], provides a way of manipulating pulses in frequency-time plane. These pulses which have been broadened by adopting time-stretch method can be served as data carriers for serial vector computations once they were modulated with encoded data. Vector multiplication results can later be obtained by compressing and accumulating the modulated pulses through dispersion fiber and photodetector (PD) based on the theory of time-stretch [26]. Combined with electronic-assisted signal encoding and logic controlling, time-stretch method has potential applications in implementing convolutional neural networks.

In this paper, we propose an optoelectronic structure based on time-stretch method (TS-CNN) to implement a 5-layer convolutional neural networks. With two convolution layers, two mean pooling layers and one fully-connected layer, not only can the whole system manage the original digit handwriting task without the need to conduct picture compression operations, but also let recognition accuracy reach up to 95%. This serial-computing structure differing from other serial-computing proposals by fully utilizing both time and frequency information of pulses, once adapted, can not only greatly improve the dimensions of optoelectronic neural networks, but also combine data computation and tranmission as a whole. The rest of this paper is developed into three parts. The analysis and proof that the structure of TS-CNN can implement convolutional neural networks are written in Section 2 while all simulations and results are shown in Section 3. Other discussions and future work will be developed in conclusion part.

2 Structure and analysis

2.1 Structure of TS-CNN

As is described in Figure 1, TS-CNN is a loop-based system. Each loop implements one basic vector multiplication of CNN. Two intensity modulators, one pulse compression component, one PD, one digital signal processor (DSP), one nonlinear component, the signal encoder1 and the arbitrary waveform generator1 (AWG1) are in the loop. Optical pulses generated by a mode-locked laser enter the loop and propagate through the optical fiber. After pulses are broadened and flattened by the pulse broaden component and the waveshaper respectively, intensity modulator1 modulates the encoded results of the previous layer from signal encoder1 and intensity modulator2 modulates encoded data configurations of the current layer from signal encoder2 onto the peaks of broadened pulses respectively. Then, the pulse compression component containing fiber with the opposite dispersion coefficient is used to compress the modulated broadened pulses. A PD whose bandwidth is narrower than that of the signals is used to convert optical signals into electronic signals and accumulate the pulse energy. DSP is used to calibrate, store, process the pulse energy obtained from optoelectronic structures and regroup them into convolution results. The nonlinear component is used as the final stage to turn linear convolution results into the feature maps as the input data for the next loop. After all pulses modulated with effective data flow over

paths of all loops, results of the TS-CNN system will come out after nonlinear component in the final loop.

2.2 Theoretical analysis

Since basic computations for convolution and mean pooling layers are convolution operations while basic computations for fully-connected layers are matrix multiplications and accumulations (MACs), proof will be focused on how our system implements these two computation operations. Theoretical analysis will be divided into two parts. In Subsection 2.2.1, signal analysis of pulse evolution and computational information transmission in single loop proves that vector multiplications can be implemented. Then by vectorize matrices, our system can implement MACs in time-division ways. In Subsection 2.2.2, analysis will focus more on designing the signals for modulation and time-division logic to prove that the whole TS-CNN system can implement convolution operations as well.

2.2.1 *Signal analysis and MAC implementation*

As is described in Figure 1, pulse whose temporal waveform noted as E_1 is firstly broadened and flattened by pulse broaden component and waveshaper to be E_2. The waveform of the pulse after the waveshaper is ought to be a rectangular shape whose width equals the absolute value of the multiplication of the spectrum bandwidth of the pulse Ω, second order propagation coefficient β_{2a} and the length of optical fiber in pulse broaden component l_a due to fiber optics and dispersion fourier transform [27] in the theory of time-stretch method. This can be described as

$$|E_2(l_a, T)| \propto \mathrm{rect}_{|\Omega\beta_{2a}l_a|}(T)\mathrm{e}^{-\frac{\alpha_a l_a}{2}}, \tag{1}$$

in which α_a means loss per length of fiber in pulse broaden component and rect$(\cdot)$ represents rectangular shaped function whose length is described in its subscript. T is the time-retarded frame which moves at the same speed as the light pulse propagates for the simplicity of derivations.

Then, the broadened and flattened pulse will be modulated with the signal generated by two intensity modulators. According to theory of linear systems, this process can be written as

$$E_3(l_a, T) = s(T)E_2(l_a, T), \tag{2}$$

in which $s(\cdot)$ represents the signal for modulation and E_3 describes temporal waveform of light pulse after modulation processes.

The signal $s(T)$ can be seen as the multiplication results of the signal from AWG1 and AWG2. If the signal of AWG1 comes from one vector $\boldsymbol{a}$ and the signal of AWG2 comes from the other vector $\boldsymbol{b}$, the signal for modulation can be further described as

$$s(T) = \sum_{k=1}^{K} \sqrt{a_k b_k} \times \mathrm{rect}_{|\Omega\beta_{2a}l_a/K|}\left(T - (2k - K - 1)\frac{|\Omega\beta_{2a}l_a|}{2K}\right), \tag{3}$$

in which a_k and b_k is the kth element of $\boldsymbol{a}$ and $\boldsymbol{b}$ both containing K elements.

In the pulse compression component, pulses are compressed through dispersion fiber with the second order propagation coefficient β_{2b} (whose symbol is the opposite of β_{2a}) and length l_b. This process can be described as

$$E_4\left(l_a + l_b, T\right) = \frac{\mathrm{e}^{-\frac{\alpha_b l_b}{2}}}{2\pi}\int_{-\infty}^{\infty}\left[S_3\left(l_a, \omega - \omega_s\right)\exp\left(\mathrm{j}\frac{\beta_{2b}l_b}{2}\left(\omega - \omega_s\right)^2\right) \times \exp\left(-\mathrm{j}\left(\omega - \omega_s\right)T\right)\right]\mathrm{d}\omega, \tag{4}$$

in which E_4 represents the temporal waveform of the pulse after pulse compression component. S_3 represents the spectrum of modulated pulse. α_b means loss per length of the fiber in pulse compression component. ω and ω_s represents angular frequency and central angular frequency of the pulse.

