Frontiers of Physics
https://doi.org/10.1007/s11467-021-1111-8
Front. Phys.
17(2), 22504 (2022)

RESEARCH ARTICLE

Revisiting the dynamics of Bose–Einstein condensates in a double well by deep learning with a hybrid network

Shurui Li[1], **Jianqin Xu**[1], **Jing Qian**[1,†], **Weiping Zhang**[2,3,4]

[1] *Department of Physics, School of Physics and Electronic Science, East China Normal University, Shanghai 200062, China*
[2] *School of Physics and Astronomy, and Tsung-Dao Lee Institute, Shanghai Jiao Tong University, Shanghai 200240, China*
[3] *Shanghai Research Center for Quantum Sciences, Shanghai 201315, China*
[4] *Collaborative Innovation Center of Extreme Optics, Shanxi University, Taiyuan 030006, China*
Corresponding author. E-mail: †*jqian1982@gmail.com*
Received May 13, 2021; accepted August 12, 2021

Deep learning, accounting for the use of an elaborate neural network, has recently been developed as an efficient and powerful tool to solve diverse problems in physics and other sciences. In the present work, we propose a novel learning method based on a hybrid network integrating two different kinds of neural networks: Long Short-Term Memory (LSTM) and Deep Residual Network (ResNet), in order to overcome the difficulty met in numerically simulating strongly-oscillating dynamical evolutions of physical systems. By taking the dynamics of Bose–Einstein condensates in a double-well potential as an example, we show that our new method makes a highly efficient pre-learning and a high-fidelity prediction about the whole dynamics. This benefits from the advantage of the combination of the LSTM and the ResNet and is impossibly achieved with a single network in the case of direct learning. Our method can be applied for simulating complex cooperative dynamics in a system with fast multiple-frequency oscillations with the aid of auxiliary spectrum analysis.

Keywords double-well, deep learning, hybrid neural network

1 Introduction

Deep learning (DL) has been devoted to solving physical problems in recent years. With the help of its extraordinary abilities to make predictions about data and to extract meaningful features, exciting progresses have been made in the research fields of quantum dynamics [1, 2], many-body physics [3], precision measurement [4], condensed-matter physics [5–9] and quantum machine learning [10–12]. When applying DL the primary problem faced by us is to establish a suitable network for the training data set. So far several neural networks (NNs) are utilized. For example, the earlier artificial neural network can approximate any distributions with only two neural layers [13]; but it becomes inaccurate when the layer number increases [14]. The restricted Boltzmann machine is an unsupervised method, which can uncover the information underlying the data by learning the joint distribution with Bayesian theory [15]. It is difficult to train the system without a fine partition-function solution [16]. Other NNs such as the convolutional NN [17] and the recurrent NN [18] are more universal in extracting local features of data and learning the relationships among elements with a high accuracy [19, 20], despite that they may face the problem of gradient exploding or gradient vanishing due to the complexity of dynamics. Therefore a single type of NN becomes incapable of solving complex population dynamics.

The quality of simulating a fast-oscillating dynamics usually depends on the precision of sampling [21]. When the number of sampling points is insufficient, exact numerical ways may arise poor outcomes that are largely deviated from its real values [22]. However a high-precision sampling will add to the costs of experiments; at the same time it also leads to a long-time calculation limited by the capacity of computers. Therefore it is quite difficult to accurately obtain the data of fast-oscillating dynamics both in theory and in experiment. In the present work we propose a hybrid NN involving two different networks which are so-called Long short-term memory (LSTM) [23] and Residual network (ResNet) [24, 25], in order to study the fast-oscillating dynamical evolution of systems. Taking a generalized double-well dynamics of Bose–Einstein condensates (BECs) as an example, we show that our modified DL method can well solve the time evolution of population difference between two wells with high fidelity of the phase and the frequency of oscillations. This mod-

* arXiv: 2104.14657. This article can also be found at http://journal.hep.com.cn/fop/EN/10.1007/s11467-021-1111-8.

© Higher Education Press 2022

ified DL refers to one kind of series connection between LSTM and ResNet with its performance enhanced by a pre-learning of periodicity in dynamics. ResNet will use the learning outcomes from LSTM to work better. In addition, an improved integrated DL based on modified DL is also proposed to facilitate the training time, which is enabled by changing the loss function for feedback. With this feedback mechanism, LSTM can benefit from ResNet, resulting in the reduction of data amount needed for training.

For atomic Bose–Einstein condensates (BECs) trapped in a double-well potential, a large number of achievements were carried out arising a well-solved theoretical model [26–30]. This leads to an easier generation of enough training data via the Runge-Kutta method beforehand. Our hybrid network enables a highly-efficient learning of frequency based on the periodicity verification of strongly-oscillating population dynamics. Moreover, even if the original data is not enough the LSTM network can also perform a reliable prediction on the higher frequency values of the population oscillating in the macroscopic quantum self-trapping (MQST) regime, which are far beyond the trained parameters. However, the numerical method fails to work in this case because of the inadequate sampling. Compared to the direct DL that only contains single kind of ResNet, integrating LSTM and ResNet can deeply increase the learning efficiency despite expending a slightly longer training time, since more layers are required for constructing this integrated network. The final infidelity obtained by estimating the deviation of fractional population difference can be much lower than 0.01 with the help of a hybrid network learning.

2 Residual network

Structure of our residual network is shown by a series of blocks in Fig. 1. The filter kernel serving as a central element in the learning process, should be carefully chosen. We use filter kernels to scan matrice which enables the feature extraction from objects. By element-wisely multiplying matrice and filter kernels, machine can well learn the features of input data. However, with deeper networks, a traditional convolutional NN becomes more difficult to train, since we may face the problem of gradient vanishing or gradient exploding, which leads to a poor performance in DL. Hence, we choose a deep residual network (or so-called *ResNet*) to solve this problem. An element-wise sum in the ResNet labeled by arrows in Fig. 1, will prevent the loss of information from passing through deep networks, and also avoid the vanishing of gradient by adding the former input with present hidden layer's output. This simple process preserves the former information which means that it increases the volume of data handled by the next layer. The detail of this technology can be seen in Ref. [24]. To be more specific, arrows linking the batch normalization layer and the element-wise sum layer in Fig. 1, point to the process of residual learning.

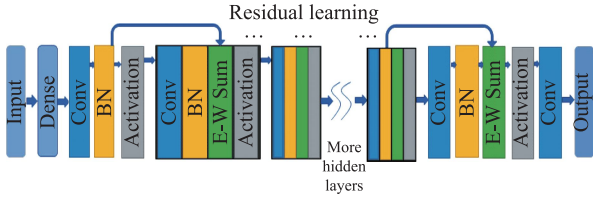

Fig. 1 Schematic diagram for ResNet involving a series of identically-distributed one-dimension convolutional (Conv) layers and batch normalization (BN) layers. At each convolutional layer it has a dilation rate except the first and the last layers. Different from the original convolutional NN scheme, arrows acting on Element-wise sum layers (E-W sum) can preserve the former hidden layer's information which facilitate the performance of DL.

