Processing IMU action recognition based on brain-inspired computing with microfabricated MEMS resonators

Tianyi Zheng¹,², Wuhao Yang¹, Jie Sun¹,², Zhenxi Liu¹,², Kunfeng Wang¹,² and Xudong Zou¹,²,

¹ The State Key Laboratory of Transducer Technology, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, 100000, People's Republic of China
² School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing, 100000, People's Republic of China

* Author to whom any correspondence should be addressed.
E-mail: zouxd@aircas.ac.cn

Keywords: MEMS resonators, human action recognition, reservoir computing

Abstract
Reservoir computing (RC) decomposes the recurrent neural network into a fixed network with recursive connections and a trainable linear network. With the advantages of low training cost and easy hardware implementation, it provides a method for the effective processing of time-domain correlation information. In this paper, we build a hardware RC system with a nonlinear MEMS resonator and build an action recognition data set with time-domain correlation. Moreover, two different universal data set are utilized to verify the classification and prediction performance of the RC hardware system. At the same time, the feasibility of the novel data set was validated by three general machine learning approaches. Specifically, the processing of this novel time-domain correlation data set obtained a relatively high success rate. These results, together with the dataset that we build, enable the broad implementation of brain-inspired computing with microfabricated devices, and shed light on the potential for the realization of integrated perception and calculation in our future work.

1. Introduction
Nowadays, due to the significant improvement of the processor computing ability and continuous optimization of software algorithms, machine learning (ML) has been widely used in the field of artificial intelligence [1]. The outstanding contributions of ML can be seen in the fields of computing vision, natural language processing, and action recognition [2–4]. In order to adapt to complex tasks, artificial neural networks, recursive neural networks (RNNs), convolutional neural networks algorithms have been successively proposed and updated iteratively. At the same time, the massive data generated and transmitted by the internet of things (IoT) has new requirements for real-time information processing of edge computing devices based on new architecture in this information age [5, 6]. With the continuous development of IoT and ubiquitous sensor applications, active training and testing using time-dependent information have become a key goal of edge computing [7].

Reservoir computing (RC) is originally designed to reduce the training complexity and mitigate the training process of RNN [8–10]. RNN requires complex algorithms to train the connection weight, resulting in slow convergence and consuming a large amount of calculation [11, 12]. The RC system avoids this problem by only changing the linear output layer weight (usually called the readout layer) instead of training each connection weight in the reservoir layer. Moreover, the fixed reservoir without an adaptive update is suitable for hardware implementation of various nonlinear dynamic systems [9, 13]. Generally, there are two ways to implement RC in a hardware system: (1) multiple nonlinear physical nodes form the reservoir layer (2) using a single nonlinear node and a single feedback loop to form the reservoir layer based on time-domain multiplexing. Compared with the first method, the RC based on the time-delay nonlinear system greatly reduces the difficulty of hardware implementation, which has been successfully proven in a variety of systems, such as spintronic
Hardware RC system requires two crucial properties, multi-dimensional nonlinear mapping ability and memory capacity that can memorize data a few steps or tens of steps ago. Due to its excellent Duffing nonlinear performance and suitable attenuation characteristics, MEMS resonators are appropriate as the nonlinear node of the hardware reservoir system [31, 32]. Most of the prediction and classification tasks processed by hardware RC are based on the general data sets, such as parity benchmark [30], nonlinear autoregressive moving average (NARMA) task [13, 33, 34], Santa Fe laser [26, 35, 36], Mackey–Glass time-series tasks [35], non-linear channel equalization benchmark task [37, 38], signal classification [18, 26], isolated word recognition [13, 18, 30], video action recognition [34, 39], and handwritten digit classification [20, 40]. But so far, there is no research on using hardware RC systems to process human action recognition data sets with time-independent information. Besides, human action recognition plays a significant role in human-to-human interaction and interpersonal relations in this modern society [41]. In order to achieve the goal of integration of perception and calculation, which is to perceive and process the acceleration signal of inertial measurement unit (IMU), we first build a time-delay nonlinear RC system based on MEMS resonator and verified its classification ability with the signal classification task and the prediction ability of the system with NARMA task. Then we design and produce a novel IMU action recognition data set. Compared with the traditional action recognition data set with video information, the information of this IMU data set is more direct and efficient because of its relatively small amount of data. The processing result of utilizing the hardware system to deal with this IMU human action recognition data set shows our hardware RC system is suitable for processing time-dependent dynamic signals. This work lays the foundation for using a MEMS resonator to sense IMU signals and process them simultaneously. Finally, we envisaged using MEMS resonators to sense and process IMU signals simultaneously based on the conclusion of this paper.

