



Classification of time-domain waveforms using a speckle-based optical reservoir computer

UTTAM PAUDEL,^{*} MARTA LUENGO-KOVAC, JACOB PILAWA, T. JUSTIN SHAW, AND GEORGE C. VALLEY

Electronics and Photonics Laboratory – The Aerospace Corporation, 2310 E El Segundo Blvd, El Segundo, CA 90245, USA

^{*}uttam.paudel@aero.org

Abstract: Reservoir computing is a recurrent machine learning framework that expands the dimensionality of a problem by mapping an input signal into a higher-dimension reservoir space that can capture and predict features of complex, non-linear temporal dynamics. Here, we report on a bulk electro-optical demonstration of a reservoir computer using speckles generated by propagating a laser beam modulated with a spatial light modulator through a multimode waveguide. We demonstrate that the hardware can successfully perform a multivariate audio classification task performed using the Japanese vowel speakers public data set. We perform full wave optical calculations of this architecture implemented in a chip-scale platform using an SiO₂ waveguide and demonstrate that it performs as well as a fully numerical implementation of reservoir computing. As all the optical components used in the experiment can be fabricated using a commercial photonic integrated circuit foundry, our result demonstrates a framework for building a scalable, chip-scale, reservoir computer capable of performing optical signal processing.

© 2020 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

1. Introduction

The rapid growth of data traffic leading to the emerging big data era has made special-purpose signal processing hardware a necessary tool. In particular, an analog processor capable of performing on-the-fly signal processing would significantly reduce the overhead of the storage and transmission of large datasets and the need for post-processing [1]. Over the past few years, several realizations of optical signal processors [2,3], such as optical neural networks [4–6] and optical reservoir computers [7–10], have been demonstrated that generate random matrices and perform matrix multiplication passively at the speed of light, with much lower power consumption compared with electronic implementation of these operations. Electrical neural networks require the storage of large matrices of weight coefficients, which inherently slows down the computation as the information must be passed back and forth between the memory and processing units and is energy costly. In an optical neural network, the weight coefficients are an intrinsic part of the photonic system, and the information is transmitted at the speed of light [11].

Photonic reservoir computing (RC), in particular, has attracted a lot of attention [2,3], as it can be used for a wide variety of classification, prediction, and memory tasks [8–21]. Due to its recurrent nature, it can mimic and predict the complex nonlinear temporal dynamics of various systems with memory [22,23]. Unlike typical recurrent neural networks (RNNs), for RC only the output weights are trained, making it significantly simpler and faster to train than most other RNNs and guaranteed to converge; RC does not suffer from the vanishing or exploding gradient problem [22,24]. Furthermore, RC does not require backpropagation, which is computationally intensive.

In this article, we present results on spatially-distributed bulk electro-optical RC using laser speckle in a multimode optical waveguide as the “reservoir”, which allows for parallelized input

and could potentially be scaled into a photonic integrated circuit (PIC) to form a multi-GHz-bandwidth, photonic processor. We show that the system can map the dynamics of a near-chaotic Mackey-Glass (MG) waveform. We demonstrate the electro-optical system performed, with good accuracy, a multi-variate, multi-category classification task with preprocessed audio data.

The existing implementations of photonic RC can be broadly split into two categories: delay based [14–16,18–21] and optical node array based [7,8,9]. Although impressive, high-speed performance has been achieved with delay-based RCs, they are intrinsically limited by the total delay time, which is often on the order of a hundred nanoseconds [19,21] and which sets the limitation for real-time processing speeds and for chip-scale integration.

Similarly, the previous work by Dong et al. used free-space propagation and speckle generated from scattering from a rough surface [7], with light modulated using a one-bit Digital Micromirror Device. This generates a larger number of neurons, but is also not compatible with high-speed, chip-scale integration. Here, we build up on the approach by Scofield et al. [9], which uses speckles generated by propagation through optical waveguides and a 1D array of 12-bit light modulators. With this approach, we demonstrate that only 200 neurons are required to perform multivariate classification tasks. The essential components of our RC, an array of high-speed modulators and detectors, can be readily fabricated using commercial foundries with bandwidth exceeding 45 GHz [25]. In addition, the calculations presented here demonstrate that a 100-micron wide, 10-cm long planar waveguide provides sufficient speckle mixing for building a 100-neuron reservoir computer. Actual realizations of the planar waveguide are expected to use spiral geometries that will have additional mode mixing due to defects and bends, and this should significantly reduce the waveguide length required for sufficient mode mixing. Such waveguides have been fabricated in tight spirals using a silicon-on-insulator (SOI) PIC with a total footprint less than 1 cm² for spectroscopic applications [25,26]. This provides an immediate path for developing a scalable chip-scale RC system. The processing speed of such a device would only be limited by the optical delay of the chip (<1 ns for <10-cm long waveguide), the modulator bandwidth, and the photodiode response time. This opens the possibility of performing real-time signal processing of radio-frequency (RF) and other high frequency signals, or rapid serial processing of very large datasets.

2. Architecture and experimental demonstration of RC

A reservoir computer consists of three layers (Fig. 1): an input layer, which may accommodate multiple simultaneous inputs, a large reservoir of neurons with random but fixed connections, and an output layer, which may yield multiple outputs (e.g. prediction, classification). The reservoir effectively expands the dimensionality of the problem by distributing the input into multiple neurons. The recurrent nature of the reservoir computer makes it suitable for performing time-series prediction and classification.

The discrete time evolution of the neurons of a reservoir computer can be expressed as a recursive non-linear mapping of the form [22,24]:

$$X(t+1) = \alpha F_{NL} [\mathbf{W} \cdot X(t) + \mathbf{V} \cdot u(t)] + (1 - \alpha) X(t) \quad (1)$$

where the N-dimensional column vector $X(t)$ corresponds to the neuron values at time t , $\alpha \in (0, 1)$ is the leaking rate, F_{NL} is a sigmoidal function (e.g. the logistic function, a hyperbolic tangent), \mathbf{W} is a matrix representing the connections between the neurons in the reservoir, \mathbf{V} is a matrix of the random weights between the inputs and the neurons, and $u(t)$ is the multi-dimensional input data used to train and test the reservoir computer.