From physical review, we are more interested in training or generating useful information within a "black box" [the red frame in Fig. 5(b)] if the input is determined. Although training a high-performance network needs a lot of expense yet it may enable a robust prediction about the unknown results in a short-time period, especially while dealing with a complex system in the absence of analytical solutions. Our motivation is building an efficient learning protocol which can bridge the results from original numerical simulation and unknown prospective result based on the given input parameters. That may facilitate an easy and robust simulation for solving complex physical problems.

3 Double-well dynamics with two BECs

A double-well potential serving as a basic platform in the study of microscopic mediums, especially for atomic BECs [31–35], has been focused by researchers for decades. A typical double-well potential can be easily created by the combination of a periodic optical potential with strong harmonic confinement, played by a periodic one-dimensional light shift [36]. If atomic BECs are confined in such a double-well optical trap, the effects of boson Josephson junction (BJJ) or MQST can be observed.

Our study starts from a brief introduction to the dynamical evolution of BECs in a double-well potential. The wave function $\psi(r,t)$ for two weakly-coupled atomic condensates is expanded as

$$\psi(r,t) = a_1(t)\phi_1(r) + a_2(t)\phi_2(r) \tag{1}$$

under the mean-field approximation. To be universal, two wells are asymmetric and has an energy difference γ between them. A standard treatment for solving the population dynamics between two wells, will adopt the nonlinear

two-mode dynamical equations (here $\hbar = 1$)

$$\mathrm{i}\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \hat{H}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \tag{2}$$

with the Hamiltonian $\hat{H}$ given by

$$\hat{H} = \begin{pmatrix} \gamma + U_1 N_1 & -K \\ -K & U_2 N_2 \end{pmatrix} \tag{3}$$

and the amplitudes $a_j = \sqrt{N_j}\mathrm{e}^{\mathrm{i}\theta_j}$ $(j \in (1,2))$ with N_j, θ_j the number and the phase of particles in the jth trap. Here $N_T = N_1 + N_2$ is a constant and U_j is the trapped atomic interaction which is related to the atomic scattering length. K is the inter-trap coupling coefficient [26]. Upon defining the phase difference $\phi = \theta_2 - \theta_1$ and the fractional population difference $z = \frac{N_1 - N_2}{N_T}$, Eq. (2) can be re-organized as

$$\begin{aligned} \dot{z}(t) &= -\sqrt{1 - z^2(t)}\sin[\phi(t)], \\ \dot{\phi}(t) &= \Delta E + \Lambda z(t) + \frac{z(t)}{\sqrt{1 - z^2(t)}}\cos[\phi(t)], \end{aligned} \tag{4}$$

in which we rescale the time by $2Kt \to t$ ($2K$ is the frequency unit). Other dimensionless parameters are

$$\begin{aligned} \Delta E &= \gamma/(2K) + (U_1 - U_2)N_T/(4K), \\ \Lambda &= (U_1 + U_2)N_T/(4K). \end{aligned} \tag{5}$$

accordingly, which indicate the imbalanced energy of two wells as well as the atomic tunneling rate respectively. Solving $z(t)$ and $\phi(t)$ essentially gives rise to the whole population dynamics of atomic BECs in a double-well system.

It is clearly shown that the pair of variables $z(t)$ and $\phi(t)$ are canonically conjugate satisfying $\dot{z} = -\frac{\partial H}{\partial \phi}$, $\dot{\phi} = \frac{\partial H}{\partial z}$. Based on that, the Hamiltonian $\hat{H}$ can be rewritten as

$$\hat{H} = \frac{\Lambda}{2}z^2 + \Delta E z - \sqrt{1 - z^2}\cos\phi, \tag{6}$$

By simply assuming $\gamma = 0$ (symmetric) and ensuring $U_1 = U_2 = U$, it leads to $\Delta E = 0$, $\Lambda = UN_T/(2K)$. The initial energy starts from

$$H_0 = \frac{\Lambda}{2}z(0)^2 - \sqrt{1 - z(0)^2}\cos\phi(0), \tag{7}$$

which is conserved during the dynamical evolution. According to the initial phase difference $\phi(0) = 0$ or π we can obtain the zero-mode or π-phase mode individually. Here we focus on the former case. $z(0) \in [-1, 1]$ stands for the initial population imbalance and is tunable. In experiment via the adjustment of $\Lambda \sim U/K$ that characterizes the relative strength between interaction and tunneling rate, we can observe two different regimes: BJJ and MQST. The former meets $\langle z(t)\rangle = 0$ and $\langle \phi(t)\rangle = 0$ around $z = 0$ when Λ is small. As increasing Λ that exceeds a critical value Λ_c, the atomic population oscillation becomes self-trapped giving to $\langle z(t)\rangle \neq 0$ on average. This behavior is also accompanied by an unbound and increasing phase that is winding with time.

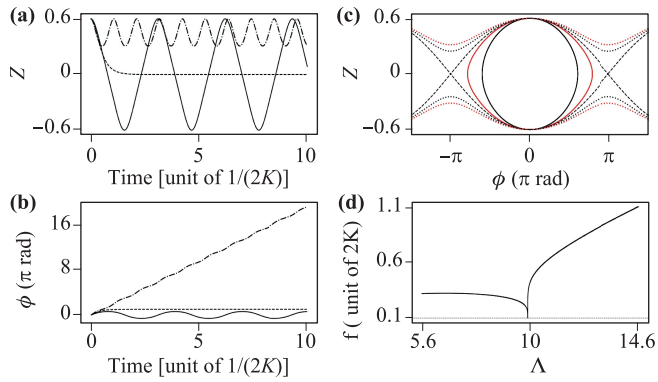

Fig. 2 **(a, b)** Numerical results $z(t)$ and $\phi(t)$ for $\Lambda = 6.2$ (solid), 10 (dashed, critical value) and 13.4 (dash-dotted). **(c)** The z–ϕ diagram under Λ= 6.2 (black solid), 8.8 (red solid), 10 (black-dashed), 11.8 (black-dotted), 13.4 (red-dotted). **(d)** Single oscillating frequency f of the population difference $z(t)$ for $\Lambda \in [5.6, 14.6]$. At the critical point $\Lambda_c = 10$, we set $f = 0.1$.

The critical value between Josephson oscillation and self-trapping effects has been analytically solved at $H_0 = 1$, arising [37]

$$\Lambda_c = \frac{2 + 2\sqrt{1 - z(0)^2}\cos\phi(0)}{z(0)^2}, \tag{8}$$

which means if the initial energy increases exceeding $H_0 = 1$ the fractional population difference will reveal a significant change, corresponding to the critical point between the two regimes.