The structure of this paper is as follows. In section 2, we describe the methods part, including the fabrication process of MEMS resonators, hardware implementation of an RC structure, and design of the IMU data set. In section 3, we give a detailed discussion about the simulation and experiment results of the RC hardware system on typical tasks. In section 4, the paper ends with a brief conclusion and an outlook for possible future work.

2. Method

2.1. Fabrication process of MEMS resonators

A crucial part of RC system processing information is the nonlinear conversion of input time-dependent data to high-dimensional [42]. Since high-dimensional information is easier to be processed by post-processing algorithms, this nonlinear mapping part is directly related to the accuracy of testing different data sets. Our hardware RC system implements this mapping part through a single clamped-clamped silicon beam.

This single clamped-clamped beam is manufactured in our fab using the silicon-on-insulator (SOI) process which is shown in figures 1(a)–(g). The process uses a total of three masks and starts with a 4-inch SOI chip (thickness of device layer: 25 μm, the thickness of oxide layer: 1 μm, the thickness of substrate layer: 300 μm). After pre-cleaning with acetone, alcohol, and deionized water sequentially, the photoresist is patterned by lithography with the first mask. Sputtering Cr (20 nm)/Au (300 nm) on the patterned photoresist, and using lift-off process to complete the electrode layer production. The photoresist is then graphed by lithography on the electrode layer using a second mask, and the device layer is etched with deep reactive ion etching (DRIE) using the patterned photoresist as a mask. For the subsequent back process, a layer of photoresist is spun to protect the front structure. And then, a third mask is used to pattern the photoresist on the back surface of the SOI chip by photolithography. Using the patterned photoresist as a mask to etch the substrate layer by DRIE. Then the middle oxide layer is released by the method of reactive ion etching (RIE), leaving the movable structure suspended. Finally, the front protective layer was removed by RIE and fuming nitric acid. The SEM image of the microfabricated structure is shown in figure 1(h). The thickness of the beam is designed to be 25 μm, while its in-plane length and width are chosen to be 580 μm and 9 μm, respectively. The electrode is set to be 150 μm × 150 μm, which is convenient for wire bonding.

2.2. Hardware implementation of RC structure

RC systems based on MEMS resonators are manufactured using standard materials and manufacturing processes, and their interface circuits and feedback circuits are easily integrated, making them particularly attractive for hardware implementations. The time-delay nonlinear RC system mainly consists of three parts: input layer, reservoir layer, and output layer. The input layer and output layer are both realized by data acquisition (DAQ) of National Instruments. In the input layer, the original input signals \( u(t) \) from various data sets are processed by mask signal \( m(t) \) and multiplied by the input gain \( \beta \). The mask signal plays an important role...
Figure 1. The fabrication process flow of the signal clamped-clamped silicon beam. (a) The original SOI chip. (b) Sputtering Cr/Au as the electrode (using the first mask). (c) Fabricate the structure with DRIE (using the second mask). (d) Use photoresist as a protective layer. (e) Back-etching the substrate layer with DRIE (using the third mask). (f) Removing the oxide layer with RIE. (g) Removing the protective layer. (h) The SEM image of the signal clamped-clamped silicon beam structure.

Figure 2. The structure of time-domain RC system with hardware implementation. The blue part stands for the DAQ card and the green part shows the inter-face and feedback circuits, while the yellow part performs the STM32F407 microcontroller. The black arrows indicate the direction of the analog signal in the system and the blue arrow shows the image of the hardware device, while the red arrow stands for the sequence waveform of some crucial signals.

in breaking the symmetry of the data and keeping the nonlinear nodes in the transient state to obtain various transient responses of the input data. Our mask signal consists of randomly selected numbers in the range \((-1, 1)\) with zero mean and unit variance. As illustrated in figure 2, the processed signal \(I = \beta u(t)m(t)\) is output by the digit to analog converter (DAC) of the DAQ card. And then, the processed signal is modulated by the ac signal \(V_{ac} \sin(2\pi ft)\) for the convenience of driving resonators. For visualization, the sequence waveform of the DAC output shows the part of the input signal of the NARMA10 data set (this data set will be introduced later). The vertical axis represents the time span and the horizontal axis represents the actual magnitude of the processed signal.