In our physical implementation of a reservoir computer, the input data and feedback neuron values, weighted by leaking rate α , are injected into the multimode fiber by intensity modulating the spatial profile of the light using a 1D array spatial light modulator (SLM). A multimode fiber transforms the input light and approximates the random weighted matrix operators \mathbf{W} and

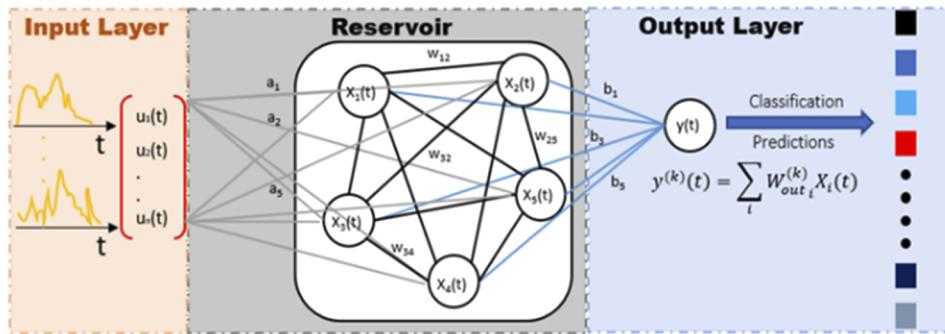


Fig. 1. Topology of the reservoir computer consisting of a single input and a single output layer connected via a complex reservoir of neurons with random but fixed connections.

V. Nonlinear activation is a critical step for machine-learning algorithms that maps between the input and the output response of the neurons. In our experiments this is achieved by using the nonlinear saturation effect of the camera pixels F_{NLcam} , which performs an operation $F_{NL}[g(t)] = F_{NLcam}(|g(t)|^2)$ on the raw neuron values.

The laboratory implementation of the RC along with the schematic of the experiment is shown in Figs. 2(a) and 1(b). A continuous-wave (CW) frequency stabilized single mode laser operating at 780 nm illuminates a 1D 12-bit SLM with 640 pixels using a pair of cylindrical lenses. The V_π voltage of the SLM is experimentally measured and then used to appropriately scale the input data to use the full dynamic range of the SLM. The size of an individual SLM pixel is 100 μm . The SLM output is collimated and focused using a pair of cylindrical lenses with 8x magnification. The beam is further focused using a 40x microscope objective lens into a 2.9 m long, 300- μm diameter, 0.39-NA multimode fiber such that the SLM is imaged into the core of the fiber. Each SLM pixel excites different values of the modal expansion coefficients in the fiber. The propagation of several thousand optical modes through the fiber performs the equivalent of the random weight operations (multiplication by \mathbf{W} and \mathbf{V}) needed for RC. The output of the fiber is imaged onto a 2D CCD camera using 20x objective lens such that each speckle lobe occupies $\sim 10 \times 10$ camera pixels, and the intensities of 200 pixels, corresponding to the neuron outputs of the reservoir computer, are recorded on a computer.

The neuron values times α are summed with the previous neuron values multiplied by $(1 - \alpha)$. The neuron values are scaled to drive the SLM and address the SLM for the next timestep. The waveforms to be processed are also fed simultaneously to the SLM pixels. D-dimensional data can be fed into the waveguide by simultaneously modulating D SLM pixels. The output of the reservoir computer at time t is given by $y(t) = \mathbf{W}_{out} X(t)$, where \mathbf{W}_{out} is a matrix of trained output weights. The state $X(t)$ of the reservoir neurons at each timestep is measured by running the system with the training time-series dataset input and collected into one state matrix $X = (X^T(t_1), X^T(t_2), \dots, X^T(t_M))$, where $X \in \mathbb{R}^{N \times M}$ and M is the total number of time steps, and then the single layer of output weights, \mathbf{W}_{out} , for each classification task is calculated using regularized linear regression:

$$\mathbf{W}_{out} = (XX^T + \lambda I_N)^{-1}XY, \quad (2)$$

where $\mathbf{Y} = (y(t_1), y(t_2), y(t_3), \dots)$ is the target output matrix, I_N is the identity matrix, and λ is the regularization parameter. In contrast to a tradition RNN, where all weights can be adjusted to train the network, training in RC is done for the output layer, which is typically fast, and is always convergent [20,24].

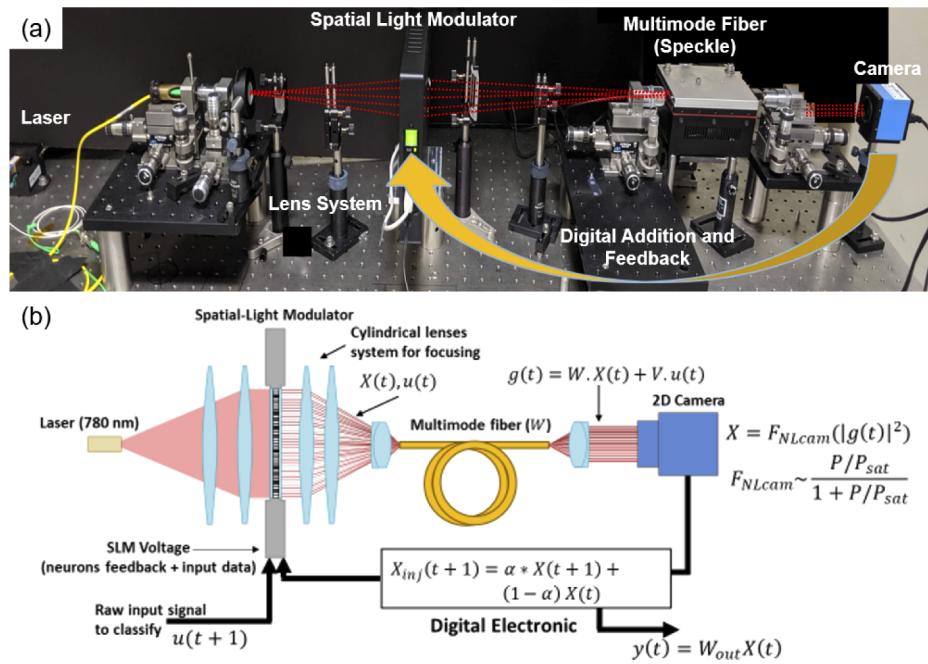


Fig. 2. (a) Laboratory realization of speckle based electro-optical reservoir hardware. The output of a 780 nm continuous-wave laser is modulated using a 1D SLM array and is imaged into a multi-mode fiber. The speckle pattern generated at the output of the multimode fiber is imaged using a 2D camera array, stored, added with the previous iteration of the experiment, and fed back into the SLM. (b) Schematic of the electro-optical RC showing the optical and electronic processing part. The non-linear activation is performed through the saturation effect of the camera.