The resulting signal after the PD in each loop can be calculated based on Plancherel theorem [28]. This is because the relationship between the energy of modulated pulse and compressed pulse can be clearly found via frequency domain so as to calculate the final result. Therefore, this process can be described precisely as

$$\mathscr{E} = \kappa\int_{-\infty}^{\infty}|E_4|^2\mathrm{d}T = \frac{\kappa}{2\pi}\int_{-\infty}^{\infty}S_4S_4^*\mathrm{d}\omega = \frac{\kappa\mathrm{e}^{-\alpha_b l_b}}{2\pi}\int_{-\infty}^{\infty}S_3S_3^*\mathrm{d}\omega$$

$$= \kappa \mathrm{e}^{-\alpha_b l_b} \int_{-\infty}^{\infty} |E_3|^2 \mathrm{d}T = \sigma \sum_{k=1}^{K} a_k b_k \propto \sum_{k=1}^{K} a_k b_k, \tag{5}$$

in which $\mathscr{E}$ represents the energy of pulse accumulated by PD, κ means the proportional coefficient brought by PD and σ represents the proportional coefficient and can be obtained by calibration process in actual implementations. Superscript * represents complex conjugation.

This means each loop of TS-CNN can implement vector multiplications and accumulations. Therefore, if the matrix is vectorized into vectors, and are modulated onto pulses in a time-division way, then our system can implement MACs [26].

2.2.2 *Convolution implementation*

The convolution operations in convolution and mean pooling layers are conducted between output feature maps from the previous layer and kernels of the current layer. Consider one convolution operation between one gray picture (feature map) G with $N \times N$ pixels and one gray kernel H with $M \times M$ pixels. In order to let kernels be centered at one pixel of the convoluted feature map, M is usually set to be an odd number less than N. This convolution process can be described precisely as

$$F(p,q) = H(i,j) \otimes G(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{M} H(M+1-i, M+1-j)\, G(p+i-1, q+j-1), \tag{6}$$

in which F represents the convolution result. Parameters p and q not only represent the displacement in row and column during convolution process, but also represent the row and column index of the convolution result respectively while parameters i and j are the row and column index of kernels and pictures.

Since all feature maps and kernels are viewed as matrices, parameters p and q are integer numbers. In traditional convolutions, p and q both take consecutive integer numbers which means that the position of the center of kernels moves from one pixel to another without omitting pixels. However, in more general cases, the center of kernels can move every two or three pixels each at a time. Therefore, parameters s_p and s_q named row and column stride are introduced to describe the number of pixels omitted in row and column direction respectively of kernels' movements during convolution processes. In this case, Eq. (6) can be further illustrated as

$$F(k_p,k_q) = H(i,j) \otimes G(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{M} H(M+1-i, M+1-j)\, G(k_p s_p - s_p + i, k_q s_q - s_q + j), \tag{7}$$

in which k_p and k_q are integers equaling

$$k_p = 1,2,3,\ldots,\left\lfloor \frac{N-M+s_p}{s_p} \right\rfloor, \quad k_q = 1,2,3,\ldots,\left\lfloor \frac{N-M+s_q}{s_q} \right\rfloor, \tag{8}$$

where the operator $\lfloor \cdot \rfloor$ means rounding the number to the nearest integer which is no larger than that number.

In convolutional neural networks, especially for TS-CNN system, various kernels are used to extract different patterns in each convolution layer. More specifically, the inputs in one neuron in each convolution or mean pooling layer can be obtained by firstly using the corresponding kernel to conduct convolutions with all feature maps from the previous layer, then adding up altogether and doing nonlinear stimulation which can be clearly seen in Figure 2. These two processes can be described as

$$\begin{aligned} F_m^{(n)}(k_p,k_q) &= f\left(b_m + \sum_{l=1}^{L} H_{ml}(i,j) \otimes F_l^{(n-1)}(i,j)\right) \\ &= f\left(b_m + \sum_{l=1}^{L} \sum_{i=1}^{M} \sum_{j=1}^{M} H_{ml}(M+1-i, M+1-j) \times F_l^{(n-1)}(k_p s_p - s_p + i, k_q s_q - s_q + j)\right), \end{aligned} \tag{9}$$

in which superscript (n) and $(n-1)$ means the nth and $(n-1)$th layer in CNN. Subscript m and l means the mth and lth neuron in each layer. Capital L describes the total number of nodes in the $(n-1)$th

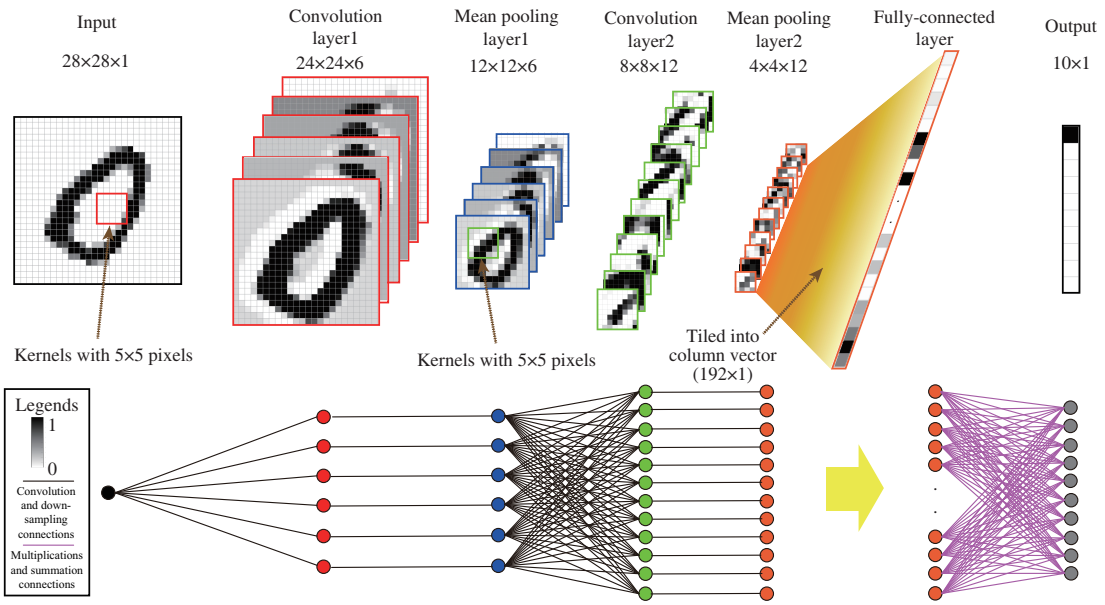

Figure 2 (Color online) Structure of theoretical CNN model. It includes two convolution layers, two mean pooling layers and one full-connected layer. The sample '0' from the training set is used as an example to show all procedures.

layer. Subscript ml means the corresponding kernel between the lth neuron of the $(n-1)$th layer and the mth neuron of the nth layer. b represents bias and f means nonlinear function like Sigmoid, ReLU.