The crucial part of the reservoir layer is the manufactured resonator. The nonlinear vibration state of a single clamped-clamped beam is described by the following Duffing equation with the forced harmonic vibration:

\[
m\ddot{x}(t) + c\dot{x}(t) + k_1x(t) + k_3x^3(t) = F \cos(\omega t)
\]

where \(m\) is effective mass, \(c\) is damping coefficient of harmonic system, \(k_1\) is linear spring constant, \(k_3\) is nonlinear spring constant, \(F \cos(\omega t)\) is harmonic force using the electrostatic drive, \(x(t)\) is the vibration amplitude over time. The value \(k_3\) determines the nonlinear performance of this system. When the resonator is driven electrostatically, \(k_3\) is determined by nonlinear mechanical spring constant \(k_{m3}\) and electrostatic coefficient \(k_{e3}\), namely, \(k_3 = k_{m3} + k_{e3}\). In this experiment, \(k_3\) is positive and approximately equal to \(3.8 \times 10^{12}\) N m\(^{-3}\). The natural frequency \(f_n\) is equal to 252,000 Hz in the open-loop test experiment. As shown in figure 1(h), the single clamped-clamped beam has four electrodes, including one driving electrode, one detection electrode, and two electrodes on both sides of the beam. When the dc bias voltage signal is applied to the beam and the harmonic driving signal is applied to the driving electrode, the nonlinear output signal can be observed from the detection electrode.
The single clamped–clamped resonator operates in a nonlinear state by electrostatic drive. The drive signal output by the DAQ card is injected into the driving metal electrode and the bias signal \( V_b \) is loaded on the beam directly. The output of the resonator is also detected by the electrostatic method, and the signal is read out by the external interface circuit which contains a trans-impedance amplifier, in-phase amplifier, and bandpass filter. After that, the voltage signal injects into the envelope detector (ENV) for the purpose of demodulation, that is, restores the original valid amplitude information. Part of the ENV output signal is depicted in figure 2. The blue line stands for the output signal from the external interface circuit and the orange signal represents the output signal of ENV. All the circuit modules are integrated into this green circuit board as shown in the figure with the dimension of 11.3 cm \( \times \) 5.55 cm.

After being processed by ENV, the output signal is captured by the analog to digital converter of the DAQ card for postprocessing and also enters the delay feedback loop at the same time. STM32F407 microcontroller with the main frequency of 168 MHz is used to realize the delay time of the signal in the digit domain. Delay time equals \( \tau = N \theta \), where \( N \) is the virtual nodes of the RC system and duration \( \theta \) is the time interval between virtual nodes. As shown in figure 2, the blue line performs the input signal of the STM32F407 microcontroller and the orange line stands for the output signal after a delay which the delay time equals \( \tau = 20 \) ms. After the output signal of the STM32F407 microcontroller is multiple by the feedback gain \( \alpha \) which is achieved by the voltage divider circuit, the obtained signal add to the input signal from the DAQ card and inject into the resonators to form the delay feedback loop. The signal processed by this RC system is captured and stored by DAQ card and then tested and trained by the ridge regression method:

\[
W = y'X^T(XX^T + \lambda I)^{-1}
\]

which \( X \) is the input matrix, \( W \) is the weight matrix that needs to be trained, \( y' \) is the target value, \( \lambda \) stands for the regularization parameters to prevent overfitting.

### 2.3. The design of the IMU action recognition data set

Most of the original information of classification data set commonly used in scientific research region is not time-domain information, such as MNIST image recognition data set and KTH video recognition data set. Each pixel of the image in the MNIST data set does not have the actual time sequence information. Articles using this data set [20, 40] only arrange each pixel in order and then input them in sequence, which does not mean that there is an actual time-domain correlation between pixels. Although the KTH video data set [34, 39] is used to identify time-related image frame information, the pixel point of each frame image is also time-independent. Therefore, we need to design a data set closely related to timing information in order to use MEMS resonators for processing.