3. Experimental results

3.1. Time series recovery

To demonstrate that the implemented RC hardware is capable of mapping complex near-chaotic waveforms into the reservoir and can replicate such a signal, we performed a time-series recovery experiment using the numerical solution to the MG equation with time delay (τ) of 17 [27]. The MG equation is a nonlinear time-delay differential equation that can generate complex, chaotic dynamics that are often found in naturally occurring systems with some temporal memory [27]. A 2300 timestep long MG solution time series is fed into the RC hardware and the output neurons are recorded. The first half of the neuron data (X) is used to train the output weights (W_{out}) such that the input waveform is recovered, and this calculated W_{out} is used with the second half of the data to predict the input data. The input and recovered waveforms are plotted in Fig. 3(a). The chaotic attractors for the input and recovered MG timeseries are plotted in Figs. 3(b) and 3(c) respectively. The plots show that the system operates in a high-dimensional regime and the reservoir computer is fully able to map the near-chaotic dynamics with minimal noise.

Since W_{out} is calculated from the training dataset and then used to recover the test input data, the optical setup needs to be highly stable for the entire time it takes to run the training and test data in order to replicate such near-chaotic dynamics. This replication data demonstrates that the opto-electronic hardware has enough dynamic range, bandwidth, capability for dimensionality expansion, and stability to track a near-chaotic time-domain signal.

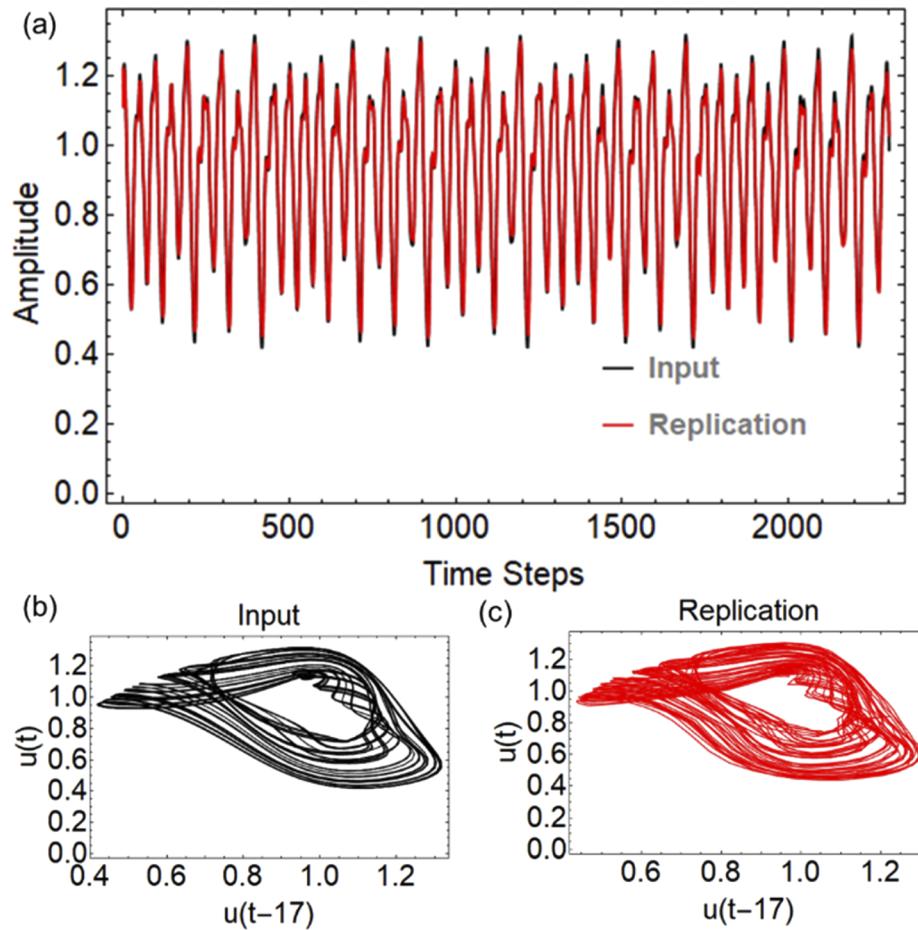


Fig. 3. (a) Amplitude of the Mackey-Glass function with $\tau = 17$ as a function of time input to the electro-optical RC (black) and recovered using the trained output weights (red). Chaotic attractor for the (b) input and (c) recovered generated from the time delayed MG input and output from the RC. The plots show that the system operates in a high-dimensional regime and the RC is fully able to map the near-chaotic dynamics.

3.2. Audio classification

To demonstrate the ability of the experimental implementation of the electro-optical reservoir computer to perform classification tasks [12,13,16, 20], we used the Japanese vowels public data set [28,29]. This is a standard benchmark data set that consists of multivariate time-series data for a multiple-category classification task. The data consists of the first twelve Mel-frequency cepstral coefficients (MFCCs) of utterances of the Japanese diphthong ‘ae’ by nine different speakers, and the task is to classify each utterance by speaker. MFCCs of audio signals are widely used for speech recognition tasks. They are calculated by computing the power spectrum of an audio signal, binning the power spectrum based on the mel scale (which more closely matches the human auditory system – e.g., less sensitivity at higher frequencies), taking the log of the power at each mel frequency bin (because human hearing follows a log scale), and then taking the discrete cosine transform of the mel log powers. The MFCCs are the amplitudes of the resulting spectrum. The MFCCs for each utterance were padded with zeros to be 29 timesteps long. The original dataset consists of 270 training utterances (30 utterances by 9 speakers) and 370 different

testing utterances (24–88 utterances by the same 9 speakers). For the experimental measurements, 200 utterances each were randomly selected from the training and testing datasets. Therefore, the 200 testing utterances were different than the 200 utterances used for training.