In order to further convert (9) into vector multiplication form which can be modulated by two intensity modulators in TS-CNN system, both feature maps and kernels need to be vectorized. This can be done by shuffling each row of kernels and feature maps into one row which is described as

$$\begin{cases} k = M + 1 - j + (M - i) \times M, \\ k_{k_p,k_q} = s_q k_q - s_q + j + (s_p k_p - s_p + i - 1) \times M, \end{cases} \quad (i, j = 1, 2, 3, \ldots, M), \tag{10}$$

in which the first equation is for kernels while the second equation is for feature maps.

Under this circumstance, Eq. (9) can be changed into

$$\begin{aligned} F_m^{(n)} = (k_p, k_q) &= f\left(b_m + \sum_{l=1}^{L} H_{ml}(i,j) \otimes F_l^{(n-1)}(i,j)\right) \\ &= f\left(b_m + \sum_{l=1}^{L}\sum_{k=1}^{M^2} H_{ml}(k) F_l^{(n-1)}\left(k_{k_p,k_q}\right)\right). \end{aligned} \tag{11}$$

In all, the temporal waveform of the modulated light pulse which was described in (2) can be written as

$$\begin{aligned} &E_{3,k_p,k_q}(l_a, T) \\ &= s_{k_p,k_q}(T) E_2(l_a, T) \\ &= \sum_{k=1}^{M^2} \left\{ \sqrt{H_{ml}(k) F_l^{(n-1)}(k_{k_p.k_q})} \times \mathrm{rect}_{|\Omega\beta_{2a} l_a / M^2|}\left(T - \left(2k - M^2 - 1\right) \frac{|\Omega\beta_{2a} l_a|}{2M^2}\right) \right\} E_2(l_a, T). \end{aligned} \tag{12}$$

As a result, accumulated energy for each pulse after PD which bandwidth is narrower than that of the signals can be obtained as

$$\mathscr{E}_{m,l,k_p,k_q}^{(n)} \propto \sum_{k=1}^{M^2} H_{ml}(k) F_l^{(n-1)}\left(k_{k_p.k_q}\right). \tag{13}$$

In the end, (9) can be rewritten as

$$F_m^{(n)}(k_p, k_q) = f\left(b_m + \sigma \sum_{l=1}^{L} \mathscr{O}_{m,l,k_p,k_q}^{(n)}\right). \tag{14}$$

Therefore, it theoretically proves that our system which is described in Figure 1 can accomplish convolution operations. Since convolution operations are the basis of convolution layers and mean pooling layers, together with fully-connected layer whose basic computations are MACs, the whole CNN model can be implemented by our TS-CNN system. All phsyical meaning of important parameters mentioned above can be clearly seen in Appendix A. More detailed mathematical analysis of the generation of the signals for modulation can be found in Appendix B.

3 Simulations and results

3.1 CNN setup and training

CNN setups and training algorithms are based on the open source deep learning toolbox written by Palm [29, 30] which can be conducted on digit computers. All training and testing data are taken from MNIST handwriting digit recognition dataset [31]. In theoretical model of CNN, 60000 training samples from MNIST are used to optimize the whole configurations of each layer in CNN via stochastic gradient descend (SGD) algorithm while 10000 testing pictures are used to test the recognition accuracy of CNN. All nonlinear functions in this CNN model are chosen to be non-negative Sigmoid function.

As can be seen from Figure 2, in total 5 layers including two convolution layers, two mean pooling layers and one fully-connected layer are made up of theoretical neural networks model. Input data are picture samples from MNIST handwriting digit recognition set with 28×28 pixels. Note that the legend of gray scale turns from white to black as value of pictures ascends in Figure 2 which may well be different from normally defined gray scale legends in order to clearly show the value since more near-zero values exist in feature maps than near-one values. Convolution layer1 has six neurons, each of which has its own kernel with 5×5 pixels. Convolutions are conducted between feature maps from the previous layer and kernels from the current layer. Stride s_p and s_q are set to be one when doing convolutions. Therefore, according to (8), in total six convolution results with 24×24 pixels are obtained after all convolutions are done in this layer. After that, each neuron will add up its own bias to the corresponding convolution result and conduct nonlinear stimulus to obtain its output feature map. All nonlinear stimulus functions in theoretical model are non-negative Sigmoid function.

Mean pooling layer1 is used to down-sample the output feature maps from convolution layer1 in order to further compress the data and improve the generalization ability of the whole network. This layer also has six neurons with each conducting down-sampling to the corresponding feature map from previous layer. Each neuron first divides each feature map with 24×24 pixels from previous layer into 144 groups of 2×2 pixels, then for each group, its mean value is taken as the new pixel of the outputs of this layer. As a result, in total six new feature maps with 12×12 pixels are obtained. These new feature maps will directly become the inputs for convolution layer2.

In convolution layer2, twelve neurons are adopted to extract various patterns for input feature maps. For each neuron in this layer, convolutions are firstly conducted between kernels with 5×5 pixels and output feature maps from the previous layer. Six new convolution results with 8×8 pixels are obtained after operations. Then, all six corresponding convolution results and the bias will be summed up. Afterwards, nonlinear stimulus function is adopted to obtain the resulting feature maps for the next layer. In all, 12 output feature maps, each with 8×8 pixels, are obtained after all these processes are accomplished.