Based on this, we designed and produced a novel type of action recognition data set named IMU action recognition. Processing this data set with the RC system with MEMS resonators is fundamental to achieving the goal of integration of perception and calculation. This data set contains six actions, namely ‘circling’, ‘stepping’, ‘walking’, ‘running’, ‘going upstairs’, ‘going downstairs’. As shown in figures 3(a) and (b), four same IMU sensors made from Wit-Motion are used to capture human movements. This IMU sensor with the dimension of 36.1 mm \( \times \) 42 mm is fixed on the experimenter’s two wrists and two ankles to collect data. In order to ensure consistency, the background environment for each data collection is the same. There are eight male experimenters with different body shapes participating in the recording of our data set, and each person does each action ten times, that is, a total of 480 sets of data. In each action, the three-axis acceleration signal of each IMU sensor is adopted and saved at a sampling rate of 20 Hz. The recording time for each action is 5 s, so the acceleration information for each axis consists of 100 data points. Each group of data consists of three-axis acceleration signals of four IMU sensors, that is, there are 12 groups of signals that represent an action. All the signals are output by the IMU sensor, without any data processing, directly forming the data set. Figure 3(c) shows the time-domain diagram of raw acceleration signals of four IMU sensors. Each figure represents an action that needs to be classified. Each figure consists of four dials, representing the data of four IMU sensors, and each dial has three time-series curves, representing the acceleration signal of each axis. The abscissa indicates the number of points recorded, and the ordinate represents the signal amplitude.

In order to verify the versatility and feasibility of this action recognition data set, we use three commonly used classification algorithms in the field of ML for testing, namely, linear regression, logistic regression, and multi-layer perceptron (MLP). In all three methods, we used 432 data sets for training and the remaining 48 data sets for testing. (The partitioning of data sets will be covered in detail in the next chapter.) Linear regression is the most basic method in ML which uses regression equations (functions) to model the relationship between one or more independent variables (eigenvalues) and dependent variables (target values). The test result of linear regression is that the accuracy is equal to 53.75%. Logistic regression is to increase the nonlinear sigmoid function on the basis of linear regression, thereby improving the accuracy of classification to a certain extent. Logistic regression test results improved to 72.5%. MLP is a highly parallel information processing system with
strong adaptive learning ability. It is a nonlinear system that can deal with complex multi-input and multi-output classification tasks. The accuracy of this relatively complex algorithm in processing IMU data sets is up to 76.46%. In general, the three ML algorithms get different results on IMU data sets, and the accuracy increases with the increase of algorithm complexity. That means the design of the IMU action recognition data set is relatively reasonable and comparable.

3. Result and discussion

Having proved the generality of the IMU action recognition data set above, we need to testify the usability of our hardware RC system based on MEMS resonator. The hardware RC system has two basic crucial performances, namely classification ability, and prediction ability. Two general data sets are utilized to illustrate these two performances of the system. The classification performance is experimentally investigated via signal classification task, while the classification performance is discussed via NARMA prediction task. Finally, we use the hardware RC system with better performance to process IMU action recognition data sets.

The MEMS resonator of the probe station works with a mechanical pump in the experiment. The measured decay time of the resonator equals $T_d \approx 2$ ms under this condition. Continuously proven by predecessors [36, 42], the system performance is optimal when the relation between the resonant decay time $T_d$ and the minimum time scale $\theta$ is $T_d = 5\theta$, that is, $\theta = 0.4$ ms.

3.1. Signal classification task

The signal classification task is relatively simple, because its input data is linearly separable, which means that this data set can be obtained with high accuracy using a linear system. The input signal $x(t)$ of this task is a random combination of the sine wave and square wave, and the period of each waveform is composed of ten discrete points. The target function $y(t)$ is 1 for sine waves and 0 for square waves.
Figure 5. Performance of hardware RC systems for the NARMA prediction test with $m = 1, 2, 5, 10$. The abscissa represents the order of the data set, and the ordinate represents the NMSE of every situation. The blue line stands for the simulation result and the orange line stands for experiment result.

Figure 6. NMSE for the NARMA prediction task in the input gain-AC drive voltage plane with hardware RC system. The color indicates the value of NMSE and the $X$-axis represents the strength of the feedback, and the $Y$-axis represents AC drive voltage. As the order $m$ increases, the accuracy of the RC hardware system gradually decreases.

Because the accuracy of data processing using the nonlinear RC system is very high which up to 100%. To illustrate the differences between systems, normalized mean square error (NMSE = \( \frac{1}{L} \sum_{n=1}^{L} (y(n) - \hat{y}(n))^2 / \text{var}(y) \)) is used to measure system performance.

The BPFM(bifurcation point frequency modulation) method discussed in detail in studies [32] is used to find the operating point of the system and determine the order of adjustment parameters. In this experiment, the drive frequency of resonators equals $f_d = 253\,000$ Hz and the DC bias voltage equals $V_{dc} = 10$ V, while AC drive voltage equals $V_{ac} = 1.4$ V. Since this data set is relatively simple, sufficiently high accuracy can be achieved without a lot of virtual nodes.