The twelve MFCCs were fed in as inputs to the electro-optical RC using twelve pixels of the SLM, and the rest of the pixels are used to input the feedback neurons values from the previous timestep. After propagation through the fiber, the input signal gets mapped to all 200 output neurons. Figure 4(a) shows a schematic of the experiment. As the input is fed in, the states of the individual neurons in the reservoir change (Fig. 4(b)). The state of the reservoir is collected at each time step, and the training portion is used to calculate \mathbf{W}_{out} using Eq. 2. This matrix is then used to predict the output and classify each utterance during the testing portion. The predicted output closely matches the input values (Fig. 4(c)), showing that the reservoir can reconstruct the input data accurately. This \mathbf{W}_{out} was also used to predict the classifications of the testing inputs. At each timestep, the classification output consists of 9 probability-like numbers corresponding to the 9 different speakers. The classification of each timestep is taken to be the speaker corresponding to the largest of these probability-like numbers. The predicted classification of each utterance was defined as the mode of the classification of each timestep within an utterance.

Figure 5(a) shows the predicted and correct classifications of the speakers for the training and testing utterances. The accuracies for the training and testing datasets are 88.5% and 81.5% respectively. The classification accuracy is expected to be higher for the training dataset as \mathbf{W}_{out} was calculated from the neuron values corresponding to this dataset. The system was still able to achieve very high classification accuracy for the testing dataset, which consisted of 200 utterances the system had never seen before. Using simulated RC with comparable number of neurons, we were able to obtain up to 98.1% classification accuracy. For comparison, with just the raw input

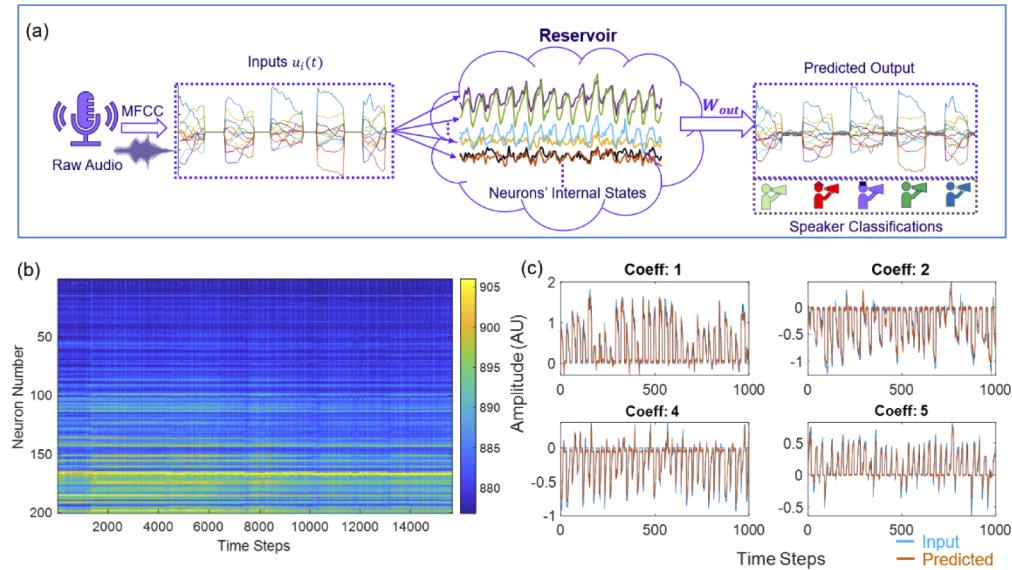


Fig. 4. (a) A high-level schematic of the experiment with the measured data. The raw audio signals are digitally converted into their MFCCs. The first 12 MFCCs are used as the input $u(t)$ into the reservoir, changing the states of the individual neurons in the reservoir. At each time step, the matrix of the neurons states is multiplied by \mathbf{W}_{out} in order to get the output and the predicted speaker classifications. (b) is experimentally recorded value of each neuron as a function of time. (c) shows the predicted value (orange) for 4 of the 12 MFCCs calculated from the output of the reservoir, compared to the input value (blue) of the MFCCs.

instead of the dimensionality-expanded neuron values, using a linear classifier (i.e. finding \mathbf{W}_{out} such that $y(t) = \mathbf{W}_{out}u(t)$, where $u(t)$ is the input signal) we achieved an accuracy of only 43.2%. The Hidden Markov Model used in [28] achieved an accuracy of only 96.2%. The reduced accuracy in the experiment is attributed to small thermal and mechanical instabilities of the multimode fiber, mostly due to external environmental conditions (e.g. air currents, temperature fluctuations, and vibrations). The system is highly stable for ~ 1 hour, although we still achieve significantly above random accuracy ($>50\%$) for test and train datasets run within a 6 hour window. We expect these instabilities to be eliminated in a temperature controlled, packaged chip-scale device.