By using a standard fourth-order Runge–Kutta method, our numerical simulations for $z(t)$ and $\phi(t)$ are explicitly presented in Figs. 2(a) and (b). As expected, both the population dynamics and the phase variation manifest the existence of a significant transition across the critical value Λ_c. If $\Lambda < \Lambda_c$ a BJJ appears between two wells accompanied by a small phase fluctuation in time, ensuring the average values $\langle z(t)\rangle = \langle \phi(t)\rangle = 0$. However when Λ becomes larger, atom tends to be self-trapped in single-well with a time-dependent and increasing phase. Figures 2(c) and (d) show a global representation of the phase diagram in the (z, ϕ) space as well as the frequency f of population dynamics versus Λ. The critical transition at $\Lambda_c = 10$ is clearly seen, agreeing with the discussions in (a, b). These numerical results allow us to establish a sufficient and accurate data set to train the neural network. In addition the DL method enables a robust prediction about unknown results which are far beyond the test parameters in data set.

4 Direct learning

To show the robustness of DL which can be applied for diverse physical problems, we first take this double-well model as an example, since both the fractional population difference $z(t)$ and the phase dynamics $\phi(t)$ of two BECs can be analytically solved here. Nowadays the DL method is regarded as a shortcut for this target. Because once a data set has been formed by using part of the original data obtained from numerical simulations, it reveals a strong ability of generating similar data $\phi(t)$ and $z(t)$ with the training network. Most interestingly, benefiting from the ability of strong prediction it can even give other unknown results which are beyond the original data set.

Our work starts from equation (4) by setting e.g. $z(0) = 0.6$, $\phi(0) = 0$, arising $\Lambda_c = 10$. The original parameter range is fixed to $\Lambda \in [5.6, 14.6]$ covering two regimes: MQST and BJJ. The data set is formed by uniformly separating 90001 Λ values within this range and solving $\phi(t)$ and $z(t)$ for each Λ via the method of fourth-order Runge-Kutta. The time interval is $\Delta t = 0.02$ with a unit of $1/2K$. As a result, we can finally obtain two two-dimension data sets for z and ϕ in the space of (Λ, t). Each contains 90001×501 data. While training the network with data sets we randomly choose 9000 samples as a validation set, leaving the remaining 81001 samples for training. All computing results are based on Intel Core i7-7900X @3.30GHz and GeForce RTX 2060 with 1024 batch size and 32 filter size on Python 3.8.

An efficient learning depends on the design of network. To search for a suitable network, in Table 1 we select different hidden-layers and calculate the losses after 400 epochs by using different activation functions. Here the "layers" means the number of hidden convolutional layers. In general as the increase of layer numbers the Mean-squared-error (MSE) becomes lower, however at the expense of a longer training time. To propose a low-expense method with high precision, we use *efficiency* (the last column in Table 1) as an indication for one network's compromised performance, which is defined by the training speed divided by the Tanh-type loss. It is clearly shown that a seven-layer model is the best one because of the maximal efficiency value ~ 9.92 (bold). Also, we compare the results based on two different activation functions: Tanh and Sine. The Tanh-type function can bring a lower learning loss than the Sine function, which will be considered in the calculation. In a practical training, we choose optimizer Adam for implementing the optimization [38]. In order to prevent over-fitting, the decay factor is 0.8, the minimum learning rate is 5×10^{-6} and the patience is 2.0 in the stage of Early-stopping and adaptive learning. The MSE loss function $\mathcal{L}$ is defined by

$$\mathcal{L} = \frac{1}{N_\Lambda}\sum_{\Lambda}^{N_\Lambda}\sum_{i}^{N}\frac{[z(\Lambda,t_i)-g(\Lambda,t_i)]^2}{N}, \tag{9}$$

where $z(\Lambda, t_i)$ is the learning outcome, i.e., the fractional population imbalance, at $t_i = \Delta t \times i$ ($i \in [1, N]$) for each Λ. $g(\Lambda, t_i)$ is the ground-truth from numerical solutions for each Λ. N is the length of arrays and N_Λ is the number of Λ learned in ResNet. Here $N = 501$, $\Delta t = 0.02$, $N_\Lambda = 81001$.

Figure 3 globally displays the population difference $z(t)$ and the phase variation $\phi(t)$ in two regimes, extracted from direct DL and numerical ways respectively. As for the learning of phase variation our results well fit with the numerical predictions see (c, d). Note that when plotting (d) we do not constrain the range of ϕ. So due to the linear increasing of ϕ accompanied by a slight oscillation in the long-time regime, the number of layers(here we choose seven hidden layers) becomes insufficient. As shown in the inset of (d), with the increase of layer numbers: e.g., seven layers (red-solid), nine layers (blue-solid), a nine-layer network in direct DL provides a better agreement with the numerical result (black-dashed). However, the direct DL with single kind of network fails to work well in the case of $z's$ learning, especially in the MQST regime.

Table 1 Comparison for the training efficiencies under different number of hidden layers, activation functions and the average spending time for one epoch. Efficiency is given by the speed (=1/Time) divided by the Tanh's loss.

Layers	Loss for Tanh	Loss for Sine	Time (s)	Efficiency
5	0.0173	0.0192	7	8.26
7	**0.0112**	**0.0131**	**9**	**9.92**
9	0.0097	0.0102	11	9.37
11	0.0088	0.0092	15	7.58
13	0.0086	0.0089	20	5.81

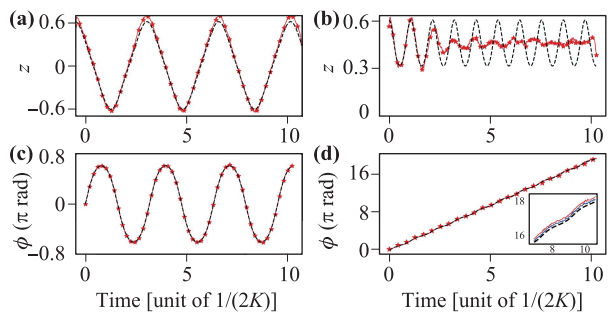

Fig. 3 Direct DL of $z(t)$ **(a, b)** and $\phi(t)$ **(c, d)** marked by red stars. Numerical results are contrastively given by the black-dashed curves, as similar as the results shown in Figs. 2(a) and (b). For (a, c) $\Lambda = 6.2$ (BJJ regime), $z(t)$ and $\phi(t)$ solved from the two methods have a perfect agreement. As turning to the regime of MQST, i.e., (b, d) $\Lambda = 13.4$, only $\phi(t)$ well agrees with the numerical prediction and $z(t)$ exhibits a larger mismatch as time increases. Inset of (d) shows the behavior of $\phi(t)$ as a function of time under different layer numbers.

Because in this regime two atomic BECs trapped in one well, will show a fast-oscillating behavior, leading to the breakdown of our thin single network (ResNet). We find the learning outcomes can not catch up with the change of $z(t)$, see (b). In the core of direct DL, such a rapid oscillation with small peak-peak amplitudes can not be easily trained by a limited database. Because the small loss in the whole process would cause gradient vanishing, making the whole learning process inefficient.