Here, we choose $N = 50$. Based on this, the delay time of the feedback delay loop can be calculated $\tau = N\theta = 20$ ms. The input gain equals $\beta = 1$ and the feedback gain equals $\alpha = 0.01$. As illustrated in figure 4, the red dotted line represents the target output $y(t)$. The result of the simulation in the blue dots is pretty close to the target with the NMSE $= 1.3 \times 10^{-4}$, while the result of the experiment in the orange dots are fluctuate around target output with the NMSE $= 2.9 \times 10^{-2}$. A good experiment result in the optoelectronic RC system is reported in [27] with NMSE $= 1.5 \times 10^{-3}$. Another performance metric named SER is reported in [43] and the SER equals $0.02$ with the virtual node $N = 56$.

It can be seen that this MEMS hardware RC system can handle classification tasks.

3.2. NARMA task
The NARMA task is one of the most commonly used data sets to measure the predictive performance of the RC system. Owing to its relatively complex logical relations and controllable memory length, it has been studied by many scholars. In this prediction task, the RC system is trained to predict the output of complex systems, namely, $m$-order NARMA driven by white noise. The difficulty of the prediction increases with the increase $m$ because a large $m$ requires a higher memory capacity of the RC system. The $m$th NARMA task is given by the following recursive formula:

\[
y(n + 1) = 0.3y(n) + 0.05y(n) \left( \sum_{j=0}^{m-1} y(n - 1) \right)
+ 1.5u(n - m + 1)u(n) + 0.1
\]
Figure 7. Performance of hardware RC systems for the IMU action recognition task with the different number of virtual nodes. The blue line stands for the simulation result, and the orange line shows the experiment result.

Figure 8. Confusion matrix of experiment result for the IMU action recognition task. The number in the middle of each square represents the probability of success for each action. The abscissa represents the predicted target and the ordinate represents the actual predicted result. The color indicates the probability of correct classification. The squares on the diagonal indicate correct identification.

where \( y(n) \) is the target value of the task, \( m \) is the order of this data set, and \( u(t) \) stands for the white noise, which is the random input derived from a uniform distribution over the interval \((0, 0.5)\). The parameters are \( V_{dc} = 10 \text{ V}, N = 50, f_d = 253.000 \text{ Hz} \) in this experiment. In this task, data of length 1000 is used as the training set of the RC system, and the following 1000 data are used for testing. The performance metric used to evaluate NARMA is NMSE.

Figure 5 shows the result of the hardware RC system with \( m = 1, 2, 5, 10 \). The simulation results match the experimental results relatively well at the condition of \( m = 1 \) and \( m = 2 \). As the difficulty of prediction increases, the experimental results gradually deviate from the simulation results. The simulation result equals NMSE = 0.115 and the experiment result is NMSE = 0.169 with \( m = 10 \). The experiment result is similar to the value obtained using the optoelectronic RC system with the same number of virtual nodes. For instance, the NMSE value of 0.168 is reported in [27] with the virtual node \( N = 50 \), and the value of 0.19 is reported in [44]. Figure 6 shows the NMSE as a function of delay feedback gain and AC drive voltage in hardware RC system. Figures 6(a)–(d) represent the color gamut diagram with \( m = 1 \) to \( m = 10 \), respectively. In the dark blue area, the reservoir system can get reasonable results. A large region NMSE < 0.02 has been obtained in the NARMA1 data set. As the order increases, the blue area gradually decreases. This means that the more memory-demanding the dataset, the more difficult it will be to adjust the parameters. In addition, when ac voltage is around 1.4 V, better results can be obtained. This experiment shows that this MEMS hardware RC system can handle prediction tasks.

3.3. IMU action recognition task

After successfully proving that the hardware RC system with MEMS resonator has basic classification and prediction capabilities, the hardware RC system needs to process IMU data sets that have been proven to be feasible. In general, software algorithms are used to preprocess data in advance when processing classification tasks, which is often referred to as feature extraction. For example, the standard Lyon cochlear ear model used when processing the Ti-46 isolated word classification task, and the histogram of oriented gradient used when...
processing the KTH video classification task. To facilitate the design concept of integration of perception and computing, it is necessary to reduce the steps of software algorithm processing raw information. Therefore, we do not use feature extraction when processing IMU data sets.

