Nonlinear Dynamics (2026) 114:422
https://doi.org/10.1007/s11071-026-12291-4

RESEARCH

Nonlinear spatiotemporal dynamics in magnetic dipole systems for physical reservoir computing

Liangshu He[1] · Yabin Jin[2,3] · Shulamit Edelstein[4] · Pedro David García[4] · Timon Rabczuk[5] · Xiaoying Zhuang[2,6] · Dani Torrent[7]

Received: 4 October 2025 / Revised: 16 January 2026 / Accepted: 18 January 2026

Abstract

To address the challenges of physical interpretability and scalability in artificial neural networks, physical reservoir computing (PRC) has emerged as a promising research direction in contemporary machine learning. PRC tackles these challenges by leveraging simple and efficient physical systems to emulate neural network dynamics. In this work, we develop a theoretical dynamic model of coupled magnetic dipoles driven by an external magnetic field and systematically analyze its dynamic behaviors. We employed this model to construct a standard physical reservoir computing architecture, and revealed the mechanism of magnetic dipole system for computing. Subsequently, the computational performance is evaluated using regression and classification benchmarks. Our results demonstrate that the models' rich nonlinear dynamic responses provide a robust foundation for achieving high accuracy in both chaotic time-series prediction and spoken digit recognition. The proposed contactless and nonlocal coupling architecture offers valuable theoretical insights for the advancement of PRC hardware development.

Keywords Reservoir computing · Spatiotemporal dynamic system · Chaotic time-series prediction · Spoken digit recognition

1 Introduction

Since the advent of artificial intelligence in the era of machine learning, artificial neural networks, which proposed by connectionism and inspired by the structure of the human brain, have emerged as a cutting-edge research direction [1, 2]. Through interdisciplinary integration, they have demonstrated remarkable vitality in applications such as image analysis [3], speech recognition [4], material discovery [5], and structural design [6–9]. However, neural networks are often considered data-driven "black boxes" whose internal processing lacks clear physical interpretability. In addition, as purely mathematical models, their computations depend heavily on conventional von Neumann architectures, resulting in inherent limitations in computational efficiency and scalability.

Against this background, a significant research direction has emerged: leveraging physical devices or systems to emulate neuronal behaviors and build physical neural networks [10], thereby facilitating the development of novel hardware platforms, collectively referred to as neuromorphic computing. A substantial body of work has explored the use of resistive arrays [11, 12], memristive devices [13],

✉ Yabin Jin
yabinjin@fudan.edu.cn

✉ Dani Torrent
dtorrent@uji.es

1 National Laboratory of Solid State Microstructures and Department of Materials Science and Engineering, Nanjing University, Nanjing 210093, China

2 Institute of Computational Mechanics × AI & College of Intelligent Robotics and Advanced Manufacturing, Fudan University, Shanghai 200433, China

3 Institute of Photonics, Department of Mathematics and Physics, Leibniz University Hannover, Hannover 30167, Germany

4 Insstituto de Ciencia de Materiales de Madrid (ICMM) Consejo Superior de Investigaciones Científicas (CSIC), Sor Juana Inés de la Cruz 3, Madrid 28049, Spain

5 Institute of Structural Mechanics, Bauhaus-Universitat Weimar, Weimar D-99423, Germany

6 Department of Geotechnical Engineering, College of Civil Engineering, Tongji University, Shanghai 200092, China

7 GROC, UJI, Institut de Noves Tecnologies de la Imatge (INIT), Universitat Jaume I, Castelló de la Plana 12071, Spain

Published online: 19 March 2026

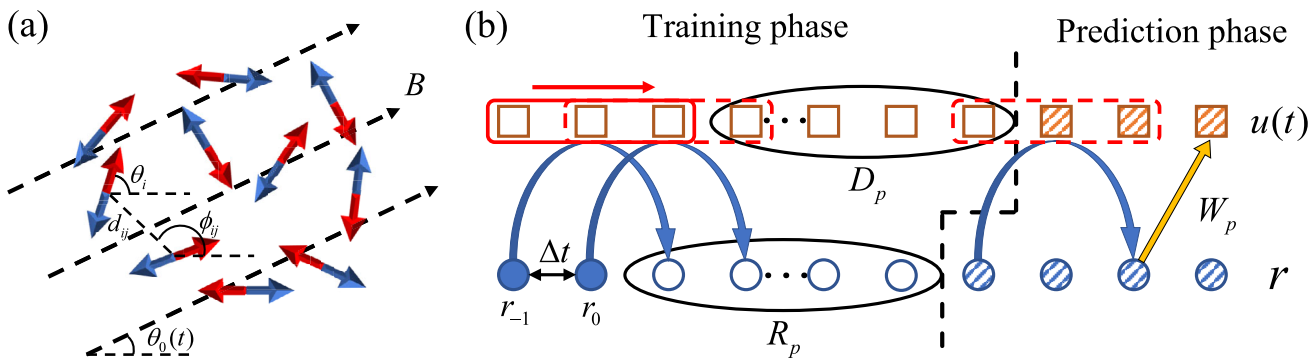

Fig. 1 Sketch of the magnetic dipole system for physical reservoir computing. (a) The nonlinear dynamical system consists of a cluster of randomly placed magnetic dipoles in a magnetic field. The blue and red arrows represent the two poles of the magnetic dipoles, while the black dashed arrows indicate the external magnetic field with a time-dependent angle $\theta_0(t)$. (b) Flowchart of the TSP numerical experiment. The time series and reservoir states are divided into training and prediction phases by the black dashed line. The red rectangles represent the computational core, which slides along the direction of the red arrow to iteratively compute the reservoir states. The blue arrows indicate the two-step iteration, and the yellow arrow represents the prediction of the time series from the reservoir states

spintronics components [14–16], and phase-change materials [17, 18] to implement such systems. Inspired by these efforts, metamaterials—renowned for their ability to manipulate complex wave phenomena—have emerged as a promising platform for neuromorphic computing, due to their inherent nonlinear modeling capabilities and their potential to effectively emulate neural network functions [19]. All-optical [20–22] and acoustic wave-based [23] diffraction deep neural networks have been demonstrated using optical and acoustic metasurfaces, respectively. Although these approaches effectively demonstrate the physical realization of feedforward neural networks, their functionality remains largely restricted to static tasks, such as image recognition and segmentation, and they are inherently unsuitable for processing temporal signals. In contrast, recurrent neural networks (RNNs), which incorporate loops or feedback mechanisms between network layers, possess inherent memory capabilities. This enables them to generate outputs based on both current inputs and prior internal states, allowing the execution of temporally dependent tasks, such as time-series prediction and speech processing. Nevertheless, the increased computational complexity of RNNs poses significant challenges for their physical implementation.

In this line of research, the underlying physical mechanisms have been progressively elucidated. In 2019, Hughes et al. [24] compared the discrete computational processes of wave equations, solved using the finite difference method, with the update rules of RNNs, thereby clarifying the mapping between the dynamics of wave systems and RNN computations. In 2022, Qu et al. [25] demonstrated that resonant structures can serve as fundamental building blocks for the construction of RNNs. By incorporating resonators with varying lifetimes, they enabled the system to exhibit both short-term and long-term memory capabilities. In 2023, Jiang et al. [26] further advanced this line of research by theoretically proposing a metamaterial-based RNN model. The model leverages locally coupled resonators within metamaterials to simulate spatiotemporal recursion, thereby achieving localized vibration energy and emulating the feedback mechanism inherent in RNN. Nevertheless, the training process in all the above works still relies on software-based computation, meaning the essential is only to simulate the trained weight matrix digitally.

As an emerging computational paradigm, reservoir computing has attracted significant attention due to its ability to operate without requiring training of the neural network [27, 28]. Specifically, the reservoir computing framework consists of three main components: an input layer that injects external signals into the system; a reservoir, typically a high-dimensional, dynamic, and nonlinear system, responsible for mapping and storing time-series information; and a trainable linear output layer. Traditionally, a RNN is employed as the reservoir, forming what is known as an Echo State Network (ESN). In this architecture, the reservoir remains fixed and only the output layer requires training-typically doing linear regression. Consequently, leveraging dynamical systems to replace RNN offers a natural pathway for building physical reservoir computing frameworks. Appeltant et al. [29] employed a single virtual time node with delayed feedback as a reservoir to demonstrate its capability for time-series prediction and speech recognition. Building on this concept, Liang et al. [30] proposed a rotating reservoir composed of multiple virtual time nodes, achieving a reservoir computing model with enhanced performance due to improved nonlinearity. Hauser et al. [31] proposed a mass-spring reservoir model, in which adjacent mass blocks are randomly connected by nonlinear springs. In this model, the input signal corresponds to an external force transmitted

through the spring network, while the output signal is defined as a linear combination of the spring lengths. Momeni et al. [32] proposed enhancing the system's nonlinear processing capabilities by employing a time-periodically modulated electromagnetic wave system. They demonstrated this concept using a dielectric plate with a slowly time-varying refractive index, combined with a conventional random layer, to construct a physical reservoir capable of predicting chaotic time series. In addition, it also includes memristive devices [33–35], photonic devices [36, 37], among others. Indeed, identifying dynamical systems that are simple, generalizable, interpretable, and easy to implement remains a central challenge in this field and continues to be a focus of ongoing research [38].

In this paper, we propose a physical reservoir computing (PRC) model based on a cluster of spatiotemporal magnetic dipoles randomly distributed within an external magnetic field. The system is driven by variations in the orientation of the external magnetic field, which serve as inputs to induce rotational motion in the magnetic dipoles. The interplay of mutual coupling and hysteresis among the dipoles gives rise to rich and nonlinear dynamic behaviors. We evaluated the proposed model on time-series prediction (TSP) and time-series classification (TSC) tasks using the Mackey—Glass time series and spoken digits, respectively. The results demonstrate that the system's nonlinear richness effectively endows the constructed PRC with both regression and classification capabilities. Furthermore, the proposed noncontact, nonlocally coupled dynamical system is relatively simple and straightforward to implement.