To improve the learning quality, especially in the case of MQST where the dynamics of $z(t)$ behaves as rapid oscillations with $\langle z(t)\rangle \neq 0$, we have to search for a new learning method. Since the periodicity of population dynamics is apparently seen, it is easy to decompose the function into two components: frequency and amplitude within one period. So we can learn them separately based on the dynamical character in one period, instead of implementing a global optimization for all hyper-parameters. This new learning method will be beneficial owing to the use of a low-expense network with fewer number of hidden layers as compared to the direct DL, if same accuracy is attainable. The modified network only needs a small receptive field size which indeed reduces the number of kernel layers and its size. Before carrying out our new method we have to verify the periodic feature of $z(t)$.

5 Modified learning

5.1 Periodicity verification

Periodicity verification can facilitate the design of network in modified DL. Compared to the traditional ways for that purpose such as Fourier Transform (FT) [39] and Auto-correlation Function (ACF) [39], here we choose the method originating from the chaotic physics to determine $z(t)$'s periodicity since it is more visible. This is called *Poincaré section* [40, 41]. For an orbit in a two-dimensional phase space, one could always select an appropriate hyperplane which should not contain or be tangent with this orbit. In general, if there exists a fixed-point or fixed number of discrete points, the orbit must have a determined period [42]. Based on the nonlinearity of Eq. (4) we use this method to determine the periodicity of dynamics.

In this double-well system, we choose ϕ=0 as a hyperplane in the BJJ regime and ϕ=π in the MQST regime. We draw the points where the trajectory of $(z-\dot{z})$ crosses this hyperplane. Specific examples are displayed in Fig. S1 of Appendix A, in which we compare the relations of $(z-\phi)$, $(z-\dot{z})$ and the Poincaré maps under different Λ values. When plotting the Poincaré maps the periodic character of $z(t)$ can be clearly confirmed by the presence of some discrete points.

5.2 Frequency learning

A direct simulation for the frequency f as a function of Λ has been solved by adopting numerical ways in Fig. 2(d), where the critical frequency at $\Lambda = \Lambda_c$ is fixed to be 0.1. When turning to the problem of solving an one-dimensional array of data, e.g., learning frequency of fractional population difference $z(t)$, ResNet is not the best choice since it typically suits for extracting features of a two-dimensional array [43, 44], such as learning $z(\Lambda, t)$ and $\phi(\Lambda, t)$ in direct DL. Here we introduce a modified Recurrent Neural network which is so-called LSTM for that purpose [23]. This method has been applied for studying knot types of polymer conformations [45] and phase-modulation stabilization in quantum key distribution [46]. LSTM is a model which could memorize the important part of the previous context in a data series and forget other irrelevant parts selectively. Therefore, it performs pretty well in time-series fitting and prediction due to the consideration of correlation in the context. Although its weakness also lies in the requirement for more time to train, in contrast to the case of using ResNet when same number of layers is required. Luckily it performs better in preventing from gradient vanishing and exploding in the long-range-series training.

Table 2 Relevant parameters under the modified LSTM network. Way of *Mixture* means that half of layers use Tanh-function and the other half layers use Sine-function. The final efficiency is given by the speed divided by the Mixture's loss. Here layers point to the number of hidden LSTM layers.

Layers	Tanh	Sine	Mixture	Time (s)	Efficiency
4	0.0220	0.0217	0.0206	20	2.43
6	**0.0119**	**0.0103**	**0.0096**	**25**	**4.17**
8	0.0103	0.0104	0.0081	32	3.86
10	0.0097	0.0099	0.0079	41	3.09
12	0.0094	0.0099	0.0078	50	2.56

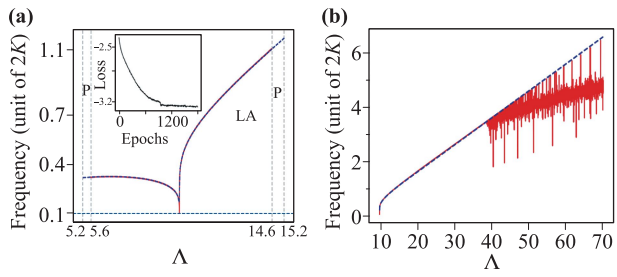

Fig. 4 **(a)** Frequency learning in the original learning area labeled by "LA": $\Lambda \in [5.6, 14.6]$. The prediction results denoted by "P" are given in the area of $\Lambda \in [5.2, 5.6]$ and $[14.6, 15.2]$. Our LSTM learning outcomes (blue-dashed) well agree with the numerical results (red-solid). Insert shows the corresponding MSE loss in the form of logarithmic function versus the training epochs. **(b)** Frequency learning under an extensive range of $\Lambda \in [10, 70]$. Here same linetypes are used.

The results of frequency learning are demonstrated in Table 2 and Fig. 4. Table 2 compares the MSE losses under different hidden layers by using various activation functions: Tanh, Sine and Mixture. For obtaining an optimal network we also consider the time expense, leading to the result of efficiency which can be used to quantitatively characterize the learning performance. We find only six hidden LSTM layers(bold) are sufficient for learning. Notice that here the definition of efficiency depends on the application of LSTM network which is different from that used in Table 1 for the case of direct DL. Besides we also use Early-stopping to avoid over-fitting. Root mean-square prop is chosen as a new optimizer in LSTM network. The original training area is the same as in direct DL i.e. $\Lambda \in [5.6, 14.6]$ and the results from an extensive range of $\Lambda \in [5.2, 15.2]$ are also given based on the prediction of LSTM. A good agreement between our learning outcomes and the numerical results is apparently observed in Fig. 4(a) except for the critical point $\Lambda_c = 10$ since the network treats it as an outlier (singularity) due to the discontinuity. Inset plots the change of MSE loss with epochs which has been reduced to be much smaller than 0.001 when epoch is larger than 1200. Therefore the way of LSTM is suitable for an individual frequency learning in the training process.

Remarkably, with the increase of Λ the oscillation of dynamics becomes stronger. The traditional numerical method is unable to give accurate results due to the calculation precision limited by a sampling interval $\Delta t = 0.02$. Reducing Δt could enhance the precision yet adding to the cost. So a bigger mismatch will appear between the numerically-estimated values and the real numbers when $\Lambda \gtrsim 40$, see the red-solid curve in Fig. 4(b). This numerical curve is obtained by computing the time duration between two same waves for extracting the information of frequency. It is apparent that the oscillating frequency becomes much larger as Λ increases which makes the fixed sampling interval inadequate. However the LSTM network can preserve a robust prediction for higher frequency values (blue-dashed, $\Lambda \in [10, 70]$) under same initial preparation. Remember the initial data set based on $\Lambda \in [5.6, 14.6]$ and $\Delta t = 0.02$ are unchanged in the learning. Therefore the LSTM network is expected to be more suitable for solving the fast-oscillating behavior even if the sampling points are inadequate or in low-frequency regime.