Mean pooling layer2, like the first mean pooling layer, is used to down-sample 12 output feature maps from convolution layer2. Thus, for each neuron, like procedures in mean pooling layer1, input feature maps with 8×8 pixels are firstly divided into 4 groups of 2×2 pixels. Then new pixels of feature map can be obtained by taking mean value for each group. Therefore, in total 12 new feature maps with 4×4 pixels are obtained. Afterwards, these feature maps will be tiled into one column vector with 192 rows and 1 column as the inputs for fully-connected layer.

Fully-connected layer, as the last layer in theoretical CNN model, is used to turn the pattern features extracted from precious four layers into recognition probability. Neurons in this layer, which represent value instead of the whole feature map like previous layers, work in the same way like traditional

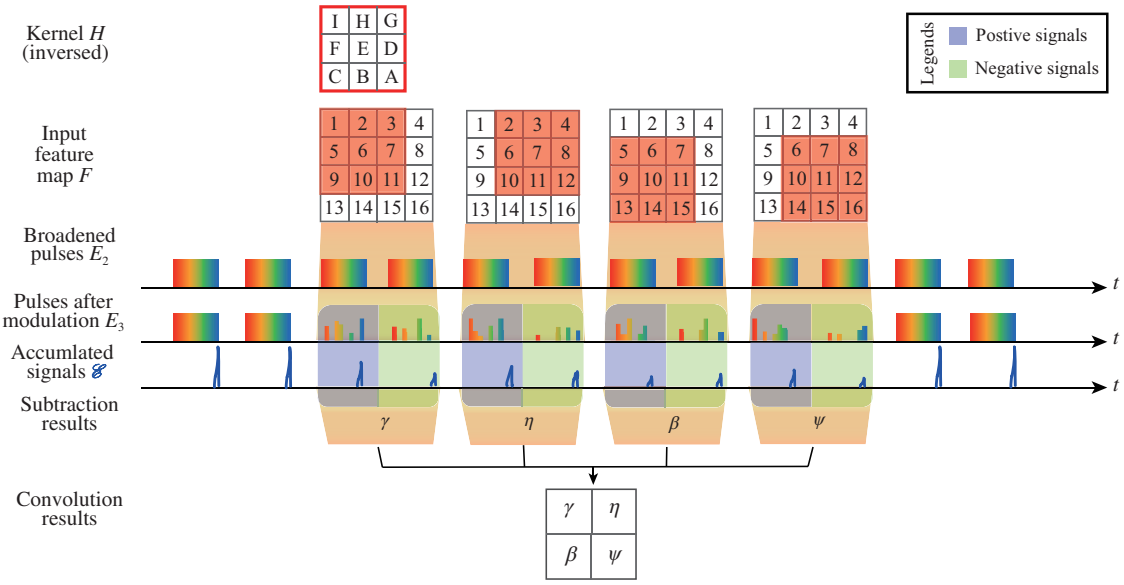

Figure 3 (Color online) Illustration of convolution operations implemented by broadened pulses in TS-CNN. All numbers and alphabets only represent pixel order.

feed-forward neural networks. For each of the ten neurons, multiplications between each weight coefficient and 192 elements in input vectors are firstly conducted. Then, summation between each bias and all multiplication results are conducted, nonlinear stimulus function will later be adopted to turn linear computational results into output recognition probability. At last, one vector with 10 elements representing the probability of recognizing input picture as 0–9 respectively will come out.

3.2 TS-CNN configurations setup

Analytically, in order to set up TS-CNN configurations, pre-trained kernels and all input feature maps of each layer must be firstly encoded into signals for modulation and computation processes. As is illustrated in Figure 3, pre-trained inversed kernels and convolution regions of feature maps (as is depicted by red square-shaped region) from previous layers should be firstly encoded into vectors by order. Then, after dividing each vector into vectors with positive symbols and vector with negative symbols since the non-cognitive optical system can only transmit intensity information, these two vectors are modulated onto peaks of adjoined broadened pulses respectively as is shown in blue and green region in Figure 3. Vector multiplication and summation results can then be obtained by conducting subtractions between adjoined two accumulated pulses after pulse compression, optical-electronic power conversion, energy accumulation and DSP calibration processes. Convolution results can be finally obtained by regrouping subtraction results by order which can be conducted by DSP as well.

In actual simulation implementation, signal encoders which are located before AWGs are crucial devices whose function is to pre-process data including feature maps, kernels, weight matrices, and input vectors into the signals for modulation. These pre-processing procedures include data normalization, symbol separation [26], quantization and time-domain equalization.

Data including kernels and weight matrices entering into signal encoders will be firstly conducted normalization to ensure all data which will later entering into TS-CNN system vary at the same appropriate range. There exist two types of data, feature maps from previous layer (or picture samples for the input) and kernels of the current layer. For feature maps, the normalization process has already done by non-negative Sigmoid function from the previous layer. Picture samples are need to be normalized to the range from 0 to 1 while kernels are need to be normalized to the range from -1 to 1.

Symbol separation can be conducted after data normalization. It is necessary to conduct this process since incoherent optical system cannot convey negative signals. This process can be conducted as follows. Assume the linear component conducts the inner product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ with both K real-number elements. In TS-CNN, since one of this vector, suppose $\boldsymbol{a}$, comes from the results from non-negative Sigmoid fuction, all elements of $\boldsymbol{a}$ is non-negative. Therefore, all we need to do is to separate positive symbol and negative symbol for $\boldsymbol{b}$. The separation is to divide $\boldsymbol{b}$ into $\boldsymbol{b}^+$ and $\boldsymbol{b}^-$. $\boldsymbol{b}^+$ records all positive