2 Theory and Equations

Here, we demonstrate how the physical system can function as an analog of an ESN. Consider a dynamical system comprising a cluster of N magnetic dipoles randomly distributed in two-dimensional space to form the PRC model, as shown in Fig. 1(a). When driven by an external magnetic field, the dipoles, with fixed relative positions, exhibit complex rotational motion. It should be noted that the concept of magnetic dipoles here is effective dipolar particles with field-dependent magnetization instead of the standard single-domain or permanent magnetic dipoles. The equation of motion for the i-th magnetic dipole can be given by a second-order ordinary differential equation as

$$\ddot{\theta}_i = -\alpha_i \sin\left[\theta_i - \theta_0(t)\right] - \gamma_i \dot{\theta}_i - \sum_{j \neq i} \beta_{ij}\, g(\theta_i, \theta_j), \tag{1}$$

where $\alpha_i = m_i B / I_i$ is the torque coefficient caused by the external magnetic field, $\gamma_i = c_i / I_i$ is the damping coefficient, $\beta_{ij} = \mu_0 m_i m_j / (4\pi I_i d_{ij}^3)$ is the nonlocal coupling strength between dipoles, and the coupling form is [39]

$$g(\theta_i, \theta_j) = \sin(\theta_i - \theta_j) - 3\cos(\theta_i - \phi_{ij}) \sin(\theta_j - \phi_{ij}). \tag{2}$$

The physical parameters are defined as follows: θ_i denotes the rotational angle; I_i is the moment of inertia; B represents the external magnetic field; $\theta_0(t)$ denotes the time-dependent direction of the magnetic field, which is used to model the input time series; c_i is the rotational damping coefficient; $\mu_0 = 4\pi \times 10^{-7}$ N/A^2 is the permeability of free space; and d_{ij} and ϕ_{ij} represent the distance and relative orientation angle between dipoles i and j, respectively.When subjected to an external magnetic field, the magnetic moment of the dipoles, denoted by m_i, exhibit a magnetic hysteresis effect, which can be characterized by the following equation

$$\tau_m \dot{m}_i = -m_i + m_s \tanh\left[\frac{B\cos(\theta_i - \theta_0(t))}{B_0}\right], \tag{3}$$

where τ_m denotes the magnetic relaxation time, m_s represents the saturation magnetic moment, and B_0 is a scaling coefficient of the magnetic field. This first-order hysteretic response model represents a simplified form derived from the Jiles–Atherton model [40, 41]; an approximate derivation is provided in Appendix A. To check the PRC model without loss of generality, we fix the parameters as follows for all models considered in this work: $I_i = 1$, $c_i = 1$, $B = 1$, $B_0 = 1.5$, $m_s = 1$, $\tau_m = 1$, d_{ij} randomly in $[0, 1]$, and ϕ_{ij} randomly in $[0, 2\pi]$. It is worth noting that $d_{ji} = d_{ij}$, $\phi_{ji} = (\phi_{ij} + \pi) \bmod 2\pi$, $d_{ii} = 0$, and $\phi_{ii} = 0$. The system's time is normalized to characteristic time $t_0 = I_i / c_i$ and the normalized angular velocity is $\omega_0 = c_i / I_i$.

In this dynamical system, the magnetic dipoles are typically treated as analogs of neurons, and the motion states of all dipoles—including rotational angle, angular velocity, and magnetic moment—constitute the reservoir. Since these states cannot change instantaneously, the current state naturally depends on the previous one. It is worth noting that, in principle, we can choose one of them as the reservoir. However, its nonlinearity is relatively weak and insufficient to handle chaotic time series. Combining all three parameters provides a more powerful internal computational capability. The system offers the advantages of easily adjustable coupling strength through dipole positioning and the benefits of contactless interactions. The coupling between dipoles, combined with hysteresis-induced coupling, endows the system with rich nonlinear dynamics.

To numerically solve this system, we here employ a modified fourth-order Runge–Kutta method. By performing an

order reduction of Eq. (1), we obtain

$$\begin{cases} \dot{\theta}_i = \omega_i \\ \dot{\omega}_i = -\alpha_i \sin\left[\theta_i - \theta_0(t)\right] - \gamma_i \omega_i - \sum\limits_{j \neq i} \beta_{ij}\, g(\theta_i, \theta_j) \\ \dot{m}_i = -\dfrac{m_i}{\tau_m} + \dfrac{m_s}{\tau_m} \tanh\left[\dfrac{B\cos\left(\theta_i - \theta_0(t)\right)}{B_0}\right], \end{cases} \tag{4}$$

where ω_i denotes the angular velocity. By combining the above equations into vector form, we obtain $\dot{r} = f(t, r)$, where $r = (\theta_1, ..., \theta_N, \omega_1, ..., \omega_N, m_1, ..., m_N)^T$ constitutes the reservoir. Considering the actual measurement, each magnetic dipole is rigidly attached to its dedicated rotation axis, allowing for the direct installation of high-resolution angle encoders (e.g., optical angle encoders [42]) to precisely track rotation angle in real time. The angular velocity can be obtained through simple numerical differentiation of the measured rotational angle. Similarly, the magnetic moment can be inferred indirectly using Eq. (3) based on the known external magnetic field and the measured rotation angle. Alternatively, highly sensitive magnetometers (e.g., Hall detector probes [43]) can be employed for direct measurement. Assume that we discrete a continue time series $u(t)$ and apply $\theta_0(t) = u(t)$ with M time steps as the input time series for training. In order to satisfy the requirements of Runge–Kutta method for computing slopes at the midpoint of each time interval, the iteration should be modified as a two-step format, as follows

$$r_i = r_{i-2} + (k_1 + 2k_2 + 2k_3 + k_4)/6, \tag{5}$$

where

$$k_1 = hf(t_{i-2}, r_{i-2}) \tag{6}$$

$$k_2 = hf(t_{i-1}, r_{i-2} + \frac{k_1}{2}) \tag{7}$$

$$k_3 = hf(t_{i-1}, r_{i-2} + \frac{k_2}{2}) \tag{8}$$

$$k_4 = hf(t_i, r_{i-2} + k_3) \tag{9}$$

and $h = 2\Delta t$ is the iteration step. We define the time interval between adjacent integration points as $\Delta t = 0.5t_0$. As demonstrations, we now consider employing this system to perform two fundamental tasks via numerical experiment, namely, time-series prediction and time-series classification.

For the TSP, which is a regression task, the system is first driven by the known time series for training, and the subsequent series is then obtained via autonomous prediction, as illustrated in Fig. 1(b). The reservoir state collection matrix is given by

$$R_p = [r_i]_{i=1}^{M-3}. \tag{10}$$

The solid circles in Fig. 1(b) represent the initial conditions, where r_{-1} is initialized with random $\theta_i \in [-\pi, \pi]$, $\omega_i = 0$, and $m_i = 0.1$; while on this basis, we directly let $r_0 = r_{-1}$, as we confirm that owing to the system's highly rich nonlinear response and the use of a small step size, the state differences between two successive inputs remain relatively small. Once the system is established, the reservoir remains fixed, and training involves only modifying the readout map. The readout map is treated as a weighted combination of the reservoir states and is typically trained via linear regression. To implement this, the desired output collection matrix must be defined; for the TSP task, it corresponds to the time series shifted one step ahead, as follows

$$D_p = [u_i]_{i=2}^{M-2}. \tag{11}$$

The output weight matrix can now be obtained by performing a ridge regression,

$$W_p = (R_p^T R_p + \lambda I)^{-1} R_p^T D_p, \tag{12}$$

where I denotes the identity matrix and λ is a non-negative regularization coefficient. Furthermore, the prediction phase is given by

$$\hat{u}_{i+1} = W_p r_i, \tag{13}$$

where $i \geq M - 2$. In the prediction phase, the output at step i serves as the predicted value for step $i + 1$ and is subsequently fed back into the reservoir as input, enabling the system to autonomously generate predictions in a closed-loop manner. Consequently, this framework can be employed to reconstruct missing signals.

For the TSC, which is a classification task, the system is driven by the time series to recognize the corresponding label. Suppose we have a dataset comprising multiple time series samples, of which typically 80% serves as the training set and 20% as the test set. Each sample is represented as $(u_i(t), y_i)$, where $u_i(t)$ denotes the signal of the i-th sample, and y_i is the corresponding label. Unlike the TSP, the system ceases evolving once the input time series is exhausted in the TSC task. When a single signal is injected into the system as the angle of the external magnetic field, the system evolves until the end, and the final reservoir state, which encodes the information of the sample, is obtained. It should be noted that, although the original waveform contains all feature information, real speech datasets often present challenges such as high dimensionality and excessive sensitivity to amplitude variations, noise, and other factors, which can reduce classification accuracy. Therefore, the original waveform is typically preprocessed for feature extraction prior to being input into the system. During the training phase, all samples in the training set are sequentially injected into the