5.3 Modified network

A new modified network can be reconstructed by taking account into the role of LSTM in the learning frequency. Thinner networks can only capture limited local information with a small receptive field size [47]. Therefore, our method makes it possible to produce a hybrid network by combining LSTM and ResNet which is able to generate the whole population dynamics within a small receptive field size. In this hybrid learning network as shown in Fig. 5(b), the former LSTM network is used for frequency learning by the way of Poincaré map at the first stage. Additionally, the ResNet is treated as a core network for generating the whole dynamics of the fractional population difference $z(t, \Lambda)$. For the ResNet learning, we also use the MSE loss function $\mathcal{L}$ [Eq. (9)], in which the parameters $z(t_i)$, N, $g(t_i)$ are all obtained after implementing LSTM. In the core of our procedure [see red box in Fig. 5(b)], we use the Poincaré maps and LSTM network integrated to

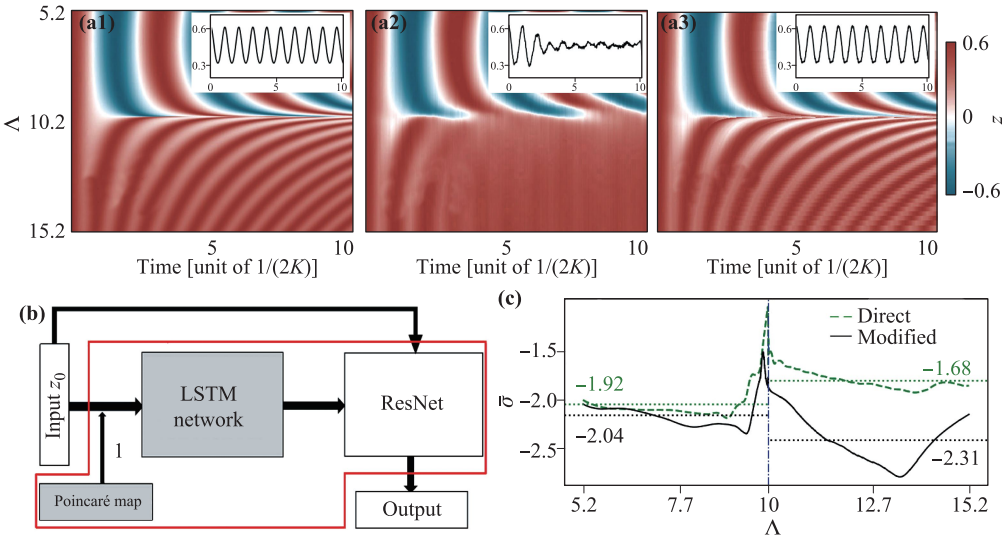

Fig. 5 Representation of the population dynamics z in the space of (t, Λ) under **(a1)** numerical simulation, **(a2)** direct DL, **(a3)** modified DL. Insets show the real population dynamics $z(t)$ for $\Lambda = 13.4$ using different methods. **(b)** The procedure of our modified network by sequentially combining LSTM and ResNet. Labeling "1" denotes the truth of dynamical periodicity verified by using *Poincaré section*, as same as the labeling "1" in Fig. 6(b). **(c)** Comparison of relative errors in logarithm form between direct DL (green dashed) and modified DL (black solid). Average values (dashed lines) are also given separately.

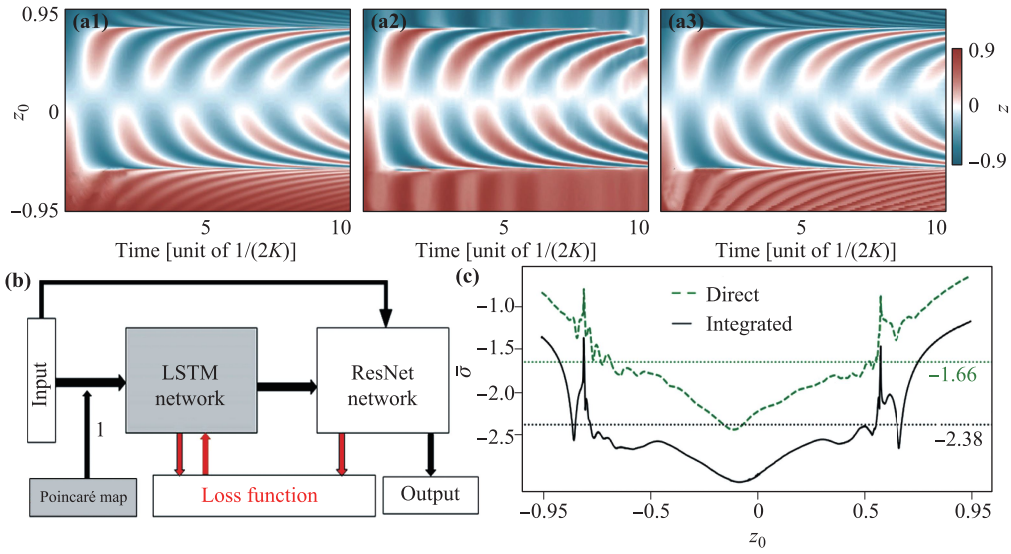

Fig. 6 **(a1–a3)** Representation of the population dynamics $z(t)$ in space of (t, z_0) using the methods of numerical simulation, direct DL and integrated DL respectively. **(b)** Integrated DL procedure. The frequency learning by LSTM is facilitated due to the use of feedback mechanism from ResNet (red arrows). **(c)** Relative errors $\bar{\sigma}$ in logarithm form versus the initial imbalanced population $z_0 = z(t = 0)$ from direct DL (dashed line) and integrated DL (solid line).

transmit extra information to ResNet. More importantly the way of Poincaré map can improve the learning fidelity via periodicity verification before learning.

Figures 5(a1)–(a3) globally present the results of $z(t, \Lambda)$ under numerical simulation, direct DL as well as our modified network learning. A special case of $z(t, 13.4)$ is shown in the insets accordingly. Clearly the direct DL method gives rise to poor results in the MQST regime as compared to the numerical calculation because of the strong oscillation. The direct DL only contains single ResNet that is insufficient for grasping the fast-oscillating behavior. However a significant improvement can be attainable if two blocks of Poincaré map and LSTM network are added in the procedure. A quantitative comparison between direct and modified DL results uses the mean relative error

$$\bar{\sigma}(\Lambda) = \frac{1}{N}\sum_{i}^{N}\frac{|z(t_i,\Lambda) - g(t_i,\Lambda)|}{0.1 + |g(t_i,\Lambda)|}, \tag{10}$$

where $z(t_i, \Lambda)$ and $g(t_i, \Lambda)$ obtained from the network training and the numerical ground-truth, are also Λ-dependent. Here 0.1 serves as an arbitrary small quantity to overcome the divergency. The relative error $\bar{\sigma}$ is a metric of the final performance standing for the derivation between learning outcome and numerical ground-truth, i.e. the infidelity of learning. While the MSE loss is an indicator at the training stage. In the BJJ region two learning ways lead to comparable errors ($10^{-1.92}$ and $10^{-2.04}$), confirming the efficiency of both learning methods. However as turning to the MQST region the modified DL apparently leads to the reduction of mean relative error from $10^{-1.68}$ to $10^{-2.31}$, which extremely increases the capability of network in learning a fast-oscillating dynamics.