In order to achieve the purpose of classification, we trained six linear classifiers for the six different actions in the data set, and each classifier was trained by the ridge regression method. If the result of the classifier classification is the corresponding action, the target output is 1. Otherwise, if the result of the classifier is the wrong action, the target output is 0. The results of the classifier are averaged over time, and the winner-takes-all method is used to get the final result. In the testing phase, the highest average classifier corresponding to the correct category is used to select the actual number. A performance metric called success rate (SR) is used to evaluate this classification task. As mentioned above, this novel action recognition task has a total of 480 sets of data. We randomly divided the 480 sets of data into 10 groups with 48 data in each group, including six actions of 8 people. They are then estimated using a ten-fold cross-validation procedure to minimize the fluctuations in results due to random selection between the training set and the test set. Ten parts are randomly selected, one for testing and nine for training.

In this task, the parameters are set as $f_d = 252,850 \text{ Hz}, \alpha = 0.4, \beta = 1, V_{dc} = 10 \text{ V}, V_{ac} = 1.4 \text{ V}$. It is very crucial to choose the number of virtual nodes $N$ when dealing with such complex classification tasks. Therefore, we first verify the influence of the number of virtual nodes on the SR. As shown in figure 7, the SR is greatly improved with the change of $N$. Parameter scan virtual node ranges from 50 to 450 in the simulation. The SR is greatly improved with $N$ increasing while $N \leq 200$. When the number of nodes is greater than 200, the SR will only float within a certain range and will not increase, which means that the adjustment ability of $N$ has reached its limit. Therefore, the number of nodes greater than 450 should not be attempted. The best SR of the simulation equals 92.3% with $N = 250$. The amount of data to be processed will increase significantly with the increase of the number of virtual nodes, which will affect the computing time and efficiency of the whole hardware RC system. In order to ensure the overall performance of the system, we take $N \leq 200$ as in actual experiments. As shown in figure 8, the best SR of the experiment equals 91.04% with $N = 200$. The confusion matrix can be more clearly observed for each action classification. The SR of these four movements is relatively high, including ‘circling’, ‘stepping’, ‘walking’, and ‘running’. It is relatively difficult to distinguish between ‘go upstairs’ and ‘go downstairs’, and the RC system occasionally classifies ‘go upstairs’ into ‘go downstairs’.

4. Conclusions

We design and process a novel IMU action recognition to testify our hardware RC system can deal with time-domain correlation classification tasks. First, we introduce the implementation of the hardware RC system based on MEMS resonators. After that, the method of processing the IMU data set is shown in detail and three basic ML classification algorithms are utilized to prove the feasibility of this novel data set. And then, we use a classification task and a prediction task, namely, the signal classification task and NAMRA task, to testify the performance of our hardware system. Finally, the IMU action recognition data set is successfully processed by the RC system and the SR of the experiment equals 91.04%. These results have significant implications with respect to hardware RC systems based on MEMS resonators to deal with time-domain correlation tasks. And it also paves the way to the actual sense and process of information simultaneously, in other words, integration of perception and calculation.

Funding

This research was partially supported by the National Key Research and Development Program of China under Grant 2018YFB2002300, the National Natural Science Foundation of China (Grant No. 61971399), and Key Research Program of Frontier Science (CAS, Grant No. ZDBS-LYJSC028).

Acknowledgments

The authors would like to thank the members of The State Key Laboratory of Transducer Technology for helpful discussions. And the authors would like to thank all the funds that have supported this work.

Data availability statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.
ORCID iDs

Tianyi Zheng https://orcid.org/0000-0003-4072-7047
Zhenxi Liu https://orcid.org/0000-0002-5072-5937