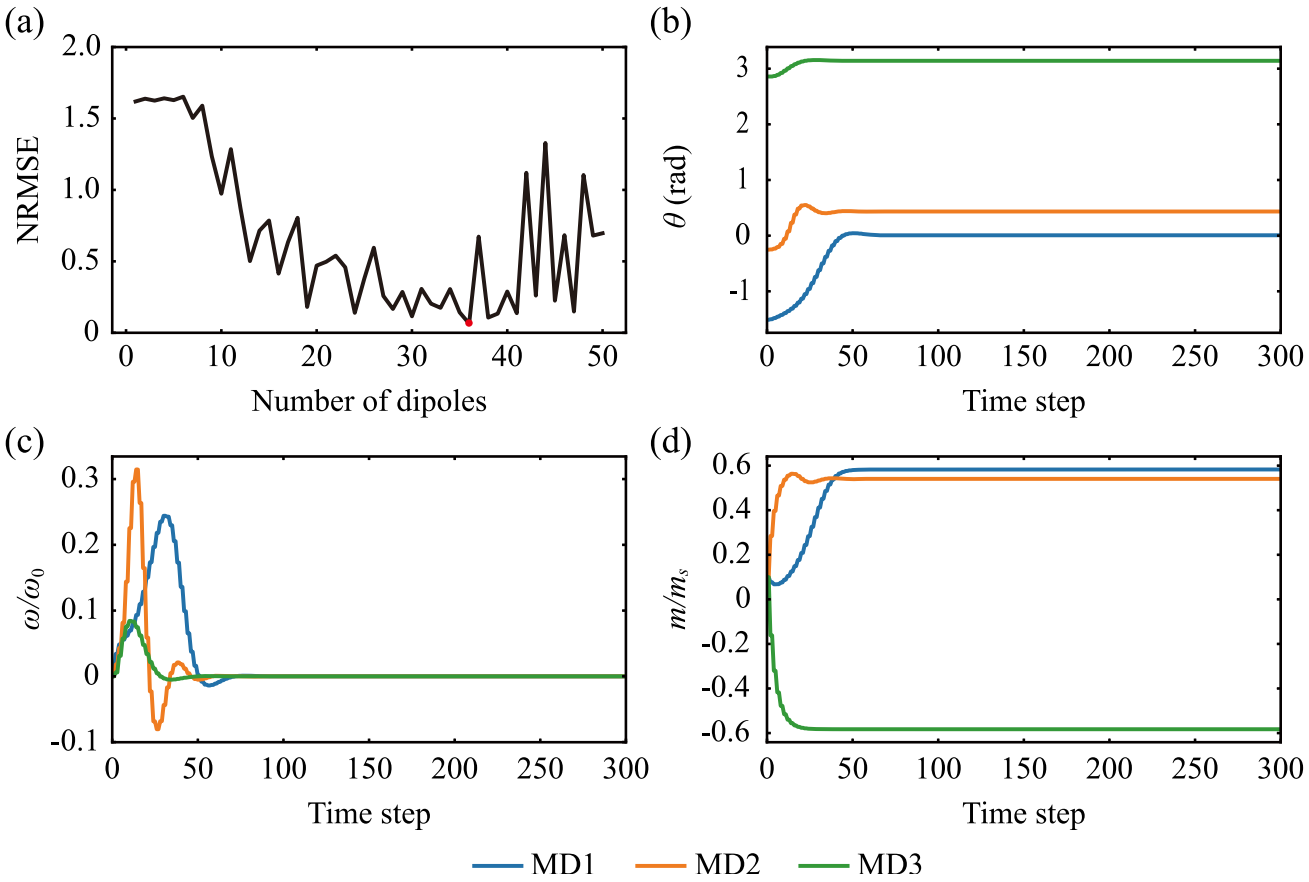

Fig. 2 Mechanism analysis of the PRC model on the TSP task. (a) NRMSE as a function of the number of dipoles when TS1 is used as the model input. The red dot ($N = 36$) indicates the optimal configuration. The temporal responses of (b) rotation angle, (c) angular velocity and (d) magnetic moment when the angle of external magnetic field is set to 0

system, and the corresponding final states collectively form the reservoir state collection matrix, as follows

$$R_c = [r_i]_{i=1}^{M}, \tag{14}$$

where M represents the number of samples in the training sets. The desired output for this task are the labels y_i. In classification tasks, one-hot encoding can enhance the algorithm's ability to handle discrete features, with the output representing the estimated probability for each class. Consequently, we transform the labels into sparse vectors Y_i in one-hot encoded form and construct the desired output collection matrix as follows

$$D_c = [Y_i]_{i=1}^{M}. \tag{15}$$

Similarly, the output weight matrix is given by

$$W_c = (R_c^T R_c + \lambda I)^{-1} R_c^T D_c. \tag{16}$$

After training, we test the model using samples in test sets. The prediction is given by

$$\hat{y}_i = \arg\max(W_c r_i). \tag{17}$$

This implies that the predicted label corresponds to the index of the maximum probability. Based on this, classification accuracy is typically assessed by constructing a confusion matrix using all prediction results from the test set.

3 Results and Discussions

3.1 Prediction of Mackey–Glass Time Series

In the present subsection, we take the widely studied Mackey–Glass time series as an example to evaluate the TSP ability of the proposed PRC model. The Mackey–Glass delay dif-

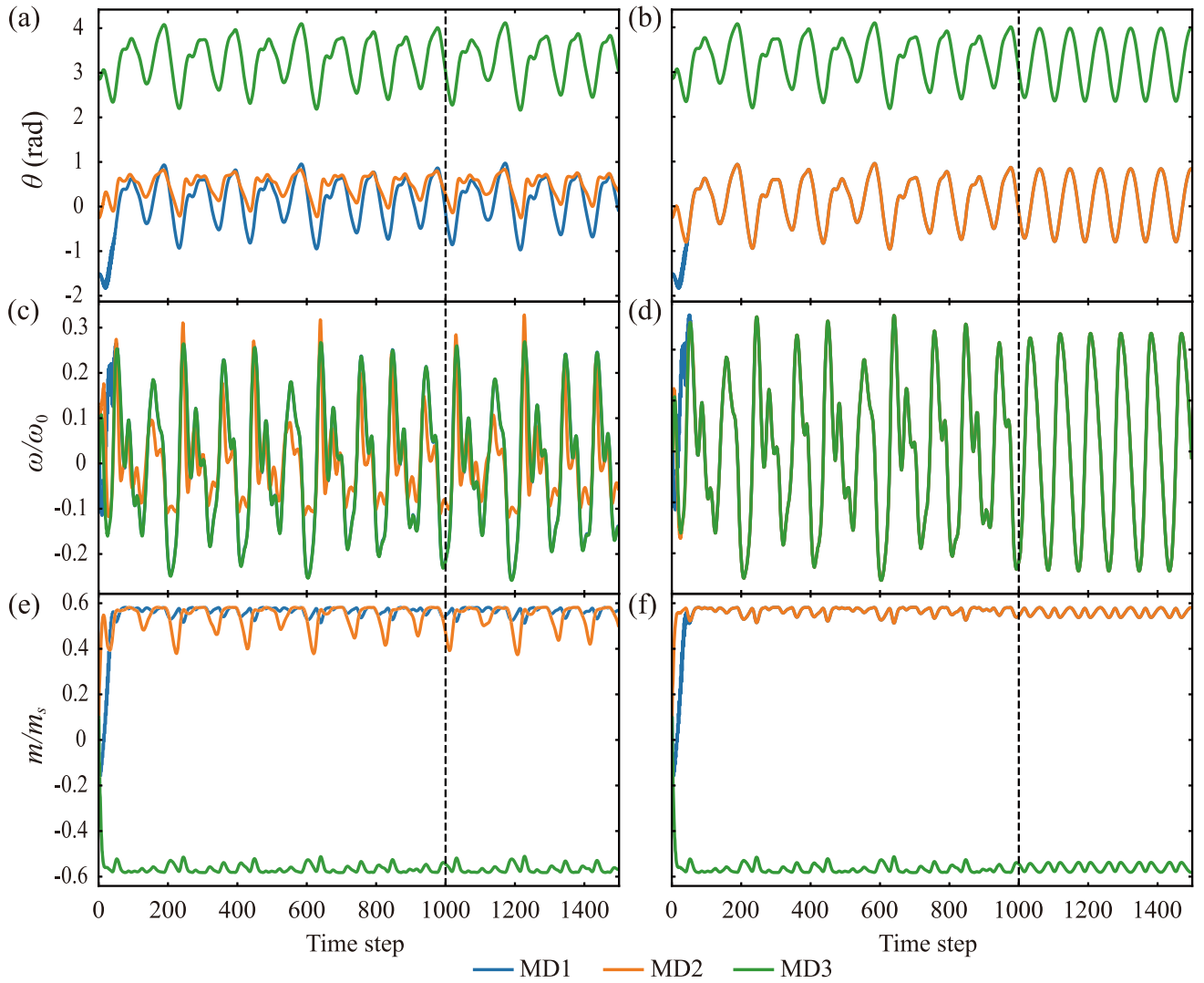

Fig. 3 Temporal responses of (a-b) rotation angle, (c-d) angular velocity, and (e-f) magnetic moment by taking TS1 as the training data, where we have considered coupling between magnetic dipoles in the left panel, but without coupling in the right panel. The vertical dashed line indicates the boundary between the training phase and prediction phases

ferential equation is defined as follows

$$\frac{du(t)}{dt} = \frac{au(t-\tau)}{1+[u(t-\tau)]^n} - bu(t), \tag{18}$$

where we set $a = 0.2$, $b = 0.1$, $n = 10$, and the time delay parameter $\tau = 17$, 19, 21 and 23 for different degrees of chaotic behavior. It should be noted that the signal exhibits periodic behavior when $\tau \leq 16$. However, once τ exceeds this threshold, the positive Lyapunov exponent leads to exponential growth for the separation of close trajectories. Therefore, chaotic time-series prediction is inherently challenging due to error accumulation in iterative predictions, which leads to exponential divergence from the ground truth over time. Furthermore, as τ increases among the values set above, the degree of chaos intensifies, making accurate prediction increasingly difficult. More details about the Lyapunov exponent are provided in the Note S1 of Supplementary Information.

We generate four sufficiently long Mackey-Glass time series with an initial value of $u_0 = 1.2$, time step of 0.5, but with different time delays to give different chaotic levels: time series 1 (TS1) with $\tau = 17$, time series 2 (TS2) with $\tau = 19$, time series 3 (TS3) with $\tau = 21$, and time series 4 (TS4) with $\tau = 23$. All series are linearly normalized to the range $[-1, 1]$. To enhance the generalization and robustness of the model, Gaussian noise with a mean of 0 and variance of 0.01 is added to each time series. We use $M = 1000$ time steps as training data, and the subsequent 500 time steps as test data.