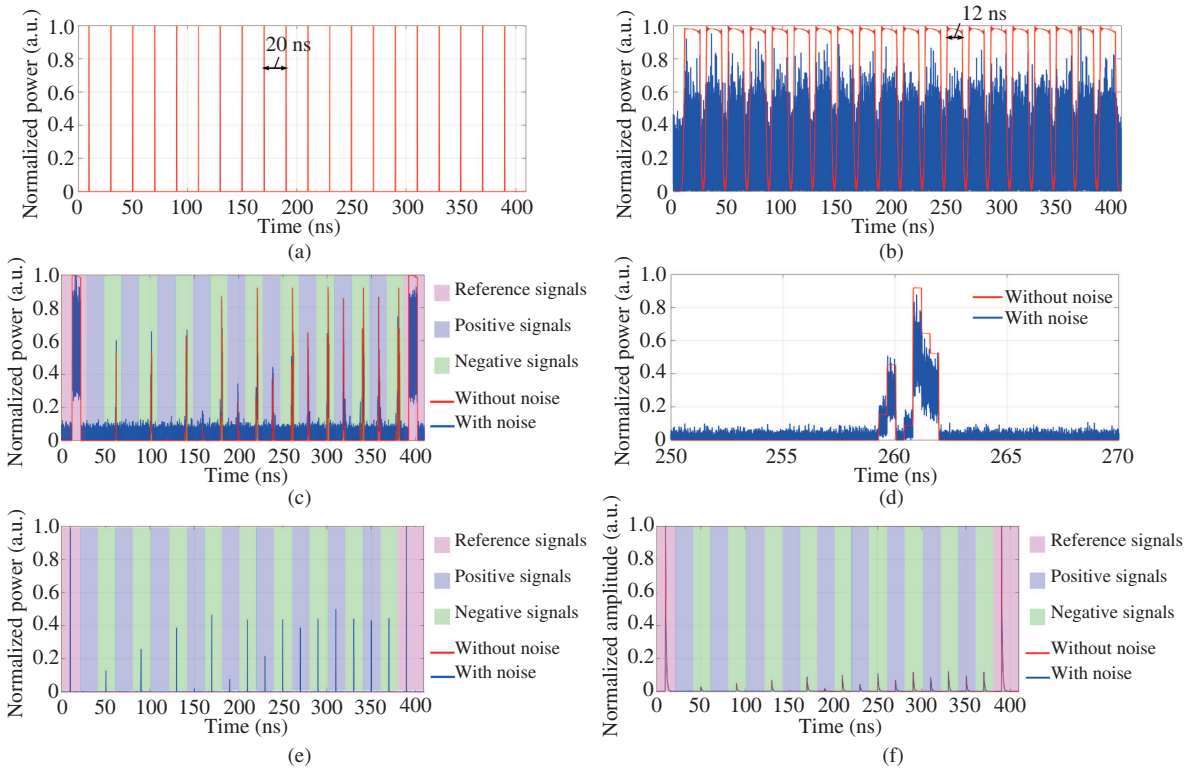

Figure 4 (Color online) Important signals of the last layer of TS-CNN system. (a) Pulses generated by mode-locked laser [①]; (b) broadened pulses after pulse broaden component [②]; (c) pulses after two intensity modulators [③]; (d) zoomed-in picture of modulated pulse [③]; (e) compressed pulses after compression component [④]; (f) accumulated signals after PD [⑤]. Note that index of circles in the square brackets represent the corresponding nodes in Figure 1. This picture depicts the convolution operations between the first kernel with input picture in convolution layer1 in TS-CNN system.

elements in $\boldsymbol{b}$. For negative elements in $\boldsymbol{b}$, the corresponding elements in $\boldsymbol{b}^+$ are all set to be 0. On the contrary, $\boldsymbol{b}^-$ is used to record all negative elements in $\boldsymbol{b}$ as to change the symbol of the correwsponding values in $\boldsymbol{b}$. For positive elements in $\boldsymbol{b}$, the corresponding elements in $\boldsymbol{b}^-$ will be all set to 0. More specifically, if $\boldsymbol{b} = [+\ +\ -\ -\ +\ -]$, then $\boldsymbol{b}^+ = [+\ +\ 0\ 0\ +\ 0]$ and $\boldsymbol{b}^- = [0\ 0\ +\ +\ 0\ +]$. Then, let $\boldsymbol{b}^+$ and $\boldsymbol{b}^-$ multiply with $\boldsymbol{a}$ separately and do the subtraction to obtain the inner product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. This process is reachable since only kernels or weight matrices which are determined in the training procedure contain negative data. After symbol separations, data with positive symbol and negative symbol will be divided apart and later be modulated onto adjoined pulses' peaks which is both shown in Figures 3 and 4(c).

Time-domain equalization can be conducted after symbol separation. In fact, it may always be necessary since only adopting waveshaper can not flatten broadened pulses in meticulous ways due to its limited wavelength resolution. As can be seen in Figure 4(b), even after waveshaper, the peak of each flattened pulse still descend slightly as time goes on. Therefore, time-domain equalization process conducting pre-multiplications between equalization coefficients and data is adopted to further flatten broadened pulses.

Quantization process is conducted after the above three procedures to match the vertical resolution of AWG which is set to be 8 bits in simulation. However, This procedure will introduce quantization noise which can not be erased after signals are received and accumulated after PD. Thus, it will affect recognition accuracy more or less during the test of MNIST handwriting digit recognition.

In all, narrow pulses with repetition rate equaling 50 MHz are generated by mode-locked laser which can be clearly seen in Figure 4(a). Then, as is described in Figure 4(b), after each of them is broadened by pulse broaden component with 170 km-long single mode optical fiber (SMF), its 3 dB width in time domain extends from 1.5 ps to approximately 12 ns. Since the shape of each pulse is irregular, waveshaper is used as a tool to flatten the peaks of these pulses. Due to the limited spectrum resolution in waveshaper, the peak cannot become extremely flat after processing. Flattened pulses are then transmitted into intensity modulator1 which modulates encoded output results, i.e., feature pictures, from previous

loop onto peaks of pulses. After that, pulses will be continuously modulated with encoded configurations, i.e., kernels or weight coefficients, of the current layer by intensity modulator2. Then, modulated broadened pulses which are vividly shown in Figure 4(c) and (d) are compressed narrow after propagating through pulse compression component mainly including 28.5 km, −100 ps/nm/km dispersion compensation fiber (DCF). A PD whose bandwidth is narrower than that of the signals is used to transfer the optical energy into electronic forms and accumulate pulse energy respectively. DSP is used to detect the pulse energy, turn it into the computation results after each loop and sum up feature pictures from the convolution operations between each neuron's kernel with all output feature maps from previous layer pixel by pixel after all computations of one layer of CNN are finished. Nonlinear component implementing non-negative Sigmoid function which is used in implementing convolution layers and fully-connected layer is adopted to change the linear computation results into output feature maps which can be seen in Figure 4(f).