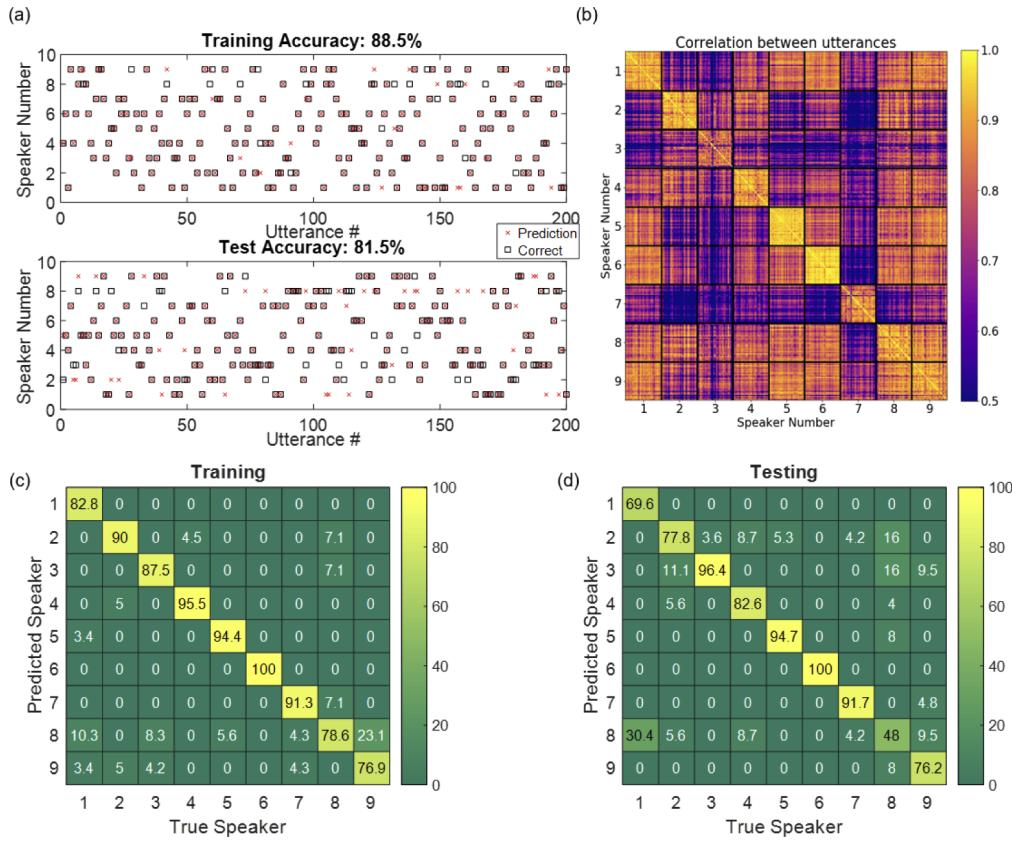


Fig. 5. (a) Predicted (red crosses) vs correct (black boxes) classification of the speakers for the training (top) and test (bottom) dataset. Overlap between a cross and a box indicates that the utterance was correctly classified. Each utterance label is represented by a box and any overlap between adjacent boxes is a plotting artifact. (b) calculated cross-correlation between each of the training utterances. The lighter yellow signifies highly correlated sounds and purple less correlation. (c) and (d) show confusion plots for the same training and testing datasets as in (a).

Speaker classification is not a trivial task as different utterances by the same speaker are not necessarily highly correlated and may in fact have large variations. Figure 5(b) shows the calculated correlation between each of the utterances in the full training dataset (270 utterances, 30 utterances per speaker). For some speakers (e.g. Speakers 5 and 6), different utterances by the same speaker are highly correlated. However, for other speakers (e.g. Speaker 3), different utterances by the same speaker show very little correlation. If one were to attempt classification

based on correlation alone, it would be difficult to distinguish highly correlated but distinct speakers, such as Speakers 8 and 9. In contrast, RC expands the dimensionality of the signal into a higher-dimensional feature space, in which features that distinguish different classes can be separated using simple linear regression techniques, thus performing the classification task.

Figures 5(c) and 5(d) show confusion plots for the same training and testing datasets as in Fig. 5(a). The data is normalized such that each column, corresponding to each correct speaker, adds up to 100%. The diagonal elements show the percentage of correctly classified utterances per speaker, and the off-diagonal elements show the percentage of incorrectly classified utterances. These plots show that the utterances by certain speakers (e.g. Speaker 8) are more likely to be misclassified, whereas utterances by certain speakers (e.g. Speaker 6) were correctly classified with very high accuracy. These confusion plots also show which classification errors are more likely (e.g. Speaker 1 is likely to be misidentified as Speaker 8, but never Speaker 6).

The results reported here are for a non-trivial multivariate dataset; for simple header classification problems, we achieve 100% classification accuracy.

4. Simulations

4.1. Reservoir size, leaking parameter, and noise

A simulated reservoir computer was also trained and tested with the Japanese vowels dataset for comparison to the performance of the experimental implementation. For the simulation, V was a randomly generated matrix with values uniformly distributed between -0.5 to 0.5, and W was a randomly generated matrix with values from -0.5 to 0.5 and then normalized such that the spectral radius was 1.25. Hyperbolic tangent was chosen as the non-linear function F_{NL} . W_{out} was then calculated by training the simulated reservoir computer with the training portion of the Japanese vowels dataset. This W_{out} was used to predict the classifications of the testing portion of the Japanese vowels dataset. Each utterance is zero-padded as necessary to be 29 time steps long, and the predicted classification for each utterance is determined by a “winner-take-all” algorithm such that the label predicted for 15 or more time steps is taken to be the label for the utterance. The accuracy was then determined by comparing the predicted classifications to the correct classifications.

Figure 6(a) shows the accuracy of the test dataset classification as a function of leaking rate and reservoir size. Even with a small reservoir size (25 neurons), we were able to achieve >90% accuracy. The accuracy was not very sensitive to the leaking rate for leaking rates >0.2. For a detailed analysis on the performance comparison between RC and traditional RNNs, see [24].

To test the reservoir computer’s robustness to noise, we added Gaussian noise to the test data. Figure 6(b) shows the classification accuracy as a function of the amplitude of the Gaussian noise for two cases. The classification accuracy drops as the amplitude of the noise increases. Furthermore, increasing the number of neurons does not improve the robustness to noise. The achieved classification accuracy with our experimental hardware is therefore not likely to be limited by the number of neurons, but by noise and other non-idealities in the system.

This simulation study demonstrates that a few hundred neurons are sufficient to perform complex classification tasks with very high accuracy. Such numbers of neurons can be implemented by fabricating an array of high-speed modulators and detectors on a chip-scale device, using readily available standard foundry processes [25].

4.2. Simulations of planar waveguides for PIC realization of RC

Our experiments and those of Dong et al. [7,8] show empirically that speckle in multimode fiber and speckle from rough surface scattering provide excellent random projections for RC. For chip-scale integration of the spatially-distributed RC architecture, it is important to demonstrate that the random matrix generation and multiplication can be performed in a planar multimode

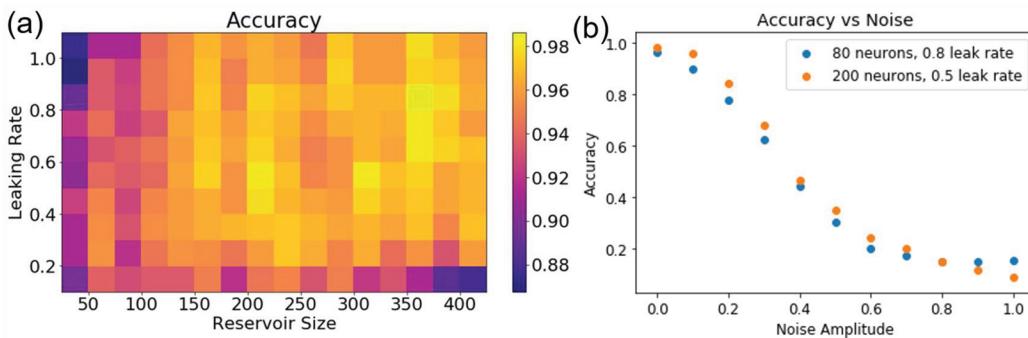


Fig. 6. Reservoir computer simulated on a computer using random matrices. (a) Accuracy of the test dataset classification as a function of leaking rate and reservoir size. (b) Classification accuracy as a function of amplitude of Gaussian noise added to the test data for (blue) 80 neurons and 0.8 leak rate, and (orange) 200 neurons and 0.5 leak rate.

waveguide. Here we perform detailed full wave calculations that demonstrate speckle in a multimode planar waveguide also works extremely well for RC and gives performance equivalent to that obtained by using a fully numerical simulation based on projections with a pseudo-random matrix with entries uniformly distributed between -1 and +1.