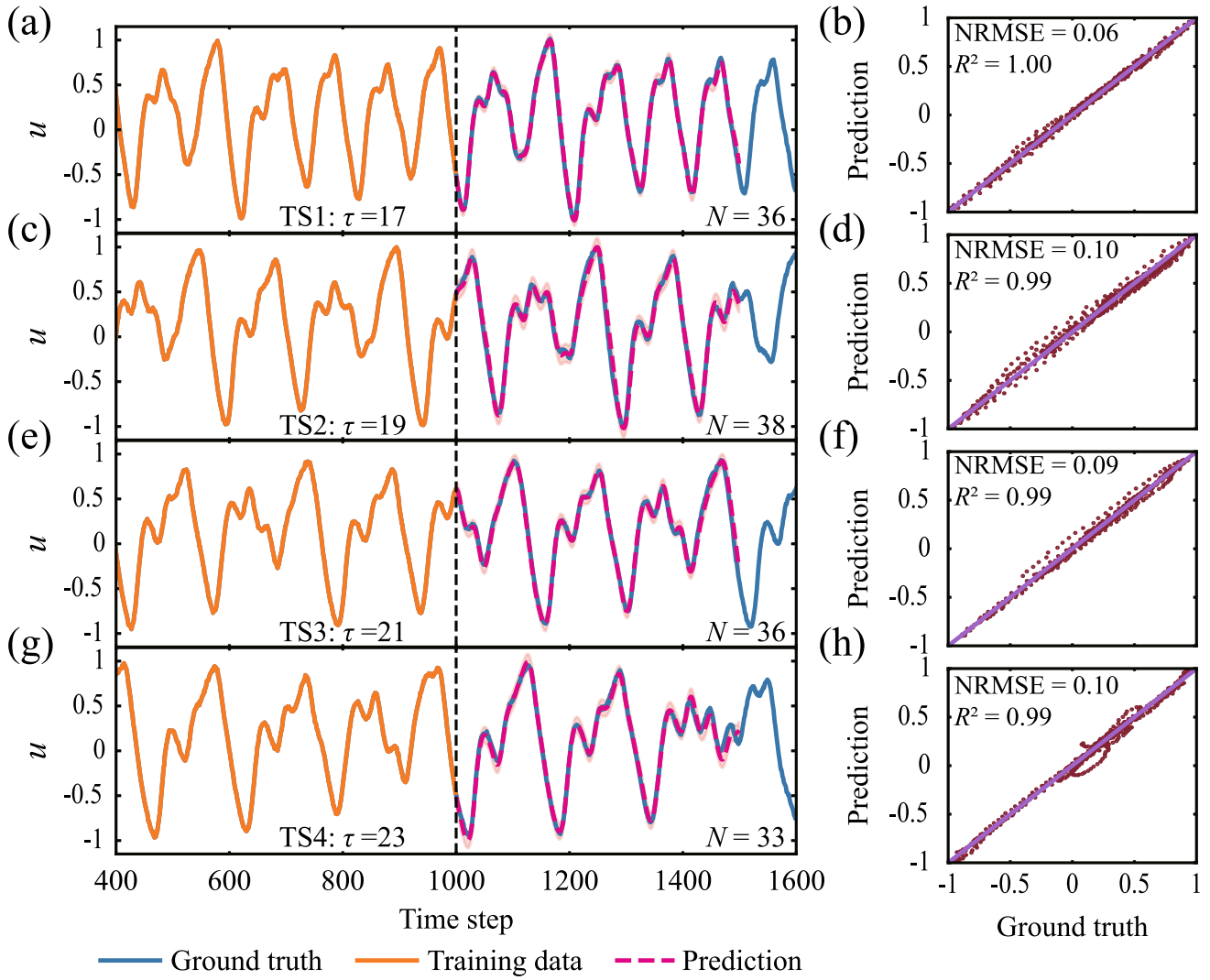

Fig. 4 Autonomous prediction results driven by Mackey–Glass time series with varying degrees of chaos. (a-b) TS1, (c-d) TS2, (e-f) TS3, and (g-h) TS4. Left panel: comparison between the ground truth and the predictions of the PRC model over 500 test time steps, with the pink shaded areas indicating the 95% confidence intervals. Right panel: scatter plots of ground truth versus prediction, with the purple diagonal lines representing perfect prediction

To evaluate performance, we employ the normalized root mean square error (NRMSE), which is defined as follows

$$\mathrm{NRMSE} = \frac{1}{\sigma_u}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(u_i - \hat{u_i})^2}, \tag{19}$$

where n denotes the number of sample points in the prediction phase, σ_u is the standard deviation, u_i and $\hat{u_i}$ represent the true and predicted values, respectively. As a case study, Fig. 2(a) shows how the NRMSE varies with the number of dipoles N (reservoir size $3N$) when driven by TS1. The randomness involved in generating reservoirs of varying sizes—manifested in d_{ij} and ϕ_{ij}—leads to fluctuations in NRMSE. Nevertheless, the NRMSE exhibits an overall convex relationship with reservoir size, initially decreasing until it reaches an optimum at $N = 36$, after which it increases with further expansion. This behavior can be attributed to the fact that an appropriately sized set of magnetic dipoles forms a complex and richly coupled dynamical system, enabling more accurate approximation of the target time series. Conversely, an undersized reservoir lacks sufficient nonlinearity for effective signal reconstruction, while an oversized configuration may induce overfitting.

To elucidate the cooperative dynamics among the magnetic dipoles, we first fix the external magnetic field angle at 0, based on the reservoir configuration with $N = 36$, and randomly select three magnetic dipoles as representative subjects—MD1, MD2, and MD3—without loss of

generality. Figures 2(b–d) show the temporal responses of the rotation angle, angular velocity, and magnetic moment, respectively. As illustrated, all dipoles are initially at rest and possess identical magnetic moments; however, randomized initial rotation angles and spatial distributions trigger system motion. The system exhibits a typical transient response before reaching the rest state again. Due to rotational damping, the system eventually ceases motion, and each magnetic dipole maintains static equilibrium with specific rotation angles and magnetic moments under the combined influence of the external magnetic field and interactions with other dipoles. When the external input varies continuously at each time step, the resulting responses become highly rich due to superposition effects. More details about the phase space are provided in the Note S1 of Supplementary Information.

Figures 3 presents the temporal responses of the magnetic rotors (MD1, MD2, and MD3) when TS1 is employed as input, corresponding to the case of fixed input shown in Fig. 2. From the left panel, it can be seen that when nonlocal coupling among the magnetic dipoles is considered, the structure resembles an artificial neural network, where each dipole acts as a neuron evolving over time and contributing individually to the overall system output. The three parameters together constitute a rich internal state of the reservoir, fully characterizing its dynamic behavior. During the prediction phase, the response curves largely follow the patterns observed during training, indicating that the system has successfully captured the characteristics of the training data. As a result, even in the absence of ground-truth input for correction, the system is able to sustain the correct dynamical behavior by feeding back its own predictions as input. To further analyze the role of nonlocal coupling, we remove the interactions among dipoles (i.e., let $g(\theta_i, \theta_j) = 0$), and the resulting responses are shown in the right panel of Fig. 3. In the absence of coupling, the only differences among dipoles stem from the random initialization of rotation angles. However, this effect quickly vanishes under the influence of the external magnetic field, and the system eventually converges to a synchronized response driven by the input. As observed from the right panel, during training, the angular velocities of all dipoles become nearly identical after being corrected by the external field at the initial stage. The random initialization of rotation angles introduces a certain degree of freedom, allowing the dipoles to evolve with different angles; nevertheless, their dynamical behaviors remain consistent. According to Eq. (4), the magnetic moment depends on the rotation angle; thus, while their values may differ numerically, their temporal evolution is identical. This analysis demonstrates that the motions of all dipoles are nearly indistinguishable, causing the high-dimensional state vector $r = (\theta_1, ..., \theta_N, \omega_1, ..., \omega_N, m_1, ..., m_N)^T$ to contain a large amount of redundant information. Consequently, the effective dimensionality is insufficient, making it difficult to capture nonlinear features of the input, which significantly reduces the computational power of the system. As a result, once entering the prediction phase, the three parameters exhibit almost periodic behavior, in stark contrast with the training stage, indicating model failure. According to Eq. (4), after inter-dipole coupling is removed, the interaction between dipoles and the external field through the sine term and hysteresis function still provides a weak degree of nonlinearity, preventing the system from total collapse and ensuring that the response amplitudes remain within a reasonable range during prediction. A comparison of the left and right panels highlights that nonlinear coupling among magnetic dipoles plays a key role in breaking system homogeneity, enhancing the effective dimensionality of the reservoir's internal state, and thereby providing sufficient prediction capacity. Although we have seen that the system evolves in a reasonable way, the numerical stability is worth to analyzing and we provide in the Appendix B. More details about the physics of prediction capacity are provided in the Note S2 and S3 of Supplementary Information.

We further examine the prediction performance. Figure 4 presents the autonomous prediction results driven by four Mackey–Glass time series with varying degrees of chaos. From the left panels, it can be seen that after training over the segment indicated by the orange solid line, the model enters the prediction phase, during which no external input is provided. Instead, the next time-step predictions are fed back into the model, allowing the system to evolve autonomously and perform step-by-step forecasting. During prediction, the results indicated by the pink dashed line align almost perfectly with the ground truth shown by the blue solid line. Therefore, with an appropriately configured reservoir size, the model achieves excellent prediction performance across all four time series, including TS4 with a time-delay parameter of $\tau = 23$, which still demonstrates a remarkably high level of predictive accuracy. The 95% confidence interval indicates that the discrepancies between prediction and ground truth are primarily concentrated at sharp turning points in the time series. The right panel shows scatter plots of prediction versus ground truth, from which it can be seen that the data points are tightly and evenly distributed around the diagonal, indicating high prediction accuracy and the absence of systematic bias. The few data points that deviate from the diagonal mainly appear near the end of the prediction horizon, reflecting the inherent error accumulation in long-term iterative forecasting. To complement this qualitative analysis, we also report the NRMSE and R^2 as quantitative performance metrics. The R^2 is a commonly used indicator for evaluating the goodness of fit and is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(u_i - \hat{u_i})^2}{\sum_{i=1}^{n}(u_i - \bar{u})^2}, \tag{20}$$