3.3 TS-CNN testing

The performance of TS-CNN is tested by pictures from MNIST handwriting digit recognition test set. In total, all 10000 pictures are adopted to test the theoretical model of convolutional neural networks. In order to test TS-CNN system, 100 pictures which are randomly selected from the whole MNIST test set with 10 pictures for each digit are used. Not only the optoelectronic convolutional neural networks under ideal circumstance in which no noise exists are tested, but also optoelectronic convolutional neural networks under different relative noise levels are tested.

Noise mainly comes from AWGs, EDFAs, PD, and DSP in TS-CNN simulation system. The quantization noise and limited bandwidth of AWGs can cause distortions of the signals for modulation which are irreversible. EDFAs generate amplified simultaneous emission (ASE) noise which will pollute signals carried by light pulses. PD which adopts standard PIN model whose responsibility equals 0.9, generates not only additive white Gaussian noise (AWGN) but also shot noise, ASE-related noise and dark current. Besides, calibration process in DSP which turns signals input into computation results can also introduce estimation noise since calibration coefficients are obtained by prior estimation. For signal amplification, EDFAs are set in this system to amplified optical signals through long distance propagation. All EDFAs' gains are fixed except that of EDFA2. Both thermal noise and shot noise has been added in PD, and noise figure of EDFAs is set to be 4 dB. In the testing of TS-CNN under different relative noise levels, the EDFA2's gain takes value from 10, 12, 15, 17, and 20 dB in each loop. Therefore, relatively better signal quality can be obtained after PD for larger gain. All confusion matrices of testing of TS-CNN are clearly shown in Figure 5.

As can be seen from Figure 5(b) and (c), the accuracy of testing of theoretical convolutional neural network model by 10000 pictures from MNIST test set and by 100 pictures randomly selected from MNIST test set is 99.02% and 99% which is similar with each other. Figure 5(d) shows the accuracy of the testing of TS-CNN without noise is 95% which is lower than the results of theoretical CNN model. This 4% drop in accuracy is mainly caused by the slightly unflat peaks of broadened pulses. It can also be caused by symbol separation, quantization or other processes before encoding configurations of each layer into the signals for modulation in each loop. Figure 5(a) and Figure 5(e)–(i) shows accuracy of testing of TS-CNN under different relative noise levels, which equals 93%, 92%, 91%, 88%, and 76% with the gain of the EDFA2 decreases as 20, 17, 15, 12, and 10 dB. With the increase of the gain of the EDFA2, test accuracy increases since signals of high quality are obtained in each layer.

In most cases, mis-classification happens due to noise, fluctuations and configuration discretizations. As can be seen from the confusion matrices which show the mis-classification of the digit 6 into 2 from Figure 5(d)–(i). In Figure 5(d) with no noise, TS-CNN classified all ten pictures of handwriting digit 6 into the correct category. While as the gain of the EDFA2 drops from 20 to 10 dB, the number of mis-classification of pictures of handwriting digit 6 into 2 increases from 0 to 6 according to Figure 5(e)–(i). Since nonlinear stimulus component exists in each layer, the changes of mis-classification are by no means in linear ways. Minor cases show that fluctuations on peaks of stretched light pulses can somewhat correct the wrong configurations of theoretical CNN model which is shown in the mis-classification of 0 to 6 in Figure 5(c) and (d). Detailed feature maps of each layer and further estimation of time consumption of TS-CNN can be found in Appendixes C and D.