By now, the hybrid system that contains a sequential LSTM and ResNet networks can provide an improved learning for the population dynamics, especially in the MQST regime. While we also note that, two networks must be trained independently suffering from a long-time expense. A typical time expense is 34 seconds for one-epoch training. To overcome this weakness we propose an integrated network by combining LSTM and ResNet together with the help of feedback mechanism. This new learning method can further accelerate the training speed, saving the time by 15% or more in an one-epoch training.

6 Integrated learning

A universal study for the population dynamics in an asymmetric double-well is performed. Motivated by Ref. [48] it is possible to utilize the feedback of loss from ResNet learning to train the former LSTM network, instead of a direct frequency learning by it, in order to accelerate the learning speed. The role of LSTM is providing the frequency information for the next ResNet learning due to the thin and small convolutional network of Resnet. Similarly, the periodicity verification is carried out by using *Poincaré section* as shown in Fig. S2. The feedback mechanism in our integrated DL scheme means that the loss function of ResNet can be treated as an indicator for the LSTM network's learning, and it would largely improve its learning efficiency. Because the MSE loss function defines the deviation between the final outcome and the ground-truth value.

In the practical training we follow the procedure dia-

Table 3 Comparison for time costs and efficiencies of four methods in the training and generation of data $z(t, \Lambda)$ in one epoch. Taking the dynamics of BECs in a symmetric double-well potential as an example, the training parameter range, the relative error and the learning efficiency are comparably given under a same number of training layers. The total layers are same for modified DL and integrated DL which include 6 hidden LSTM layers and 7 convolutional ResNet layers. For direct DL we use a total 13-layer ResNet.

Method	Training (s)	Generation (ms)	Parameter Λ	Relative error $\bar{\sigma}$	Efficiency
Runge–Kutta	–	1.15	[5.6, 14,6]	–	–
Direct learning	20	3.98	[5.2, 10]	0.0123	4.07
	20	3.98	[10, 15.2]	0.0209	2.39
Modified learning	34	5.37	[5.2, 15.2]	0.0066	4.46
Integrated learning	29	5.19	[5.2, 15.2]	0.0063	**5.47**

gram in Fig. 6(b). It differs from the modified DL mainly by using a new loss function $\mathcal{L}_{\mathcal{I}}$, which can be expressed as

$$\mathcal{L}_{\mathcal{I}} = \mathcal{L} - k \sum_{z_0}^{N^*_{z_0}} \frac{[f(z_0) - f_g(z_0)]^2}{N^*_{z_0}}, \tag{11}$$

where k is a weight factor and $N^*_{z_0}$ denotes the partial data number to train LSTM. In an asymmetric well we use the initial population difference z_0 as a tunable parameter, instead of Λ. The first term $\mathcal{L}$ standing for the loss estimation in single ResNet, takes a similar form like Eq. (9). The second term serves as an auxiliary for showing the deviation of real frequency f from the ground-truth f_g with respect to each z_0. Therefore the ResNet learning can offer an efficient feedback for the frequency learning in the LSTM network. Our scheme follows the procedure in Fig. 6(b). In order to minimize $\mathcal{L}_{\mathcal{I}}$ as well as to speed up the convergence of this integrated network, we carry out an individual training for two networks at initial steps. After that we fix the ResNet and let the LSTM being trained for steps to minimize $\mathcal{L}_{\mathcal{I}}$ accompanied by generating an array of frequency for each z_0. In addition we also simultaneously train ResNet to reduce the value of $\mathcal{L}$. Based on our practical training, we set four steps for LSTM's training and five steps for ResNet in each iteration, the weight factor is $k = 1.2$.

The integrated DL bases on the fractional population difference $z(t, z_0)$ in an asymmetric double well where $\gamma = 1.0$ is assumed. Here we set $\Lambda = 8$ and treat $z_0 = z(t = 0) \in [-0.95, 0.95]$ as a variable. After implementing a periodic verification of z with the method of Poincaré maps (see Fig. S2 in Appendix A) we do the learning following the procedure of Fig. 6(b). Figures 6(a1)–(a3) plot similar results as Figs. 5(a1)–(a3) yet within the (t, z_0) space. Clearly as compared to the direct DL, the results obtained from the integrated DL in (a3) have a better agreement with the numerical results shown in (a1). To make this discussion more quantitative, we also calculate the relative errors $\bar{\sigma}$ of two learning methods. From Fig. 6(c), it is clearly shown that the integrated DL method performs better than the direct DL for arbitrary z_0 owing to its smaller relative errors. Besides we observe that $\bar{\sigma}$ will increase if $|z_0|$ is enhanced because of the increasing frequency. In the integrated DL we only have to compute partial data for estimating the frequency i.e. $N^*_{z_0} = 1201$ (the total number is $N_{z_0} = 7601$), benefiting from the feedback mechanism. While it can reduce the relative error $\bar{\sigma}$ i.e. the infidelity, from $10^{-1.66}$ to $10^{-2.38}$ on average by using an integrated neural network.

7 Conclusion and outlook

Before ending we compare the efficiencies of three DL methods. Note that here the efficiency is defined as the training speed divided by the relative error $\bar{\sigma}$, not the loss function. Given the original data are obtained by solving Eq. (4) we display the learning results of various methods in Table 3. The original data set for $z(t)$ is generated by setting $\Lambda \in [5.6, 14.6]$. Although the training process costs much longer time than the generation it is possible to predict unknown results beyond the given parameter range. Note that here a direct DL with a total 13-layer ResNet fails to fit the real population dynamics on the MQST regime, leading to a poor efficiency ~ 2.39 (the infidelity attains 0.0209 when $\Lambda \in [10, 15.2]$). However, accompanied by an auxiliary LSTM network that provides a certification of the oscillating frequencies, the resulting hybrid network (modified DL or integrated DL) has shown a significant improvement for the learning efficiency. The efficiency reaches as high as ~ 5.47 (the final infidelity obtained for $z(t)$ is far smaller than 0.01) for the integrated DL. Besides the hybrid network can also provide a precise learning in both BJJ and MQST regimes, and can even make convincing predictions about data with respect to the unknown parameters. When turning to the case of learning higher-frequency dynamics such as in the MQST regime, we find that the Runge–Kutta method becomes inaccessible due to the finite sampling interval. Instead, our LSTM network reveals an ability of making predictions about higher frequency values based on the same data set.

In conclusion we develop a hybrid "*LSTM+ResNet*" network to revisit the population and phase dynamics of two BECs in a double-well potential. Because the fast-

oscillating behavior of population difference in the MQST regime makes the traditional direct DL inaccessible, we propose a novel learning method which integrates LSTM and ResNet networks and then brings about a strong promotion to the learning efficiency. The LSTM also represents a robust prediction for the characteristic periodicities of data sets. This hybrid network performs better with a high precision and a very low relative error (the final infidelity obtained for the fractional population difference) ~ 0.006. Most importantly, it shows a strong ability of prediction for the data in the fast-oscillating regime which are far beyond the original data set given by the exact numerical calculations in the slow-oscillating regime. This prediction capability of method could find more applications in solving problems of physics and other sciences, which are lack of sufficient parameters restrained by the experimental conditions or costs.