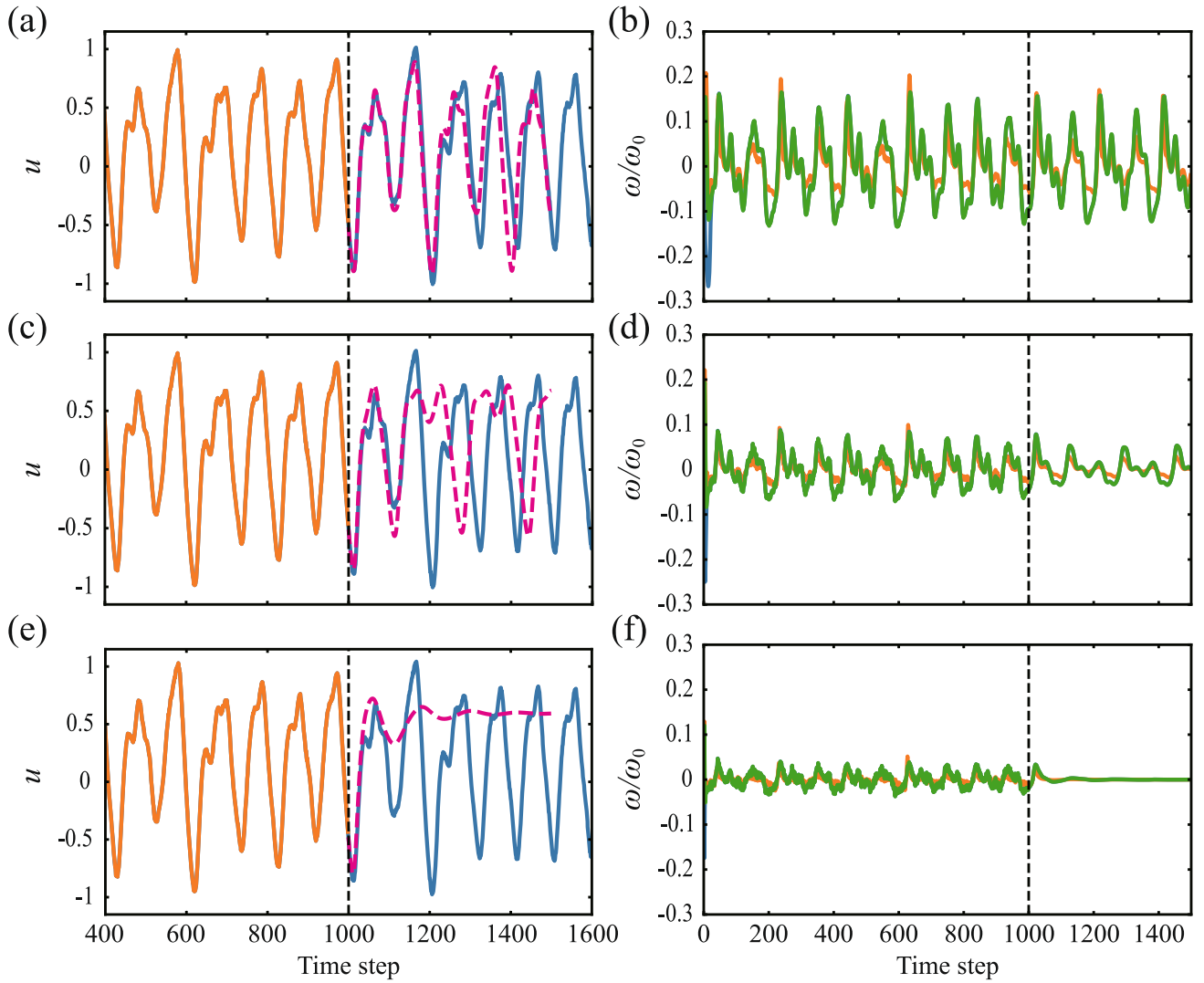

Fig. 5 Effects of different readout step sizes on the autonomous prediction and temporal responses of angular velocity. (a-b) $\beta = 2$, (c-d) $\beta = 4$, (e-f) $\beta = 8$. Left panel: the autonomous prediction. Right panel: the temporal responses of angular velocity. The legends is the same as Figs. 3 and 4

where $\bar{u}$ denotes the mean of the ground truth. As shown in the right panels, the NRMSE are all close to 0, while the corresponding R^2 are consistently close to 1. These two metrics confirm that the proposed PRC model accurately captures the fine details of the time series, effectively reconstructing subsequent sequences. Compared with existing report [32], the results obtained in the Mackey–Glass time series prediction demonstrate that our model exhibits highly competitive computational performance, in which we further extends prior work by predicting longer sequences and sequences with higher degrees of chaos. A concise benchmark comparison with other reservoir systems is provided in Note S4 of Supplementary Information.

The above analysis is based on a stepwise driving strategy, in which new data are injected at every time step to maintain the continuous evolution of the system. This approach enables the system to gradually accumulate nonlinear effects induced by the successive inputs, thereby generating a rich dynamic response. However, it is also worth investigating an alternative driving scheme: after data are injected at a given time step, the system evolves under this input for β time steps, after which its final state is read out before the next data injection. Compared with the stepwise strategy, this approach allows the system to unfold its internal dynamics over a longer time scale. The left panel of Fig. 5 presents the prediction results under different readout step sizes. It can be observed that as the readout step size β increases, the overall prediction accuracy decreases significantly, and the length of the accurately predictable horizon becomes notably shorter. This phenomenon indicates that the system's prediction capacity gradually deteriorates with increasing β. When the system is allowed to evolve freely for an extended period,

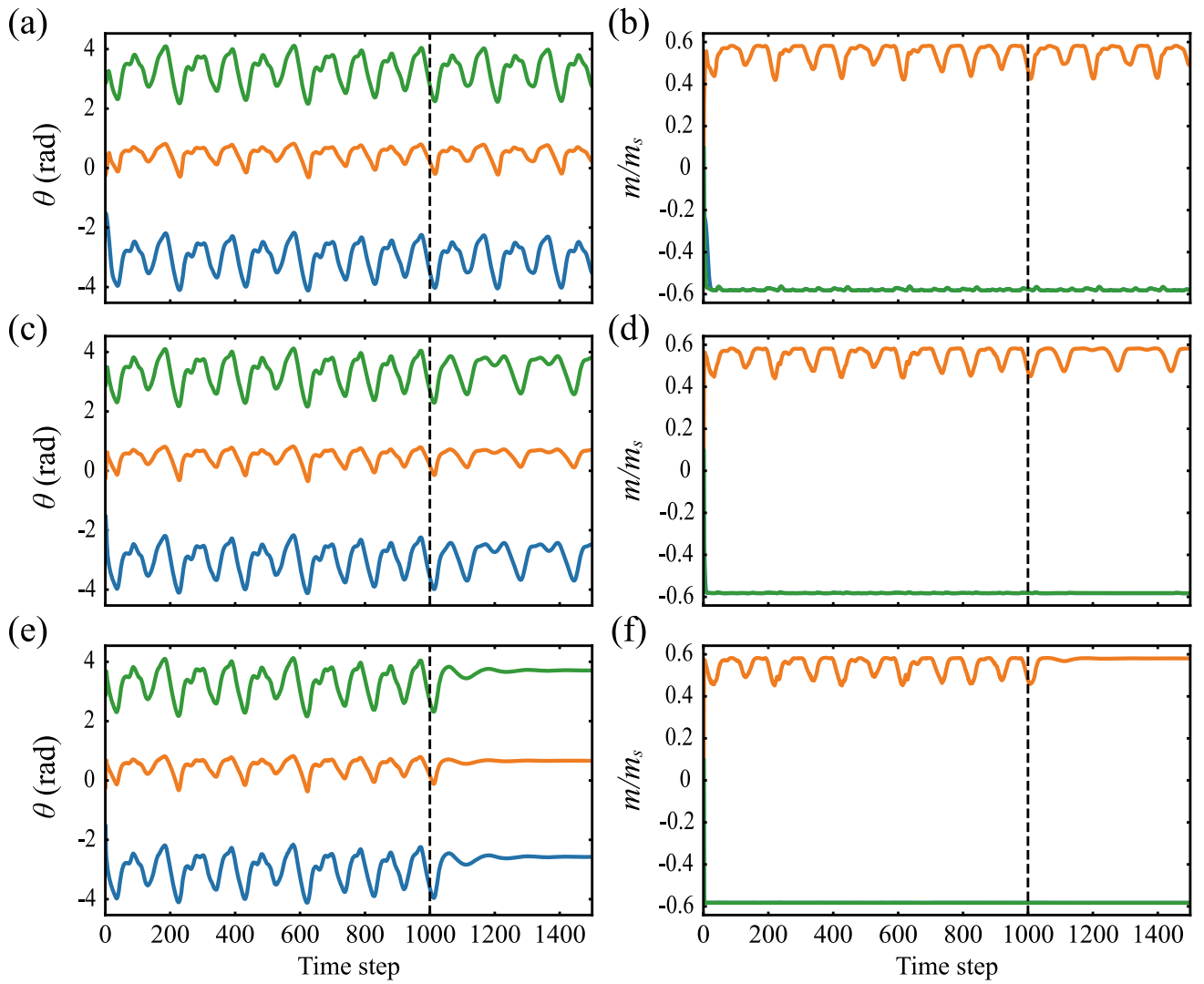

Fig. 6 Effects of different readout step sizes on the temporal responses of rotation angle and magnetic moment. (a-b) $\beta = 2$, (c-d) $\beta = 4$, (e-f) $\beta = 8$. Left panel: the temporal responses of rotation angle. Right panel: the temporal responses of magnetic moment

most of the information from the previous input is lost before the arrival of the next one. As shown in the right panel of the temporal responses of angular velocity, the readout angular velocity decreases progressively with increasing β. This indicates that under the influence of damping, the system loses a significant portion of its velocity during free evolution, such that by the time the next input arrives, most of the initial momentum has already dissipated. After entering the prediction phase, the absence of real data injection to correct the system state causes the system, under self-prediction driving, to maintain dynamics similar to those observed during training only for a short period. Moreover, as the readout step size β increases, the duration over which this consistency can be sustained becomes progressively shorter. When β approaches infinity, the system dynamics degenerate into the fixed-time driving scenario shown in Fig. 2. In this extreme case, the system rapidly decays and returns to a resting state after receiving a single input.

Figure 6 further illustrates the temporal responses of the rotation angle and magnetic moment. From the left panel, it can be seen that during the training phase, the temporal responses of angular velocity are identical for different β, as their values are determined by the injected angle for external magnetic field. Consequently, larger readout step sizes allow the system to reach a given rotation angle at a slower angular velocity. Since the magnetic moment depends on the rotation angle, its temporal response follows a pattern similar to that of the rotation angle. However, when β increases, according to Eq. (4), the rotation angles of the magnetic dipoles tend to align with the angle of external magnetic field, causing the cosine term to approach 1 and the magnetic moments to reach a stable, saturated state, as observed for dipoles MD1

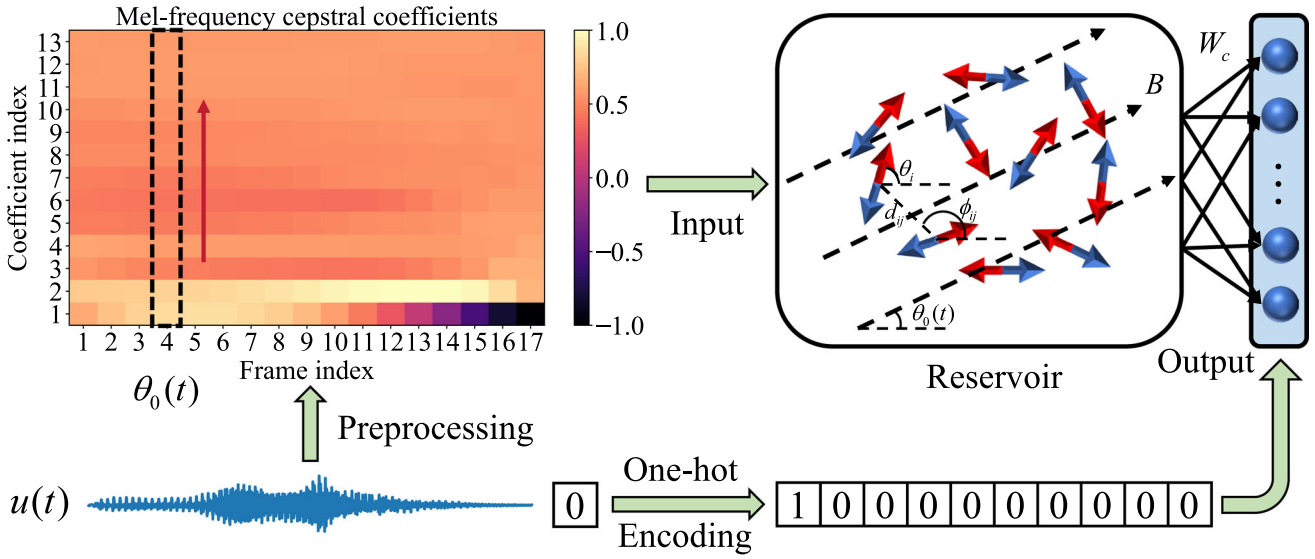

Fig. 7 Numerical experiment framework for spoken digit recognition. The blue curve represents a voice signal, which is subsequently transformed into Mel-frequency cepstral coefficients before being input into the reservoir. The solid black box denotes the physical reservoir and the arrows indicate the output to the readout layer. The corresponding labels of the voice signals are one-hot encoded and serve as the target readout

and MD3. If a dipole's rotation angle exhibits a dynamic deviation from the angle of external field, the fluctuations of the cosine term induce dynamic variations in its magnetic moment, as seen for dipole MD2. During the prediction phase, the system gradually comes to a stop, with the rotation angles becoming constant (Fig. 6(e)) and the cosine term approaching 1, resulting in all magnetic dipoles eventually reaching a stable saturated state (Fig. 6(f)).