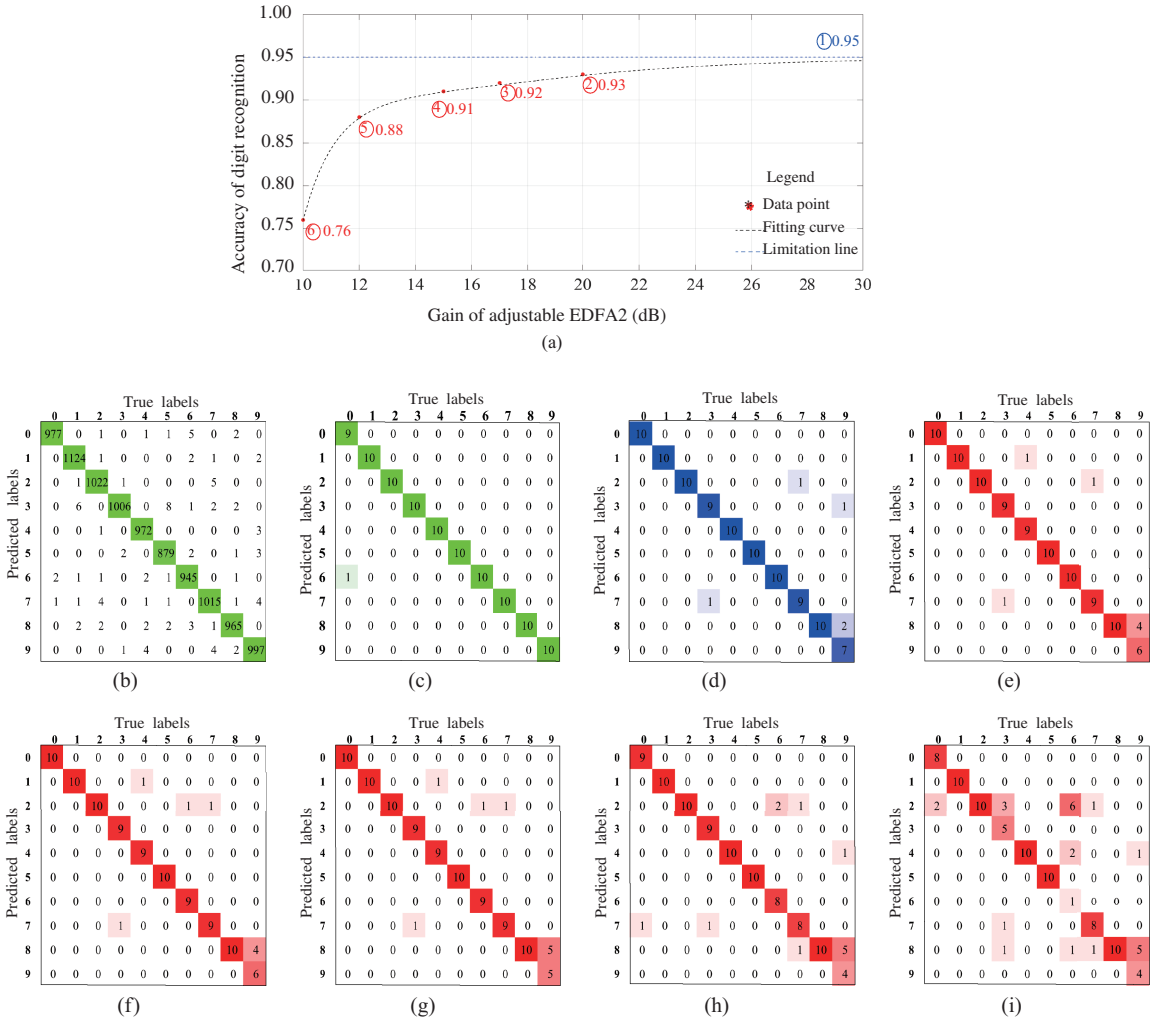

Figure 5 (Color online) Confusion matrices of testing of TS-CNN. (a) The trend of accuracy of handwriting digit recognition with regard to the gain of adjustable EDFA2 in the optoelectronic TS-CNN system. Detailed information of confusion matrix of each data point can be seen in Figure 5(d)–(i). The numbers in circle mark the data points which are shown in Figure 5(a) with their corresponding confusion matrices. (b) Test of theoretical CNN by all pictures from MNIST test set. (c) Test of theoretical CNN by 100 pictures from MNIST test set. (d) Test of noise free TS-CNN [①]. (e) Test of TS-CNN with the gain of EDFA2 equals 20 dB [②]. (f) Test of TS-CNN with the gain of EDFA2 equals 17 dB [③]. (g) Test of TS-CNN with the gain of EDFA2 equals 15 dB [④]. (h) Test of TS-CNN with the gain of EDFA2 equals 12 dB [⑤]. (i) Test of TS-CNN with the gain of EDFA2 equals 10 dB [⑥].

4 Conclusion and discussions

In this paper, we have proposed a new optoelectronic convolutional neural networks. Based on time-stretch method, data can flow in serial ways in TS-CNN system. By implementing a theoretical convolutional neural network with two convolution layers, two mean pooling layers and one fully-connected layer, this structure reaches 95% accuracy under the test of handwriting digit recognition. Besides, this new structure of TS-CNN can have a great robust performance under considerable noise or interference.

Several advantages can be obtained once adopting this loop-shaped optoelectronic structure. Firstly, neural networks with large scale of inputs and more complicated structures can be implemented. Thanks to the adoption of time-stretch technology, information which is modulated onto peaks of broadened pulses can be transmitted and processed in a serial way. Therefore, thorny robust problems which become severe when implementing large-scale complicated neural network models with parallel structures of optoelectronic or all-optical neural networks no longer exist in TS-CNN. Secondly, high power efficiency can be obtained. Instead of adopting von Neumann structure like neural networks implemented by electronic chips or devices in which most power is wasted by exchanging data between memory and computing unit, data will be immediately processed as pulses flow through the optical fiber. Besides,

optical pulses can propagate through long distance with relatively low loss. Thirdly, compared with all-optical structure implementing neural networks, the electronic devices in our optoelectronic system can better process information and compensate for the defects cause by optical devices. For example, peaks of light pulses cannot be completely cut flat only by waveshaper due to its limited spectrum resolution. However, thanks to electronic devices like the signal encoder1 and AWG1, time domain compensation, together with the processed encoded results from the previous layer can be modulated onto peaks of flattened pulses to make them more flat, only leaving slight fluctuations which are acceptable from engineering view. Last but not least, high ability of reconfiguration can be obtained. Various structures of CNN can be implemented by easily changing configurations of each layer as mentioned in part 2 which can reduce the cost of reconfiguration.

Future work will focus more on the integration of the whole system. Technology of hybrid integration [32] which integrates electronic and photonic devices into one chip may does great help in designing and manufacturing integrated TS-CNN system. Photonic devices like lasers [33] , waveguides, reconfigurable optical filter [34], modulators [35] will be integrated as the core computational unit into the chip while electronic devices such as signal encoders, waveform generators, DSP, nonlinear component will be integrated as external processing module. After core optical computing units conduct operations in serial and fast ways, external processing module will further process the results and control the whole optical system. The nonlinear component in the external processing module can be further implemented by optical materials [36] such as saterable absorber [37], phase change material (PCM) [12], or monolayer grapheme [38–41] so as to implement a fully-optical convolutional neural network system based on time-stretch method.

Acknowledgements This work was supported by National Key Research and Development Program of China (Grant No. 2019YFB1803501), National Natural Science Foundation of China (Grant No. 61771284), and Beijing Natural Science Foundation (Grant No. L182043).