We start with the well-known solutions for the TM modes $\Psi_i(x,z)$ in planar dielectric waveguides [30] of the form

$$\Psi_i(x,z) = [A_i \cos(h_i x) + B_i \sin(h_i x)] \exp(i \beta_i z) \quad (3)$$

where x is the transverse dimension and z is the direction of propagation, A_i and B_i are constants determined by the boundary conditions and input field, h_i is the transverse mode wavenumber, and β_i is the propagation constant. We ignore the field in the cladding as this is unimportant for the speckle pattern used in RC and use solutions for the TM mode. The parameters h_i and β_i are obtained by numerical solution of the standard transcendental equation for waveguide modes [30,31]. For a 100-micron wide silicon on SiO₂ waveguide, solution of this transcendental equation yields 409 roots each with its own spatial pattern.

The spatial profile of m^{th} pixel of the SLM imaged into the fiber can be expanded in terms of the 409 TM modes of the fiber:

$$E_m(x,z) = \sum_{i=1}^{409} a_{m,i} \Psi_i(x,z) \quad (4)$$

where $a_{m,i}$ is the i^{th} spatial expansion coefficient for the m^{th} pixel. $a_{m,i}$ can be calculated by taking the spatial profile at the entrance of the waveguide ($z=0$), multiplying each side of Eq. 4 by $\Psi_j(x,z)$, integrating over the width of the guide, and using the orthogonality of the modes to perform the integral on the righthand side to obtain $a_{m,j} = \int_0^L E_m(x,0) \Psi_j(x,0) dx$, where L is the width of the guide.

For a 100-micron silicon guide, the correlation length for the speckle pattern in the output plane, at which the correlation function drops to 0.5, is 0.7 micron. Output pixels much larger than 0.7 micron will average over multiple speckle lobes while pixels smaller than 0.7 micron will give redundant information. The calculation was performed for 1-micron wide pixels or 100 pixels at the output plane. There are 409 modes in the 100-micron guide; the smallest transverse wavenumber is $h_1 = 0.03 \text{ micron}^{-1}$ and the largest $h_{409} = 13. \text{ micron}^{-1}$. Thus, the spatial scales range from $2\pi/13 \sim 0.5 \text{ microns}$ for the 409th mode to $2\pi/0.03 \sim 200 \text{ micron}$ for the first mode. These modes are excited at 100 1-micron-wide input ports, they mix as they propagate down the 10-cm long waveguide, and finally they form the speckle pattern to be measured at 100 output

ports, also 1 micron wide. Since the propagation constants β_i are different for each mode, the interference pattern of all the modes varies rapidly with propagation down the guide. If one input port is excited, less than 5 cm of propagation is needed for this 1-micron input field to fill the guide.

The first step in the calculations is to find the expansion coefficients for each of the 100 input fields. These expansion coefficients are multiplied by the values of the appropriate feedback neuron [elements of the vector $X(t)$ in Section 2] or the input signal $u(t)$. Second, the electric field at the output of the guide is calculated by finding the superposition of all these modes. Next, we take the absolute value squared of the field at the output plane and integrate it over the individual output pixels. Then a nonlinearity is applied to these values and they are added to a fraction ($1-\alpha$) of the neurons at the previous time step. We tried simulations using only the square law but these perform poorly. In all our calculations we need a saturable nonlinearity, such as a sigmoidal or hyperbolic tangent, but the results rarely depend on the exact shape of the nonlinear function. Experimentally, the camera is taking the modulus square of the optical electric field and saturates above a certain field intensity. This saturation level can be controlled by changing the laser power. In our experiments, we ran with typically $\sim 10\%$ of the pixels saturated. We find that the square-law and saturation response of the camera pixels, which has qualitatively similar saturation behavior as a sigmoidal at higher intensities, is sufficient as the nonlinear activation function.

To summarize the full wave calculations, the matrix products $W.X$ and $V.u$ in Eq. (1) are replaced by modulation of the modal expansion coefficients at the input to the guide, propagation down the guide and calculation of the intensity of the optical speckle field at the output of the guide. Figures 7(a)–7(d) compare the results for the MG waveforms using speckle mixing as described above to a conventional calculation based on Eq. (1) with a random matrix W that is calculated from random numbers uniformly distributed between -1 and $+1$. The top row gives the test MG waveform superposed on the recovered MG waveform and the difference between the recovered and test waveforms for the conventional calculation while the bottom row shows the same task using the full wave speckle mixing calculation. The parameters of the calculations are 99 neurons, leaking rate $\alpha = 0.5$ and nonlinearity $F_{NL} = 1/[1 + \exp(-x)]$. The training sequence consisted of 8 Mackey-Glass waveforms with time delays from 6 to 20 in steps of 2. The test sequence consisted of the same 8 MG waveforms but in random order. The obvious conclusion is that for MG waveforms the planar-waveguide speckle-based reservoir computer performs nearly identically to a reservoir computer in which a random matrix is used. We have performed similar calculations for test and training waveforms composed of multiple sinusoids—again with no significant difference.

We note that the speckle calculations take a lot longer than the random matrix calculations, but of course, there is no intent to make a numerical reservoir computer based on speckle calculations. The intent is to do the speckle “calculations” passively using the optical RC hardware in the time it takes light to propagate through the waveguide.

In addition to the time-series reproduction, we perform a classification task using the full wave simulator. The performance of the speckle-based RC was nearly identical to that of numerical RC. These results demonstrate the viability of a speckled-based RC architecture implemented in a chip-scale platform using a planar multimode waveguide.

Simulations also show that multiple small reservoir computers (less than ~ 100 neurons) can be used simultaneously with the same input waveform to increase the effective number of neurons, and this implies that large reservoir computers (~ 1000 s of neurons) can be constructed from multiple copies of the same small PIC. Such multiple near-identical copies of the PIC can be readily fabricated through a commercial foundry.