Therefore, the larger the readout step size, the more severe the memory decay, and the fewer time steps can be accurately predicted during the prediction phase. For highly chaotic time series, frequent input during training helps maintain the system's activity, generating rich state variations that capture the complex dynamics of the sequence and enhance the system's prediction capacity. This allows the system to sustain longer periods of accurate prediction even in the absence of real data injection during the prediction phase.

3.2 Spoken Digit Recognition

In this subsection, we further evaluate the capability of the proposed PRC model for time-series classification through a spoken digit recognition task. Specifically, we use the free spoken digit dataset (FSDD) compiled by Zohar Jackson [44], which contains recordings of the digits 0–9 spoken by six speakers with different English accents. Each speaker recorded 50 utterances per digit, resulting in a total of 3,000 recordings sampled at 8 kHz. Then, we randomly partition the dataset into 2,400 samples for training and 600 for testing, corresponding to an 8:2 split. It is worth noting that due to significant variations in accent, speaking speed, and intonation among the speakers, distinguishing between classes is particularly challenging. Therefore, spoken digit recognition serves as an effective and widely used benchmark for evaluating the classification capability of the model.

The numerical validation framework for the spoken digit recognition task is illustrated in Fig. 7. The raw voice signals $u(t)$ contain the latent features of the digits but present several challenges, including high dimensionality, variable lengths, noise, and redundant information. Therefore, preprocessing is required. In this study, we adopt the Mel-frequency cepstral coefficients (MFCCs), which are widely used in speech recognition tasks. These coefficients are obtained by applying a discrete cosine inverse transform to the logarithm of the signal's energy spectrum [45, 46]. The preprocessing details are provided in Appendix C.

After this step, each sample is divided into 17 frames, with each frame comprising 13 MFCCs. For each frame of the Mel spectrogram, a pseudo-time axis is constructed along the direction of increasing Mel coefficient indices, as indicated by the red arrow in Fig. 7. The Mel-frequency cepstral coefficients of each frame are treated as the angles of the external magnetic field injected into the reservoir, thereby driving the system. The corresponding labels of the sample signals are transformed into vectors via one-hot encoding, enabling the model to output estimated class probabilities. Training is then performed following the time-series classification procedure described in Sec. 2, using 2,400 spoken signals in the training set.

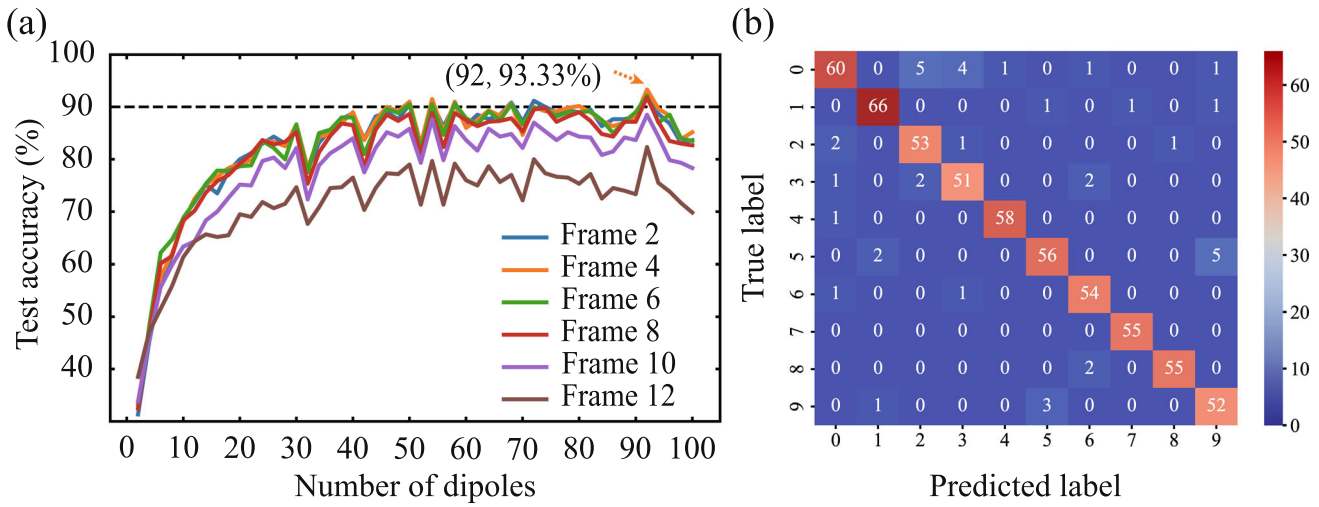

Fig. 8 Performance of the PRC model on the spoken digit recognition task. (a) The test accuracy as a function of the number of dipoles for different frames. The black dashed line serves as a reference for 90% test accuracy. (b) Confusion matrix corresponding to the reservoir configuration marked by the dotted arrow in (a)

After training, we evaluate the PRC model by using the 600 spoken signals in the test set. It can be seen from Fig. 8(a) that as the the number of dipoles increases, the test accuracy initially rises rapidly, reflecting that an increased reservoir size provides the system with more nonlinear couplings, thereby enhancing its prediction capacity. The rate of improvement then gradually levels off, indicating that the memory has approached saturation and cannot be further increased by enlarging the reservoir. If the reservoir size continues to grow, the overly complex nonlinear couplings may capture unnecessary details from the training set, leading to overfitting and a subsequent decrease in test accuracy. Similar to the time-series prediction task, the fluctuations in each curve arise from the randomness in the positions of the magnetic dipoles, but these do not affect the overall trend.

By comparing different frames, it can be observed that the distinctive speech features of the digits vary across frames, resulting in different predictive performance when the model is driven by different frames. Notably, the model achieves better performance when driven by frames with smaller indices, which may suggest that the initial segments of English pronunciations of Arabic numerals contain more distinctive features than the later segments. In Fig. 8(a), the dotted arrow indicates the optimal configuration corresponding to the 4th frame, where a reservoir composed of 92 magnetic dipoles achieved a test accuracy of 93.33%. The corresponding confusion matrix is shown in Fig. 8(b), where correctly classified samples are highly concentrated along the main diagonal. These results demonstrate that the proposed PRC model possesses sufficient nonlinear expressive capability to successfully perform time-series classification tasks. Please refer to Note S4 of Supplementary Information for benchmark comparison with other reservoir systems.

4 Conclusions

In conclusion, we proposed a PRC model consisting of a cluster of magnetic dipoles driven by a time-varying external magnetic field. Through mechanistic analysis, we demonstrated that the system can serve as an effective physical analogue of ESN for neuromorphic computing tasks. The model was validated on both regression and classification benchmarks. For regression, it achieved accurate autonomous prediction of Mackey–Glass time series across different chaotic regimes. For classification, it attained over 90% accuracy on the Free Spoken Digit Dataset, highlighting its capability for robust spoken digit recognition. These results collectively confirm that the proposed PRC framework provides a promising platform for PRC with broad applicability.

The results validate that the richness of the internal dynamics in the proposed model provides the essential memory and nonlinearity required for effective reservoir computing. Moreover, the proposed system features a noncontact, nonlocally coupled structure, which offers a scalable and hardware-friendly pathway for future physical implementations. This study not only introduces a new paradigm for constructing PRC systems but also opens promising avenues for leveraging magnetic dynamical systems as efficient reservoirs in neuromorphic computing.

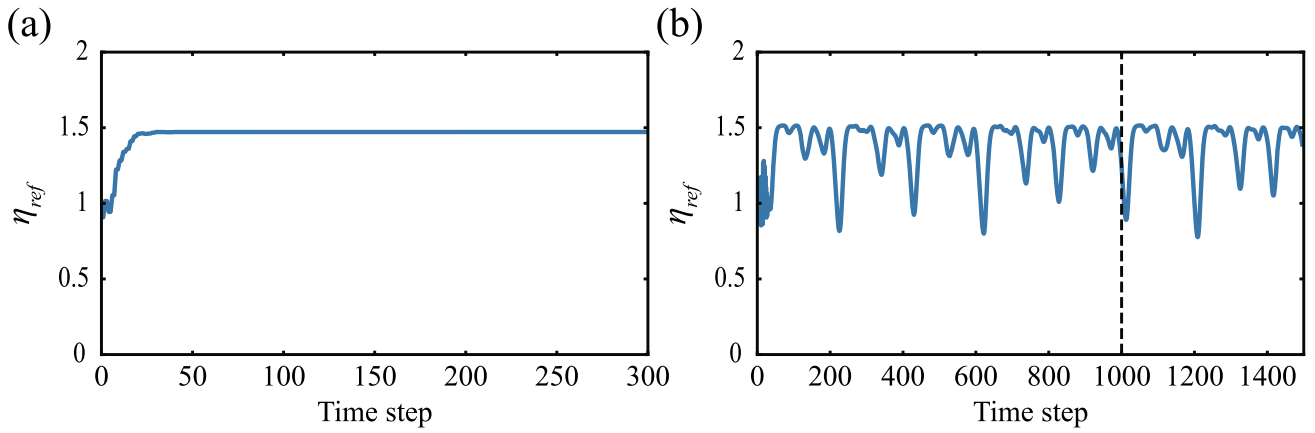

Fig. 9 Numerical stability test. (a) and (b) correspond to the systems with fixed angle injection and TS1 injection, respectively