References

[1] LeCun Y, Bengio Y and Hinton G 2015 Deep learning Nature 521 436–44
[2] Krizhevsky A, Sutskever I and Hinton G E 2012 ImageNet classification with deep convolutional neural networks Commun. ACM 60 64–90
[3] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition Proc. IEEE conf. on Computer Vision and Pattern Recognition pp 770–8
[4] Tan M and Le Q 2019 EfficientNet: rethinking model scaling for convolutional neural networks Proc. Int. Conf. on Machine Learning pp 6105–14
[5] Merolla P A et al 2014 A million spiking-neuron integrated circuit with a scalable communication network and interface Science 345 686–73
[6] Benjamin B V et al 2014 Neurogrid: a mixed-analog–digital multichip system for large-scale neural simulations Proc. IEEE 102 699–716
[7] Tanaka G, Yamane T, Héroux J B, Nakane R, Kanazawa N, Takeda S, Numata H, Nakano D and Hirose A 2019 Recent advances in physical reservoir computing: a review Neural Netw. 115 100–23
[8] Jaeger H and Haas H 2004 Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication Science 304 78–80
[9] Bengio Y, Simard P and Frasconi P 1994 Learning long-term dependencies with gradient descent is difficult IEEE Trans. Neural Netw. 5 157–66
[10] Lukoševičius M and Jaeger H 2009 Reservoir computing approaches to recurrent neural network training Comput. Sci. Rev. 3 127–49
[11] Maass W, Natschlager T and Markram H 2002 Real-time computing without stable states: a new framework for neural computation based on perturbations Neural Comput. 14 2531–60
[12] Pearlmutter B A 1995 Gradient calculations for dynamic recurrent neural networks: a survey IEEE Trans. Neural Netw. 6 1212–28
[13] Appeltant L, Soriano M C, Van der Sande G, Danckaert J, Massar S, Dambre J, Schrauwen B, Mirasso C R and Fischer I 2011 Information processing using a single dynamical node as complex system Nat. Commun. 2 1–6
[14] Araujo F A et al 2020 Role of non-linear data processing on speech recognition task in the framework of reservoir computing Sci. Rep. 10 1–11
[15] Riou M et al 2019 Temporal pattern recognition with delayed-feedback spin-torque nano-oscillators Phys. Rev. App. 12 024049
[16] Nakajima K, Fujii K, Negoro M, Mitarai K and Kitagawa M 2019 Boosting computational power through spatial multiplexing in quantum reservoir computing Phys. Rev. Appl. 6 013077
[17] Sun L et al 2021 In-sensor reservoir computing for language learning via two-dimensional memristors Sci. Adv. 7 eabd1455
[18] Zhong Y, Tang J, Li X, Gao B, Qian H and Wu H 2021 Dynamic memristor-based reservoir computing for high-efficiency temporal signal processing Nat. Commun. 12 1–9
[19] Moon J, Ma W, Shin J H, Cai F, Lee S H and Lu W D 2019 Temporal data classification and forecasting using a memristor-based reservoir computing system Nat. Electron. 2 480–7
[20] Du C, Cai F, Zidan M A, Ma W, Lee S H and Lu W D 2017 Reservoir computing using dynamic memristors for temporal information processing Nat. Commun. 8 1–10
[21] Chang T, Jo S-H and Lu W 2020 Short-term memory to long-term memory transition in a nanoscale memristor ACS Nano 5 7696–769
[22] Govia L C G, Ribeill G J, Rowlands G E, Krovi H K and Ohki T A 2021 Quantum reservoir computing with a single nonlinear oscillator Phys. Rev. Res. 3 013077
[23] Ghosh S, Krisnanda T, Paterek T and Liew T C H 2021 Realising and compressing quantum circuits with quantum reservoir computing Commun. Phys. 4 1–7
[24] Nakajima K, Fujii K, Negoro M, Mitarai K and Kitagawa M 2019 Boosting computational power through spatial multiplexing in quantum reservoir computing Phys. Rev. Appl. 11 034021
[25] Larger L, Soriano M C, Brunner D, Appeltant L, Gutierrez J M, Pesquera L, Mirasso C R and Fischer I 2012 Photonic information processing beyond Turing: an opto-dynamical implementation of reservoir computing Opt. Express 20 3241–9
[26] Soriano M C, Ortín S, Brunner D, Larger L, Mirasso C R, Fischer I and Pesquera L 2013 Optoelectronic reservoir computing: tackling noise-induced performance degradation Opt. Express 21 12–20
[27] Hasan M H, Al-Ramini A, Abdelrahman E, Safari A and Alsalem F 2020 Colocalized sensing and intelligent computing in micro-sensors Sensors 20 6346