As an outlook we discuss how our hybrid network could work for other systems possessing more complex population dynamics which cannot be analytically solved. For example, when the effective three-body interactions of BECs are not negligible there is no longer an analytical solution of the double-well dynamics and the evolution of population difference possesses complex multi-frequency oscillations [49]. Then a Fourier frequency analysis combined with two algorithms, *Attention* and *Sparsity* can help us to realize efficient learning. If there exists several frequency components from the Fourier transformed data, *Attention*, which is a method in computer science to assign weights for values, will learn the weight of each frequency in the multi-frequency oscillating dynamics. However, if the exact frequencies are unknown, we have to put a series of candidates into the set. Thus, the frequency set may have many irrelevant elements, making the learning outcome polluted. Serving as an enhancement trick for the Attention mechanism, *Sparsity* ensures the sparsity of used frequencies, which can make the selected frequencies more effective and reduce the complexity and time expense for learning [50]. In short, with the aid of *Attention*, we can obtain the weight of each frequency component and reconstruct them into the final outcome where *Sparsity* may help learning in a continuous parameter system. Finally, remember that our DL method depends on a data set to train. This data set can be generated by solving theoretical equations. Even if the problem is analytically unsolvable the sample data can still be collected from experimental measurements. So our method not only applies to theoretical simulations but also to experimental estimation. Moreover, our integrated DL method is not restricted to a sufficient sampling which means an extensive application in various systems is desirable in the future.

Acknowledgements This work was supported by the NSFC under Grant Nos. 12174106, 11474094, 11104076, and 11654005; the Science and Technology Commission of Shanghai Municipality under Grant No. 18ZR1412800; the National Key Research and Development Program of China under Grant No. 2016YFA0302001; the Shanghai Municipal Science and Technology Major Project under Grant No. 2019SHZDZX01, the Shanghai talent program.

Appendix A Numerical results of Poincaré section

Some specific examples of Poincaré section for verifying the periodicity of population dynamics $z(t)$, are given in Figs. S1 and S2.

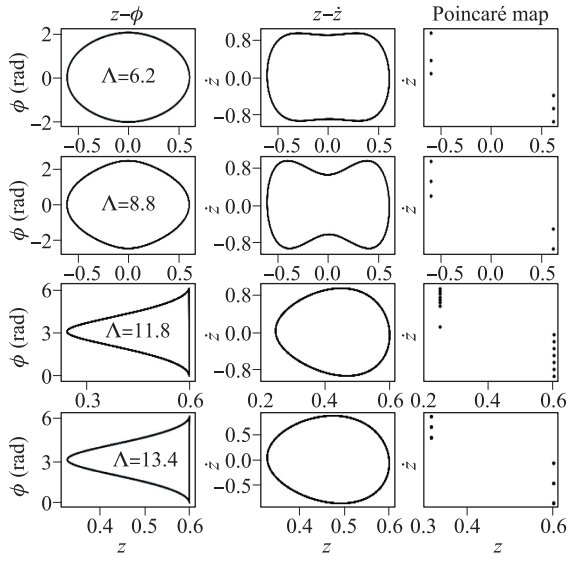

Fig. S1 Poincaré section for a symmetric double-well: Diagrams of z–ϕ (left), z–$\dot{z}$ (middle), Poincaré maps (right). From top to bottom $\Lambda = (6.2, 8.8, 11.8, 13.4)$, respectively.

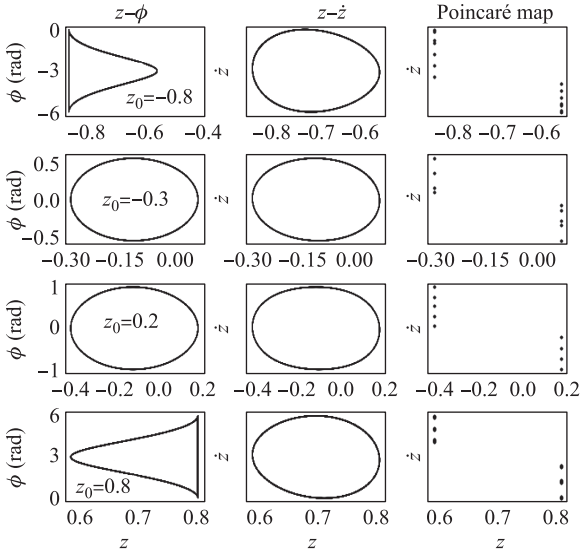

Fig. S2 Poincaré section for an asymmetric double-well: Diagrams of z–ϕ (left), z–$\dot{z}$ (middle), Poincaré maps (right). From top to bottom $z_0 = (-0.8, -0.3, 0.2, 0.8)$, respectively.