There are a number of potential serial and parallel methods for combining multiple RCs with a small number of pixels into a more powerful RC system with a larger number of pixels. We have

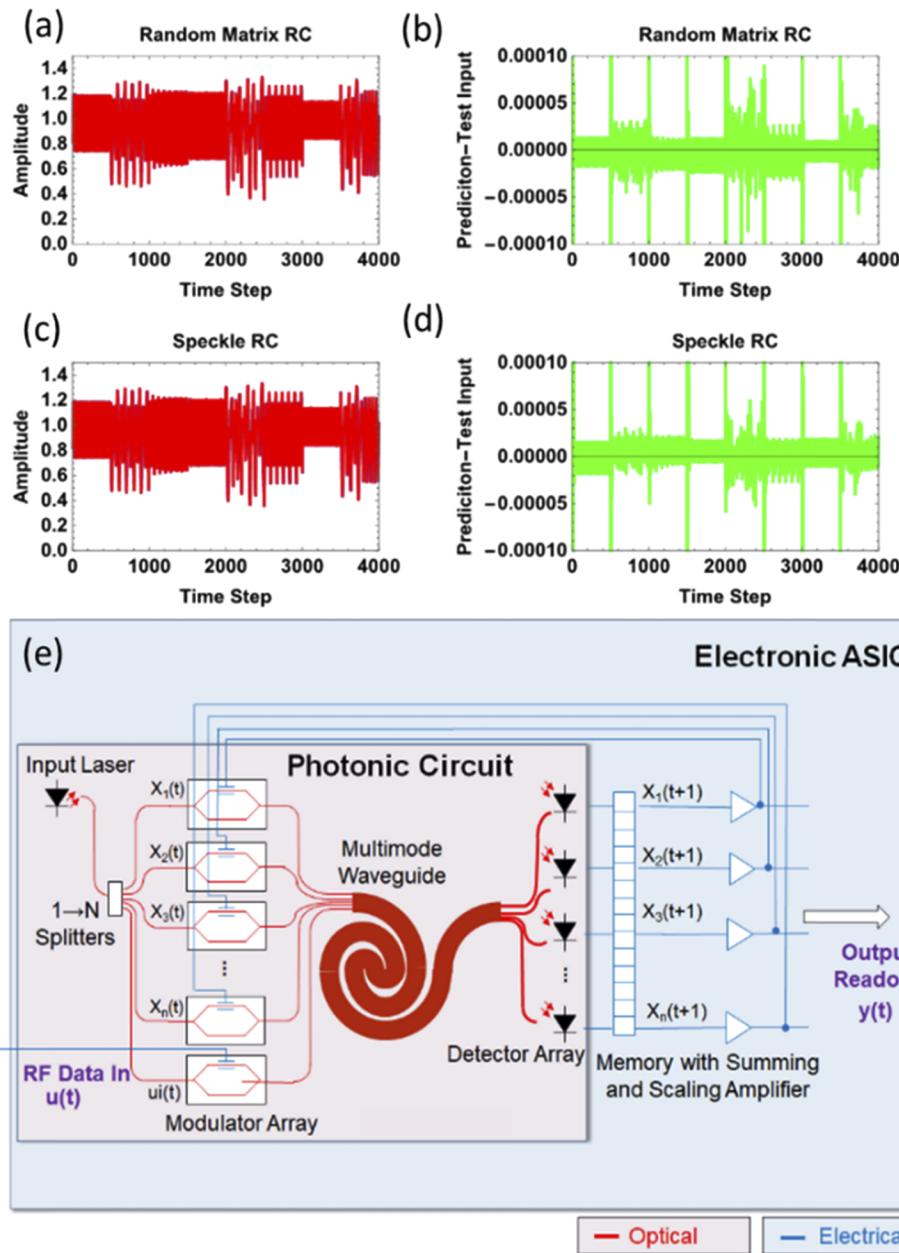


Fig. 7. (a) Mackey-Glass amplitude recovery and test. Amplitude of recovered signal (red) and input signal (blue) as a function of time step for a random matrix reservoir computer. Due to a near-perfect reconstruction of the input data, the red curve obscures almost all of the blue curve. (b) Difference between recovered and input signals for random matrix. (c) Amplitudes for speckle-based RC. (d) Difference for speckle-based reservoir computers. (e) Concept schematic for a chip-scale reservoir computer consisting of a single laser, an array of modulators, a multimode planar waveguide, and an electronic adder and amplifier to drive the modulator with the analog signal. The shaded red region consists of optical components and the blue region consists of electrical components.

performed simulations of one parallel method in which the input signal $u(t)$ is injected into n RCs and the output of the n RCs is processed as one large neuron matrix to obtain the output weights. These simulations gave similar results to simulations using a single n times larger RC. Essentially, the multiple small RCs give the equivalent of a block diagonal matrix \mathbf{W} for the large RC and no difference was found between a block diagonal random matrix \mathbf{W} and a fully populated matrix \mathbf{W} . We also performed simulations on one class of serial RC system in which $u(t)$ is injected into a small RC, the output of the small RC is injected into another small RC etc., thus forming a deep RC [32]. Much further investigation is required to determine the optimal strategy for using multiple small RCs in a single system.

An example schematic of the integrated circuit implementation of the reservoir computer is given in Fig. 7(e). The 1D SLM can be replaced with an array of on-chip modulators, the optical fiber can be implemented as a planar multimode waveguide, and the camera can be replaced with an array of on-chip integrated detectors [25]. The laser can also be fabricated on-chip with indium phosphide technology or surface mounted on the SiO_2 platform. It is important to note that the intention is not to implement all-optical RC, as a hybrid device can take advantage of the strengths of both optical and electrical systems to achieve a multi-GHz, low size, weight, and power hardware accelerator. For example, using the built-in saturable nonlinearity of the detectors, the neuron activations can be performed with no added processing complexity. The summing and scaling of the signal to drive back the modulators can be performed electrically at a multi-GHz clock rate using a digital field-programmable gate array (FPGA) or an analog application-specific integrated circuit (ASIC) that can be seamlessly integrated with the photonic chip. The readout of the neuron values X will require an array of analog-to-digital converters, as is the case with most analog device, and the training to calculate W_{out} will be done digitally. Such a hybrid system will benefit from the low power consumption required for generating random matrices and performing matrix multiplications optically, in parallel and in one clock cycle, at the speed of light.