Supporting information Appendix A for lists of important variables and notations used in the article, Appendix B for the detailed generation of the the signals for modulation in each layer, Appendix C for the lists of feature maps extracted by electronical implemented CNN and TS-CNN under different relative noise levels, and Appendix D for the estimation of time consumption in TS-CNN. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Garg R, Bg V K, Carneiro G, et al. Unsupervised CNN for single view depth estimation: geometry to the rescue. In: Proceedings of European Conference on Computer Vision, 2016. 740–756
2 Sercu T, Puhrsch C, Kingsbury B, et al. Very deep multilingual convolutional neural networks for LVCSR. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016
3 Prabhakar G, Kailath B, Natarajan S, et al. Obstacle detection and classification using deep learning for tracking in high-speed autonomous driving. In: Proceedings of IEEE Region 10 Symposium (TENSYMP), 2017
4 Al-Qizwini M, Barjasteh I, Al-Qassab H, et al. Deep learning algorithm for autonomous driving using GoogLeNet. In: Proceedings of IEEE Intelligent Vehicles Symposium (IV), 2017
5 Stefaniga S A, Gaianu M. Face detection and recognition methods using deep learning in autonomous driving. In: Proceedings of the 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2018
6 Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. Commun ACM, 2017, 60: 84–90
7 Schaller R R. Moore's law: past, present and future. IEEE Spectr, 1997, 34: 52–59
8 Shen Y, Harris N C, Skirlo S, et al. Deep learning with coherent nanophotonic circuits. Nat Photon, 2017, 11: 441–446
9 Lin X, Rivenson Y, Yardimci N T, et al. All-optical machine learning using diffractive deep neural networks. Science, 2018, 361: 1004–1008
10 Paquot Y, Duport F, Smerieri A, et al. Optoelectronic reservoir computing. Sci Rep, 2012, 2: 287
11 Larger L, Soriano M C, Brunner D, et al. Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing. Opt Express, 2012, 20: 3241–3249
12 Feldmann J, Youngblood N, Wright C D, et al. All-optical spiking neurosynaptic networks with self-learning capabilities. Nature, 2019, 569: 208–214
13 Bueno J, Maktoobi S, Froehly L, et al. Reinforcement learning in a large-scale photonic recurrent neural network. Optica, 2018, 5: 756–760
14 Qian C, Lin X, Lin X B, et al. Performing optical logic operations by a diffractive neural network. Light Sci Appl, 2020, 9: 59
15 Hughes T W, Minkov M, Shi Y, et al. Training of photonic neural networks through in situ backpropagation and gradient measurement. Optica, 2018, 5: 864–871
16 Hengameh B, Scott S, Yichen S, et al. On-chip optical convolutional neural networks. 2018. ArXiv:1808.03303
17 Mahjoubfar A, Churkin D V, Barland S, et al. Time stretch and its applications. Nat Photon, 2017, 11: 341–351
18 Ng W, Rockwood T D, Sefler G A, et al. Demonstration of a large stretch-ratio ($M = 41$) photonic analog-to-digital converter with 8 ENOB for an input signal bandwidth of 10 GHz. IEEE Photon Technol Lett, 2012, 24: 1185–1187
19 Wong J H, Lam H Q, Li R M, et al. Photonic time-stretched analog-to-digital converter amenable to continuous-time operation based on polarization modulation with balanced detection scheme. J Lightwave Technol, 2011, 29: 3099–3106

20 Valley G C. Photonic analog-to-digital converters. Opt Express, 2007, 15: 1955–1982
21 Huang Y Y, Zhang W J, Yang F, et al. Programmable matrix operation with reconfigurable time-wavelength plane manipulation and dispersed time delay. Opt Express, 2019, 27: 20456–20467
22 Chen C L, Mahjoubfar A, Jalali B. Optical data compression in time stretch imaging. PLoS One, 2015, 10: 1–11
23 Goda K, Tsia K K, Jalali B. Serial time-encoded amplified imaging for real-time observation of fast dynamic phenomena. Nature, 2009, 458: 1145–1149
24 Goda K, Ayazi A, Gossett D R, et al. High-throughput single-microparticle imaging flow analyzer. Proc Natl Acad Sci USA, 2012, 109: 11630–11635
25 Xing F J, Chen H W, Lei C, et al. A 2-GHz discrete-spectrum waveband-division microscopic imaging system. Opt Commun, 2015, 338: 22–26
26 Zang Y B, Chen M H, Yang S G, et al. Electro-optical neural networks based on time-stretch method. IEEE J Sel Top Quantum Electron, 2020, 26: 1–10
27 Goda K, Solli D R, Tsia K K, et al. Theory of amplified dispersive Fourier transformation. Phys Rev A, 2009, 80: 043821
28 Plancherel M, Leffler M. Contribution Á L'Étude de la reprÉsentation D'une fonction arbitraire par des intÉgrales dÉfinies. Rend Circ Matem Palermo, 1910, 30: 289–335
29 Palm R B. MNIST_recognization done by BP networks. 2019. https://github.com/rasmusbergpalm/DeepLearnToolbox
30 Palm R B. Prediction as a candidate for learning deep hierarchical models of data. Dissertation for M.Sc. Degree. Copenhagen: Technical University of Denmark, 2012
31 Lecun Y, Cortes C, Burges C J C. The MNIST database of handwritting digits. 2019. http://yann.lecun.com/exdb/mnist/
32 Kato K, Tohmori Y. PLC hybrid integration technology and its application to photonic components. IEEE J Sel Top Quantum Electron, 2000, 6: 4–13
33 Davey R P, Smith K, McGuire A. High-speed, mode-locked, tunable, integrated erbium fibre laser. Electron Lett, 1992, 28: 482
34 Zhang D K, Feng X, Huang Y D. Tunable and reconfigurable bandpass microwave photonic filters utilizing integrated optical processor on silicon-on-insulator substrate. IEEE Photon Technol Lett, 2012, 24: 1502–1505
35 Kensuke O. High-speed silicon-based integrated optical modulators for optical-fiber telecommunications. In: Proceedings of International Society for Optical Engineering, 2014
36 Miscuglio M, Mehrabian A, Hu Z, et al. All-optical nonlinear activation function for photonic neural networks. Opt Mater Express, 2018, 8: 3851–3863
37 Selden A C. Pulse transmission through a saturable absorber. Br J Appl Phys, 1967, 18: 743–748
38 Bao Q L, Zhang H, Ni Z H, et al. Monolayer graphene as a saturable absorber in a mode-locked laser. Nano Res, 2011, 4: 297–307
39 Lim G K, Chen Z L, Clark J, et al. Giant broadband nonlinear optical absorption response in dispersed graphene single sheets. Nat Photon, 2011, 5: 554–560
40 Hu X, Wang A D, Zeng M Q, et al. Graphene-assisted multiple-input high-base optical computing. Sci Rep, 2016, 6: 32911
41 Yadav R K, Aneesh J, Sharma R, et al. Designing hybrids of graphene oxide and gold nanoparticles for nonlinear optical response. Phys Rev Appl, 2018, 9: 044043