Appendix A: Magnetic Hysteresis Model

The Jiles–Atherton model is one of the famous magnetic hysteresis models whose basic equation is as follows [40, 41]

$$\frac{dM}{dH} = \frac{M_{an} - M}{\delta k - \nu(M_{an} - M)}, \tag{21}$$

where M is the magnetization of material, H denotes the external magnetic field, k is a dissipation coefficient, ν is the coefficient that describe the magnetic domain coupling, δ takes value $+1$ and -1 respectively for $dH/dt > 0$ and $dH/dt < 0$, and M_{an} is the anhysteretic magnetization. M_{an} is originally defined by Langevin function as [40, 41]

$$M_{an}(H) = m_s \left[\coth\left(\frac{H + \nu M}{a}\right) - \frac{a}{H + \nu M} \right], \tag{22}$$

where a is Langevin parameter. The Eq. (22) is often approximated by the hyperbolic tangent function due to their similar shape and asymptotic behavior as

$$M_{an}(H) = m_s \tanh \frac{H}{B_0}, \tag{23}$$

where we have considered the weak magnetic coupling situation $\nu \approx 0$. Further, we seek to describe the hysteresis response in the form of a first-order differential equation. By using the Eq. (21) and chain rule we obtain

$$\frac{dM}{dt} = \frac{dM}{dH} \cdot \frac{dH}{dt} = \frac{1}{\tau_m}(-M + M_{an}). \tag{24}$$

We have take a single relaxation time constant $\tau_m \approx \delta k/\dot{H}$ to capture the average lag between the applied magnetic input and the magnetization response. Thus, by combining Eq. (23) and Eq. (24), we arrive

$$\tau_m \dot{M} = -M + m_s \tanh \frac{H}{B_0}. \tag{25}$$

In our model, the projection of the external magnetic field in the direction of the magnetic dipole should be considered, i.e. $H = B\cos(\theta - \theta_0(t))$. In the end, we get the simplified equation of magnetic hysteresis effect expressed in Eq. (3).

Appendix B: Numerical Stability Analysis of System Solutions

Numerical stability assessment is a crucial metric for evaluating the sensitivity of error propagation in an algorithm. It aims to determine whether the output remains reliable under small perturbations in the input data. Here, we employ the basic linearization method to evaluate the stability of the modified Runge–Kutta method in the main text, thereby illustrating the rationale for the chosen iteration step h. The well-known linearization method is to linearize the nonlinear system near the reference point and use the stability theory of linear system to analyze the stability of the original nonlinear system.

Based on the state vector relationship $\dot{r} = f(t, r)$, we expand it around the reference point as a Taylor series

$$f(r) = f(r_{ref}) + J(r_{ref})(r - r_{ref}) + o(r - r_{ref})^2, \tag{26}$$

where the Jacobian matrix is given by

$$J_{ij} = \frac{\partial f_i}{\partial r_j}. \tag{27}$$

By introducing a small perturbation using the difference scheme, we obtain

$$J_{:,k} = \frac{f(r + \epsilon e_k) - f(r - \epsilon e_k)}{2\epsilon}, \tag{28}$$

where $\epsilon = 1 \times 10^{-6}$ is a small quantity, $k = 1, ..., 3N$, e_k is unit vector defined as

$$(e_k)_i = \begin{cases} 1 & \text{if } i = k, \\ 0 & \text{if } i \neq k. \end{cases} \tag{29}$$

Thus, the eigenvalues λ_k of the Jacobian matrix can be determined using the following formula

$$|J(r) - \lambda_k I| = 0, \tag{30}$$

where λ_k is the identity matrix. Here, We employ the stability region solution of the Runge–Kutta method proposed by Ait–Haddou [47]. Specificlly, for an iterative test equation

$$r_{i+2} = P(\eta) r_i, \tag{31}$$

where $\eta = \lambda h$. The numerical stability region condition is

$$S = \{\eta \in \mathbb{C} || P(\eta) \leq 1|\}. \tag{32}$$

For Runge–Kutta method we have the stability polynomial $P(\eta)$ as

$$P(\eta) = 1 + \eta + \frac{\eta^2}{2!} + \frac{\eta^3}{3!} + \frac{\eta^4}{4!}. \tag{33}$$

Thus, by setting $P(\eta) = \pm 1$, we obtain the right and left boundaries of the stability region as $\eta = 0$ and $\eta = -2.785$. Therefore, numerical stability at the reference point can be determined by testing if the product of smallest negative eigenvalue and the iteration step belongs to [-2, 0], where we adopt a stricter bound of -2 in place of the actual value of -2.785. This condition can be further summarized as

$$\eta_{ref} = |\lambda^{-}_{min}| h \begin{cases} < 2 & \text{Stable} \\ \geq 2 & \text{Unstable.} \end{cases} \tag{34}$$

Figure 9 show the results of the stability test at each time step, where the iteration step $h = 0.5$. For the model with a fixed angle of external magnetic field as input, the numerical stability factor (η_{ref}) of the system evolves until the motion ceases. Once the system rest, the numerical stability of the solution naturally remains constant, as shown in Fig. 9(a). For the model with Mackey–Glass time series as input, the numerical stability factor remains in an oscillatory state throughout both the training and prediction phases. Across both tests, η_{ref} is consistently bellow 2, suggesting that the numerical solution remains in the stable region throughout. This confirms the reliability and effectiveness of the system solution.

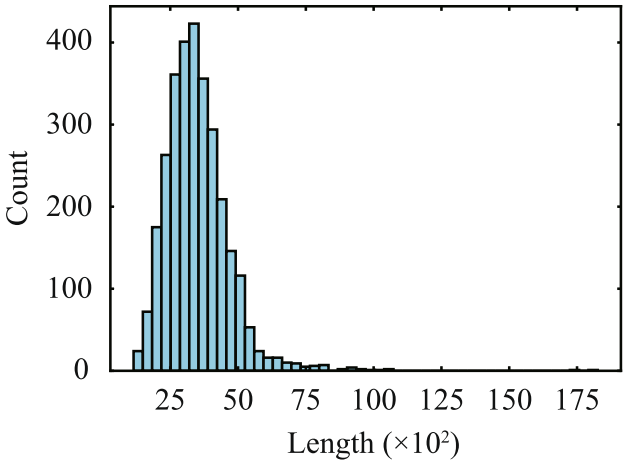

Fig. 10 Histogram of sample signal lengths in the spoken digit dataset

Appendix C: Mel-frequency Cepstral Coefficients

Here, we describe the preprocessing procedure employed for spoken digit recognition, which converts the raw audio signals into MFCCs. As shown in Fig. 10, the histogram of the FSDD dataset highlights substantial variability in the lengths of the recorded voice signals, and most of the samples have lengths below 10000 data points, while a few samples are excessively long and potentially redundant.

To address this problem, all samples were firstly standardized to a length of 8192 data points by truncation and zero-padding, thereby removing excessively long segments while preserving information in shorter samples through padding. Next, the number of data points used for each frame was set to $N = 8192$, with a frame shift of $R = 512$ between consecutive frames, resulting in 17 frames per signal sample. For each frame, 4096 data points on either side of the Fourier transform center were considered, with zero-padding applied to any points that exceeded the signal boundaries. For a given frame $u(n)$, the discrete Fourier transform is computed as follows

$$U(k) = \sum_{n=0}^{N-1} u(n)\omega(n)e^{-j2\pi kn/N}, \tag{35}$$

where $\omega(n) = 0.5[1+\cos[2\pi n/(N-1)]$ is Hanning window. Further, we get the unit power spectrum

$$P(k) = \frac{|U(k)|^2}{N}. \tag{36}$$

The Mel frequency is a nonlinear frequency scale designed to match human auditory perception. Its relationship to the

linear frequency is given by

$$f_{mel} = 2595 \log_{10}(1 + f/700). \tag{37}$$

Here, we divide the Mel frequency axis into $M = 128$ equally spaced segments. These points are then mapped back to the linear scale to obtain the corresponding frequency values, and the following piecewise function is used to construct 127 triangular Mel filters

$$H_m(k) = \begin{cases} 0, & k < f_{m-1} \\ \frac{k - f_{m-1}}{f_m - f_{m-1}}, & f_{m-1} \le k < f_m \\ \frac{f_{m+1} - k}{f_{m+1} - f_m}, & f_m \le k < f_{m+1} \\ 0, & k \ge f_{m+1} \end{cases} \tag{38}$$

Applying the filters to the power spectrum, we obtain the output energy of the m-th filter

$$E_m = \sum_{k=f_{m-1}}^{f_{m+1}} P(k) H_m(k) \tag{39}$$

Take the logarithm of the output energy, and apply the discrete cosine inverse transform to obtain the Mel-frequency cepstral coefficients as follows

$$C_i = \sum_{m=1}^{M-1} \log E_m \cos[\pi i (m - 0.5)/(M - 1)], \quad i = 0, 1, ..., 12. \tag{40}$$

Thus, after preprocessing, each frame of the original signal is ultimately transformed into 13 Mel-frequency cepstral coefficients that encapsulate the internal voice features, as illustrated by the spectrogram in Fig. 7. The coefficients are normalized to the range $[-1, 1]$.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11071-026-12291-4.

Acknowledgements This work is supported by the National Natural Science Foundation of China (12272267, 52278411), the Shanghai Science and Technology Committee (Grant No.22JC1404100), the Fundamental Research Funds for the Central Universities and the Shanghai Oriental Talents Top- Tier Program. This publication is part of the projects PID2024-158832NB-C22 and PID2024-158832NB-C21, and the European project 101150911. We thank Project No. CNS2023-145510 funded by MCIN/AEI/10.13039/501100011033, "European Union NextGenerationEU/PRTR". This work was supported by DYNAMO project (101046489), funded by the European Union. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or European Innovation Council. Neither the European Union nor the granting authority can be held responsible for them.