5. Conclusions

In this Article, we demonstrated an electro-optical reservoir computer using speckles generated from a multimode waveguide. We demonstrated that the hardware can perform a classification task of multivariate time series audio data. We performed full wave optical calculations of the speckle-based optical reservoir computer implemented in a chip-scale platform with a 100 μm wide SiO_2 multimode planar-waveguide and demonstrated that the system performed as well as a fully numerical implementation for time-series reconstruction and classification tasks. The RC architecture demonstrated here through bulk optics can be readily built on an integrated platform combining a photonic chip fabricated at a commercial foundry with electrical ASIC components. The demonstrated architecture and results provide a pathway for building a hybrid hardware accelerator capable of performing on the order of GHz bandwidth real-time signal processing.

Funding

U.S. Air Force (FA8802-19-C-0001).

Disclosures

The authors declare no conflicts of interest.

References

1. J. Capmany and D. Novak, "Microwave photonics combines two worlds," *Nat. Photonics* **1**(6), 319–330 (2007).
2. G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, "Recent advances in physical reservoir computing: a review," *Neural Networks* **115**, 100–123 (2019).

3. G. Van der Sande, D. Brunner, and M. C. Soriano, "Advances in photonic reservoir computing," *Nanophotonics* **6**(3), 561–576 (2017).
4. 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**(6406), 1004–1008 (2018).
5. Y. Shen, N. C. Harris, S. Skirla, 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**(7), 441–446 (2017).
6. 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**(6), 756–760 (2018).
7. J. Dong, S. Gigan, F. Krzakala, and G. Wainrib, "Scaling up echo-state networks with multiple light scattering," in *2018 IEEE Statistical Signal Processing Workshop (SSP)*, pp. 448–452.
8. J. Dong, M. Rafayelyan, F. Krzakala, and S. Gigan, "Optical reservoir computing using multiple light scattering for chaotic systems prediction," *IEEE J. Sel. Top. Quantum Electron.* **26**(1), 7701012 (2020).
9. A. C. Scofield, G. A. Sefler, T. J. Shaw, and G. C. Valley, "Recent results using laser speckle in multimode waveguides for random projections," in *Optical Data Science II, Vol. 10937 of 2019 International Society for Optics and Photonics*, pp. 109370B.
10. K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, "Experimental demonstration of reservoir computing on a silicon photonics chip," *Nat. Commun.* **5**(1), 3541 (2014).
11. Y. Zhang, M. Chen, S. Yang, and H. Chen, "Electro-optical neural networks based on time-stretch method," <https://arxiv.org/abs/1909.07788>.
12. A. Dejonckheere, F. Duport, A. Smerieri, L. Fang, J. L. Oudar, M. Haelterman, and S. Massar, "All-optical reservoir computer based on saturation of absorption," *Opt. Express* **22**(9), 10868–10881 (2014).
13. Q. Vinckier, F. Duport, A. Smerieri, K. Vandoorne, P. Bienstman, M. Haelterman, and S. Massar, "High-performance photonic reservoir computer based on a coherently driven passive cavity," *Optica* **2**(5), 438–446 (2015).
14. F. Duport, A. Smerieri, A. Akroud, M. Haelterman, and S. Massar, "Fully analogue photonic reservoir computer," *Sci. Rep.* **6**(1), 22381 (2016).
15. A. Argyris, J. Bueno, and I. Fischer, "Photonic machine learning implementation for signal recovery in optical communications," *Sci. Rep.* **8**(1), 8487 (2018).
16. S. Ortín, M. C. Soriano, L. Pesquera, D. Brunner, D. San-Martín, I. Fischer, and J. M. Gutiérrez, "A unified framework for reservoir computing and extreme learning machines based on a single time-delayed neuron," *Sci. Rep.* **5**(1), 14945 (2015).
17. C. Mesaritakis and D. Syvridis, "Reservoir computing based on transverse modes in a single optical waveguide," *Opt. Lett.* **44**(5), 1218–1221 (2019).
18. F. Duport, B. Schneider, A. Smerieri, M. Haelterman, and S. Massar, "All-optical reservoir computing," *Opt. Express* **20**(20), 22783–22795 (2012).
19. L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutiérrez, L. Pesquera, C. R. Mirasso, and I. Fischer, "Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing," *Opt. Express* **20**(3), 3241 (2012).
20. L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udal'tsov, Y. K. Chembo, and M. Jacquot, "High-speed photonic reservoir computing using a time-delay-based architecture: million words per second classification," *Phys. Rev. X* **7**(1), 011015 (2017).
21. D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer, "Parallel photonic information processing at gigabyte per second data rates using transient states," *Nat. Commun.* **4**(1), 1364 (2013).
22. H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks - with an erratum note," *GMD Technical Report* **148**(34), 13 (2001).
23. H. Jaeger and H. Haas, "Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication," *Science* **304**(5667), 78–80 (2004).
24. M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Comput. Sci. Rev.* **3**(3), 127–149 (2009).
25. AIM Photonics brochure, "Integrated photonics process design kit (PDK)" (AIM Photonics, 2019). http://www.aimphotonics.com/s/PDK-One-Pager_2019.pdf
26. M. Piels and D. Zibar, "Compact silicon multimode waveguide spectrometer with enhanced bandwidth," *Sci. Rep.* **7**(1), 43454 (2017).
27. M. C. Mackey and L. Glass, "Oscillation and chaos in physiological control systems," *Science* **197**(4300), 287–289 (1977).
28. M. Kudo, J. Toyama, and M. Shimbo, "Multidimensional curve classification using passing-through regions," *Pattern Recognit. Lett.* **20**(11–13), 1103–1111 (1999).
29. UCI Machine Learning Repository, "Japanese Vowels Data Set," <https://archive.ics.uci.edu/ml/datasets/Japanese+Vowels>
30. A. Yariv, "Coupled-mode theory for guided-wave optics," *IEEE J. Quantum Electron.* **9**(9), 919–933 (1973).
31. Y. Dattner and O. Yadid-Pecht, "Analysis of the effective refractive index of silicon waveguides through the constructive and destructive interference in a Mach-Zehnder interferometer," *IEEE Photonics J.* **3**(6), 1123–1132 (2011).
32. C. Gallicchio, A. Micheli, and L. Pedrelli, "Deep reservoir computing: A critical experimental analysis," *Neurocomputing* **268**, 87–99 (2017).