References

1. I. A. Luchnikov, S. V. Vintskevich, D. A. Grigoriev, and S. N. Filippov, Machine learning non-Markovian quantum dynamics, *Phys. Rev. Lett.* 124(14), 140502 (2020)
2. X. Liu, G. Zhang, J. Li, G. Shi, M. Zhou, B. Huang, Y. Tang, X. Song, and W. Yang, Deep learning for Feynman's path integral in strong-field time-dependent dynamics, *Phys. Rev. Lett.* 124(11), 113202 (2020)
3. G. Carleo and M. Troyer, Solving the quantum many body problem with artificial neural networks, *Science* 355(6325), 602 (2017)
4. A. Hentschel and B. C. Sanders, Machine learning for precise quantum measurement, *Phys. Rev. Lett.* 104(6), 063603 (2010)
5. P. Zhang, H. Shen, and H. Zhai, Machine learning topological invariants with neural networks, *Phys. Rev. Lett.* 120(6), 066401 (2018)
6. G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, Neural-network quantum state tomography, *Nat. Phys.* 14(5), 447 (2018)
7. K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, Machine learning phases of strongly correlated fermions, *Phys. Rev. X* 7(3), 031038 (2017)
8. E. van Nieuwenburg, Y. Liu, and S. D. Huber, Learning phase transitions by confusion, *Nat. Phys.* 13(5), 435 (2017)
9. J. Carrasquilla and R. Melko, Machine learning phases of matter, *Nat. Phys.* 13(5), 431 (2017)
10. S. Lu, L. Duan, and D. Deng, Quantum adversarial machine learning, *Phys. Rev. Research* 2(3), 033212 (2020)
11. X. Ouyang, X. Huang, Y. Wu, W. Zhang, X. Wang, H. Zhang, L. He, X. Chang, and L. Duan, Experimental demonstration of quantum-enhanced machine learning in a nitrogen-vacancy-center system, *Phys. Rev. A* 101(1), 012307 (2020)
12. V. Dunjko, J. M. Taylor, and H. J. Briegel, Quantum enhanced machine learning, *Phys. Rev. Lett.* 117(13), 130501 (2016)
13. G. Cybenko, Approximation by superpositions of a sigmoidal function, *Math. Contr. Signals Syst.* 2(4), 303 (1989)
14. L. Ba and R. Caruana, Do deep nets really need to be deep? *Adv. Neural Inf. Process. Syst.* 3, 2654 (2014)
15. D. S. P. Salazar, Nonequilibrium thermodynamics of restricted Boltzmann machines, *Phys. Rev. E* 96(2), 022131 (2017)
16. G. Hinton, S. Osindero, and Y. W. Teh, A fast learning algorithm for deep belief nets, *Neural Comput.* 18(7), 1527 (2006)
17. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE* 86(11), 2278 (1998)
18. M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, Recurrent neural network wave functions, *Phys. Rev. Research* 2(2), 023358 (2020)
19. A. Diaz Rivero and C. Dvorkin, Direct detection of dark matter substructure in strong lens images with convolutional neural networks, *Phys. Rev. D* 101(2), 023515 (2020)
20. M. K. Mulimani, J. K. Alageshan, and R. Pandit, Deep learning-assisted detection and termination of spiral and broken-spiral waves in mathematical models for cardiac tissue, *Phys. Rev. Research* 2(2), 023155 (2020)
21. V. Novičenko, J. Ruseckas, and E. Anisimovas, Quantum dynamics in potentials with fast spatial oscillations, *Phys. Rev. A* 99(4), 043608 (2019)
22. C. E. Shannon, Communication in the presence of noise, *Proceedings of the IRE* 37(1), 10 (1949)
23. S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Comput.* 9(8), 1735 (1997)
24. K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770 (2016)
25. X. Liang, H. Zhang, S. Liu, Y. Li, and Y. S. Zhang, Generation of Bose–Einstein condensates' ground state through machine learning, *Sci. Rep.* 8, 16337 (2018)
26. G. Spagnolli, G. Semeghini, L. Masi, G. Ferioli, A. Trenkwalder, S. Coop, M. Landini, L. Pezzè, G. Modugno, M. Inguscio, A. Smerzi, and M. Fattori, Crossing over from attractive to repulsive interactions in a tunneling bosonic Josephson junction, *Phys. Rev. Lett.* 118(23), 230403 (2017)
27. M. Pigneur and J. Schmiedmayer, Analytical pendulum model for a bosonic Josephson junction, *Phys. Rev. A* 98(6), 063632 (2018)
28. S. Martínez-Garaot, G. Pettini, and M. Modugno, Nonlinear mixing of Bogoliubov modes in a bosonic Josephson junction, *Phys. Rev. A* 98(4), 043624 (2018)
29. J. Dobrzyniecki, and T. Sowiński, Effective two-mode description of a few ultra-cold bosons in a double-well potential, *Phys. Lett. A* 382(6), 394 (2018)
30. G. Valtolina, A. Burchianti, A. Amico, E. Neri, K. Xhani, J. A. Seman, A. Trombettoni, A. Smerzi, M. Zaccanti, M. Inguscio, and G. Roati, Josephson effect in fermionic superfluids across the BEC–BCS crossover, *Science* 350(6267), 1505 (2015)
31. M. Jääskeläinen and P. Meystre, Dynamics of Bose–Einstein condensates in double-well potentials, *Phys. Rev. A* 71(4), 043603 (2005)
32. N. R. Thomas, A. C. Wilson, and C. J. Foot, Double-well magnetic trap for Bose–Einstein condensates, *Phys. Rev. A* 65(6), 063406 (2002)
33. R. W. Spekkens and J. E. Sipe, Spatial fragmentation of a Bose–Einstein condensate in a double-well potential, *Phys. Rev. A* 59(5), 3868 (1999)
34. H. M. Cataldo and D. M. Jezek, Dynamics in asymmetric double-well condensates, *Phys. Rev. A* 90(4), 043610 (2014)
35. Y. Shin, M. Saba, T. A. Pasquini, W. Ketterle, D. E. Pritchard, and A. E. Leanhardt, Atom interferometry with Bose–Einstein condensates in a double-well potential, *Phys. Rev. Lett.* 92(5), 050405 (2004)

36. M. Albiez, R. Gati, J. Fölling, S. Hunsmann, M. Cristiani, and M. K. Oberthaler, Direct observation of tunneling and nonlinear self-trapping in a single bosonic Josephson junction, *Phys. Rev. Lett.* 95(1), 010402 (2005)
37. A. Smerzi, S. Fantoni, S. Giovanazzi, and S. R. Shenoy, Quantum coherent atomic tunneling between two trapped Bose–Einstein condensates, *Phys. Rev. Lett.* 79(25), 4950 (1997)
38. D. P. Kingma and J. L. Ba, Adam: A method for stochastic optimization, arXiv: 1412.6980 (2014)
39. J. D. Scargle, Studies in astronomical time series Analysis (iii) — Fourier transforms, autocorrelation functions, and cross-correlation functions of unevenly spaced data, *Astrophys. J.* 343, 874 (1989)
40. D. O. Pushkin, D. E. Melnikov, and V. M. Shevtsova, Ordering of small particles in one-dimensional coherent structures by time-periodic flows, *Phys. Rev. Lett.* 106(23), 234501 (2011)
41. F. Tajik, M. Sedighi, M. Khorrami, A. A. Masoudi, and G. Palasantzas, Chaotic behavior in Casimir oscillators: A case study for phase-change materials, *Phys. Rev. E* 96(4), 042215 (2017)
42. Z. Liu, Chaotic Dynamics Foundation and Its Application in Brain Functions, Beijing: Science Press, 2018
43. S. Efthymiou, M. J. S. Beach, and R. G. Melko, Super-resolving the Ising model with convolutional neural networks, *Phys. Rev. B* 99(7), 075113 (2019)
44. Z. Li, S. Wu, J. Gao, H. Zhou, Z. Yan, R. Ren, S. Yin, and X. Jin, Fast correlated-photon imaging enhanced by deep learning, *Optica* 8(3), 323 (2021)
45. O. Vandans, K. Yang, Z. Wu, and L. Dai, Identifying knot types of polymer conformations by machine learning, *Phys. Rev. E* 101(2), 022502 (2020)
46. J. Liu, H. Ding, C. Zhang, S. Xie, and Q. Wang, Practical phase-modulation stabilization in quantum key distribution via machine learning, *Phys. Rev. Appl.* 12(1), 014059 (2019)
47. F. Milletari, N. Navab, and S. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 Fourth International Conference on 3D Vision (3DV), pp 565–571 (2016)
48. L. Xie, Q. Yu, Y. Zhou, Y. Wang, E. K. Fishman, and A. L. Yuille, Recurrent saliency transformation network for tiny target segmentation in abdominal CT scans, *IEEE Trans. Med. Imaging* 39(2), 514 (2020)
49. J. Dobrzyniecki, X. Li, A. E. B. Nielsen, and T. Sowiński, Effective three-body interactions for bosons in a double well confinement, *Phys. Rev. A* 97(1), 013609 (2018)
50. B. Yun, Y. Wang, J. Chen, H. Wang, W. Shen, and Q. Li, SpecTr: Spectral transformer for hyperspectral pathology image segmentation, arXiv: 2103.03604 (2021)