Author Contributions L.H., D.T., and Y.J. conceived the physical mechanism. L.H., S.E., X.Z., T.B., and P.D.G. developed the computational framework. L.H. and D.T. wrote the main manuscript. All authors reviewed the manuscript. D.T. and Y.J. supervised the work.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Data Availability The data that support the findings of this study are available from the corresponding authors upon reasonable request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Jordan, M.I., Mitchell, T.M.: Machine learning: Trends, perspectives, and prospects. Science **349**(6245), 255–260 (2015)
2. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
3. Wei, X.S., Song, Y.Z., Mac Aodha, O., Wu, J., Peng, Y., Tang, J., Yang, J., Belongie, S.: Fine-grained image analysis with deep learning: A survey. IEEE Trans. Pattern Anal. Mach. Intell. **44**(12), 8927–8948 (2021)
4. Nassif, A.B., Shahin, I., Attili, I., Azzeh, M., Shaalan, K.: Speech recognition using deep neural networks: A systematic review. IEEE access **7**, 19143–19165 (2019)
5. Kim, Y., Kim, Y., Yang, C., Park, K., Gu, G.X., Ryu, S.: Deep learning framework for material design space exploration using active transfer learning and data augmentation. npj Comput. Mater. **7**(1), 140 (2021)
6. Kiarashinejad, Y., Abdollahramezani, S., Adibi, A.: Deep learning approach based on dimensionality reduction for designing electromagnetic nanostructures. npj Comput. Mater. **6**(1), 12 (2020)
7. Ma, W., Liu, Z., Kudyshev, Z.A., Boltasseva, A., Cai, W., Liu, Y.: Deep learning for the design of photonic structures. Nat. Photonics **15**(2), 77–90 (2021)
8. Jin, Y., He, L., Wen, Z., Mortazavi, B., Guo, H., Torrent, D., Djafari-Rouhani, B., Rabczuk, T., Zhuang, X., Li, Y.: Intelligent on-demand design of phononic metamaterials. Nanophotonics **11**(3), 439–460(2022)
9. Muhammad, Kennedy, J., Lim, C.: Machine learning and deep learning in phononic crystals and metamaterials–A review. Mater. Today Commun. **33**, 104606 (2022)
10. Momeni, A., Rahmani, B., Scellier, B., Wright, L.G., McMahon, P.L., Wanjura, C.C., Li, Y., Skalli, A., Berloff, N.G., Onodera, T.,

et al.: Training of physical neural networks. Nature **645**(8079), 53–61(2025)
11. Sun, Z., Pedretti, G., Ambrosi, E., Bricalli, A., Wang, W., Ielmini, D.: Solving matrix equations in one step with cross-point resistive arrays. Proc. Natl. Acad. Sci. **116**(10), 4123–4128 (2019)
12. Sun, Z., Pedretti, G., Bricalli, A., Ielmini, D.: One-step regression and classification with cross-point resistive memory arrays. Sci. Adv. **6**(5), eaay2378 (2020)
13. Yao, P., Wu, H., Gao, B., Tang, J., Zhang, Q., Zhang, W., Yang, J.J., Qian, H.: Fully hardware-implemented memristor convolutional neural network. Nature **577**(7792), 641–646 (2020)
14. Torrejon, J., Riou, M., Araujo, F.A., Tsunegi, S., Khalsa, G., Querlioz, D., Bortolotti, P., Cros, V., Yakushiji, K., Fukushima, A., et al.: Neuromorphic computing with nanoscale spintronic oscillators. Nature **547**(7664), 428–431 (2017)
15. Prychynenko, D., Sitte, M., Litzius, K., Krüger, B., Bourianoff, G., Kläui, M., Sinova, J., Everschor-Sitte, K.: Magnetic skyrmion as a nonlinear resistive element: a potential building block for reservoir computing. Phys. Rev. Appl. **9**(1), 014034 (2018)
16. Raab, K., Brems, M.A., Beneke, G., Dohi, T., Rothörl, J., Kammerbauer, F., Mentink, J.H., Kläui, M.: Brownian reservoir computing realized using geometrically confined skyrmion dynamics. Nat. Commun. **13**(1), 6982 (2022)
17. Tuma, T., Pantazi, A., Le Gallo, M., Sebastian, A., Eleftheriou, E.: Stochastic phase-change neurons. Nat. Nanotechnol. **11**(8), 693–699 (2016)
18. Boybat, I., Le Gallo, M., Nandakumar, S., Moraitis, T., Parnell, T., Tuma, T., Rajendran, B., Leblebici, Y., Sebastian, A., Eleftheriou, E.: Neuromorphic computing with multi-memristive synapses. Nat. Commun. **9**(1), 2514 (2018)
19. Zangeneh-Nejad, F., Sounas, D.L., Alù, A., Fleury, R.: Analogue computing with metamaterials. Nat. Rev. Mater. **6**(3), 207–225 (2021)
20. Lin, X., Rivenson, Y., Yardimci, N.T., Veli, M., Luo, Y., Jarrahi, M., Ozcan, A.: All-optical machine learning using diffractive deep neural networks. Science **361**(6406), 1004–1008 (2018)
21. Zuo, Y., Li, B., Zhao, Y., Jiang, Y., Chen, Y.C., Chen, P., Jo, G.B., Liu, J., Du, S.: All-optical neural network with nonlinear activation functions. Optica **6**(9), 1132 (2019)
22. Li, J., Mengu, D., Yardimci, N.T., Luo, Y., Li, X., Veli, M., Rivenson, Y., Jarrahi, M., Ozcan, A.: Spectrally encoded single-pixel machine vision using diffractive networks. Sci. Adv. **7**(13), eabd7690 (2021)
23. Weng, J., Ding, Y., Hu, C., Zhu, X.F., Liang, B., Yang, J., Cheng, J.: Meta-neural-network for real-time and passive deep-learning-based object recognition. Nat. Commun. **11**(1), 6309 (2020)
24. Hughes, T.W., Williamson, I.A., Minkov, M., Fan, S.: Wave physics as an analog recurrent neural network. Sci. Adv. **5**(12), eaay6946 (2019)
25. Qu, Y., Zhou, M., Khoram, E., Yu, N., Yu, Z.: Resonance for analog recurrent neural network. ACS Photonics **9**(5), 1647–1654 (2022)
26. Jiang, T., Li, T., Huang, H., Peng, Z.K., He, Q.: Metamaterial-based analog recurrent neural network toward machine intelligence. Phys. Rev. Appl. **19**(6), 064065 (2023)
27. Cucchi, M., Abreu, S., Ciccone, G., Brunner, D., Kleemann, H.: Hands-on reservoir computing: a tutorial for practical implementation. Neuromorph. Comput. Eng. **2**(3), 032002 (2022)
28. Yan, M., Huang, C., Bienstman, P., Tino, P., Lin, W., Sun, J.: Emerging opportunities and challenges for the future of reservoir computing. Nat. Commun. **15**(1), 2056 (2024)
29. Appeltant, L., Soriano, M.C., Van der Sande, G., Danckaert, J., Massar, S., Dambre, J., Schrauwen, B., Mirasso, C.R., Fischer, I.: Information processing using a single dynamical node as complex system. Nat. Commun. **2**(1), 468 (2011)
30. Liang, X., Zhong, Y., Tang, J., Liu, Z., Yao, P., Sun, K., Zhang, Q., Gao, B., Heidari, H., Qian, H., et al.: Rotating neurons for all-analog implementation of cyclic reservoir computing. Nat. Commun. **13**(1), 1549 (2022)
31. Hauser, H., Ijspeert, A.J., Füchslin, R.M., Pfeifer, R., Maass, W.: Towards a theoretical foundation for morphological computation with compliant bodies. Biol. Cybern. **105**(5), 355–370 (2011)
32. Momeni, A., Fleury, R.: Electromagnetic wave-based extreme deep learning with nonlinear time-Floquet entanglement. Nat. Commun. **13**(1), 2651 (2022)
33. Du, C., Cai, F., Zidan, M.A., Ma, W., Lee, S.H., Lu, W.D.: Reservoir computing using dynamic memristors for temporal information processing. Nat. Commun. **8**(1), 2204 (2017)
34. Moon, J., Ma, W., Shin, J.H., Cai, F., Du, C., Lee, S.H., Lu, W.D.: Temporal data classification and forecasting using a memristor-based reservoir computing system. Nat. Electron. **2**(10), 480–487 (2019)
35. Zhong, Y., Tang, J., Li, X., Gao, B., Qian, H., Wu, H.: Dynamic memristor-based reservoir computing for high-efficiency temporal signal processing. Nat. Commun. **12**(1), 408 (2021)
36. Larger, L., Baylón-Fuentes, A., Martinenghi, R., Udaltsov, V.S., Chembo, Y.K., Jacquot, M.: High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification. Phys. Rev. X **7**(1), 011015 (2017)
37. Antonik, P., Marsal, N., Brunner, D., Rontani, D.: Human action recognition with a large-scale brain-inspired photonic computer. Nat. Mach. Intell. **1**(11), 530–537 (2019)
38. Tanaka, G., Yamane, T., Héroux, J.B., Nakane, R., Kanazawa, N., Takeda, S., Numata, H., Nakano, D., Hirose, A.: Recent advances in physical reservoir computing: A review. Neural Netw. **115**, 100–123 (2019)
39. Burgess, A., Waller, M.C., Gauger, E.M., Bennett, R.: Engineering dipole-dipole couplings for enhanced cooperative light-matter interactions. Phys. Rev. Lett. **134**(11), 113602 (2025)
40. Jiles, D.C., Atherton, D.L.: Theory of ferromagnetic hysteresis. J. Magn. Magn. Mater. **61**(1–2), 48–60 (1986)
41. Jiles, D.C., Thoelke, J., Devine, M.: Numerical determination of hysteresis parameters for the modeling of magnetic properties using the theory of ferromagnetic hysteresis. IEEE Trans. Magn. **28**(1), 27–35 (2002)
42. Lin, W.Y., Huang, C.W.: Absolute rotary encoder system based on optical sensor for angular measurement. J. Supercomput. **77**(8), 8355–8373 (2021)
43. Shaw, G., Kramer, R., Dempsey, N., Hasselbach, K.: A scanning Hall probe microscope for high resolution, large area, variable height magnetic field imaging. Rev. Sci. Instrum. **87**(11), 113702 (2016)
44. Jackson, Z.:Free spoken digit dataset. https://github.com/Jakobovski/free-spoken-digit-dataset (2020). Accessed: 2025-06-10
45. Davis, S., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. Speech Signal Process. **28**(4), 357–366 (1980)
46. Sahidullah, M., Saha, G.: Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition. Speech Commun. **54**(4), 543–565 (2012)
47. Ait-Haddou, R.: New stability results for explicit Runge–Kutta methods. BIT Numer. Math. **59**(3), 585–612 (2019)