[28] Barazani B, Dion G, Morissette J-F, Beaudoin L and Sylvestre J 2020 Microfabricated neuroaccelerometer: integrating sensing and reservoir computing in MEMS J. Microelectromech. Syst. 29 338–47
[29] Dion G, Mejiaoui S and Sylvestre J 2018 Reservoir computing with a single delay–coupled non-linear mechanical oscillator J. Appl. Phys. 124 152132
[30] Papangelo A, Fontanella F, Grolet A, Ciavarella M and Hoffmann N 2019 Multistability and localization in forced cyclic symmetric structures modelled by weakly–coupled Duffing oscillators J. Sound Vib. 440 202–11
[31] Hassani M H, Al-Ramini A, Abdelrahman E, Safari A and Alsalem F 2020 Colocalized sensing and intelligent computing in micro-sensors Sensors 20 6346
[32] Barazani B, Dion G, Morissette J-F, Beaudoin L and Sylvestre J 2020 Microfabricated neuroaccelerometer: integrating sensing and reservoir computing in MEMS J. Microelectromech. Syst. 29 338–47
[33] Dion G, Mejiaoui S and Sylvestre J 2018 Reservoir computing with a single delay–coupled non-linear mechanical oscillator J. Appl. Phys. 124 152132
[34] Papangelo A, Fontanella F, Grolet A, Ciavarella M and Hoffmann N 2019 Multistability and localization in forced cyclic symmetric structures modelled by weakly–coupled Duffing oscillators J. Sound Vib. 440 202–11
[35] Hassani M H, Al-Ramini A, Abdelrahman E, Safari A and Alsalem F 2020 Colocalized sensing and intelligent computing in micro-sensors Sensors 20 6346
[36] Barazani B, Dion G, Morissette J-F, Beaudoin L and Sylvestre J 2020 Microfabricated neuroaccelerometer: integrating sensing and reservoir computing in MEMS J. Microelectromech. Syst. 29 338–47
[37] Dion G, Mejiaoui S and Sylvestre J 2018 Reservoir computing with a single delay–coupled non-linear mechanical oscillator J. Appl. Phys. 124 152132
[38] Papangelo A, Fontanella F, Grolet A, Ciavarella M and Hoffmann N 2019 Multistability and localization in forced cyclic symmetric structures modelled by weakly–coupled Duffing oscillators J. Sound Vib. 440 202–11
[39] Hassani M H, Al-Ramini A, Abdelrahman E, Safari A and Alsalem F 2020 Colocalized sensing and intelligent computing in micro-sensors Sensors 20 6346
[40] Barazani B, Dion G, Morissette J-F, Beaudoin L and Sylvestre J 2020 Microfabricated neuroaccelerometer: integrating sensing and reservoir computing in MEMS J. Microelectromech. Syst. 29 338–47
[36] Appeltant L, Van der Sande G, Danckaert J and Fischer I 2014 Constructing optimized binary masks for reservoir computing with delay systems Sci. Rep. 4 1–5
[37] Hou Y S, Xia G Q, Jayaprashath E, Yue D Z and Wu Z M 2020 Parallel information processing using a reservoir computing system based on mutually coupled semiconductor lasers Appl. Phys. B 126 1–9
[38] Chen Y, Yi L, Ke J, Yang Z, Yang Y, Huang L, Zhuge Q and Hu W 2019 Reservoir computing system with double optoelectronic feedback loops Opt. Express. 27 27431–40
[39] Antonik P, Marsal N, Brunner D and Rontani D 2019 Human action recognition with a large-scale brain-inspired photonic computer Nat. Mach. Intell. 1 530–7
[40] Schaetti N, Salomon M and Couturier R 2016 Echo state networks-based reservoir computing for MNIST handwritten digits recognition Proc. IEEE Int. Conf. on Computational Science and Engineering (CSE) and IEEE Int. Conf. on Embedded and Ubiquitous Computing (EUC) and 15th Int. Symp. on Distributed Computing and Applications for Business Engineering (DCABES) pp 484–91
[41] Vrigkas M, Nikou C and Kakadiaris I A 2015 A review of human activity recognition methods Front. Robot. AI 2 28
[42] Soriano M C, Ortín S, Keuninckx L, Appeltant L, Danckaert J, Pesquera L and Van der Sande G 2014 Delay-based reservoir computing: noise effects in a combined analog and digital implementation IEEE Trans. Neural Netw. Learn. Syst. 26 388–93
[43] McDonald N, Hollingsworth A, Beninati S, Lim C, Ji W, Shi C, Agiorgousis M L, Zhang S and Huang Z R 2020 Analysis of an ultra-short true time delay line optical reservoir computer J. Lightwave Technol. 38 3584–91
[44] Duport F, Akrout A, Smerieri A, Haelterman M and Massar S 2014 Analog input layer for optical reservoir computers Computer Science