Machine learning topological invariants of non-Hermitian systems

Ling-Feng Zhang,1 Ling-Zhi Tang,1 Zhi-Hao Huang,1 Guo-Qing Zhang,1,2, * Wei Huang,1 and Dan-Wei Zhang1,2, †

1Guangdong Provincial Key Laboratory of Quantum Engineering and Quantum Materials, GPETR Center for Quantum Precision Measurement and SPTE, South China Normal University, Guangzhou 510006, China
2Frontier Research Institute for Physics, South China Normal University, Guangzhou 510006, China
(Dated: September 16, 2020)

The study of topological properties by machine learning approaches has attracted considerable interest recently. Here we propose machine learning the topological invariants that are unique in non-Hermitian systems. Specifically, we train neural networks to predict the winding of eigenvalues of three different non-Hermitian Hamiltonians on the complex energy plane with nearly 100% accuracy. Our demonstrations on the Hatano-Nelson model, the non-Hermitian Su-Schrieffer-Heeger model and generalized Aubry-André-Harper model show the capability of the neural networks in exploring topological invariants and the associated topological phase transitions and topological phase diagrams in non-Hermitian systems. Moreover, the neural networks trained by a small data set in the phase diagram can successfully predict topological invariants in untouched phase regions. Thus, our work pave a way to reveal non-Hermitian topology with the machine learning toolbox.

I. INTRODUCTION

Machine learning, which lies at the core of the artificial intelligence and data science, has recently achieved huge success from industrial applications (especially in computer vision and natural language process) to fundamental researches in physics, cheminformatics and biology [1–4]. In physics, machine learning has shown its availability in experimental data analysis [5–7] and classification of phases of matter [8–23]. Among these applications, one of the most interesting problems is to extract the global properties of topological phases of matter from local inputs, such as the topological invariants that intrinsically nonlocal. Recent works have shown that artificial neural networks can be trained to predict the topological invariants of band insulators with high accuracy [16, 17]. The advantage of this approach is that the neural network can capture global topology directly from local raw data inputs. Other theoretical proposals for using supervised or unsupervised learning in identifying topological phases have been suggested [15, 18, 21–26]. Notably, the convolutional neural network (CNN) trained from raw experimental data has been demonstrated to identify topological phases [27, 28].

On the other hand, growing efforts have been invested in uncovering exotic topological states and phenomena in non-Hermitian systems in recent years [29–68]. The non-Hermiticity may come from the gain and loss effects [36–40], the non-reciprocal hopping [46, 47], or the dissipation in open systems [29, 30]. The non-Hermiticity-induced topological phases are also investigated in disordered [53–62] and interacting systems [63–68]. In the non-Hermitian topological systems, there are not only topological properties defined by the eigenstates (such as topological Bloch bands), but also topological invariants lying on solely the eigenenergies. For instance, the complex energy landscapes (and exceptional points) give rise to new topological invariants, which include the winding number (vorticity) defined solely in the complex energy plane [48–51]. This winding number and several closely related winding numbers in the presence of other symmetries lead to richer topological classification than that of their Hermitian counterparts. In addition, it was revealed [69–71] that the nonzero winding number in the complex energy plane is the topological origin of the non-Hermitian skin effect [31–35]. In view that the topological invariants in Hermitian systems have been recently studied based on the machine learning approach [15–18, 21–26], the flexibility of machine learning the new kind of winding numbers in non-Hermitian systems is an urgent and meaningful research.

In this work, we adapt the machine learning with artificial neural networks to predict non-Hermitian topological invariants and classify the topological phases in three non-Hermitian models. We first take the Hatano-Nelson model [46, 47] as a feasibility verification of the machine learning method in identifying non-Hermitian topological phases. We show that the trained CNN can predict the winding numbers even for those phases that are not included in the training with high accuracy, whereas the fully connected neural network (FCNN) can only predict winding numbers in the trained phases. We interpolate the intermediate value of the CNN and find a strong relationship with the winding angle of the eigenenergies in the complex plane. We then use the CNN to study topological phase transitions in a non-Hermitian Su-Schrieffer-Heeger (SSH) model [72] with non-reciprocal hopping. We find that the CNN can precisely detect the transition points near the boundary of each phase even though trained only by the data in the deep phase region. Finally, by using the CNN, we obtain the topological phase diagram of a non-Hermitian generalized Aubry-André-Harper (AAH) model [73–75] with...
non-reciprocal hopping and complex quasiperiodic potential. The winding numbers evaluated from the CNN show an accuracy of more than 99% with theoretical values in the whole parameter space, even though the complex on-site potential is absent in the training process. Our work provides an efficient and general approach to reveal non-Hermitian topology based on the machine learning.

The rest of this paper is organized as follows. We first study the winding number of the Hatano-Nelson model as a feasibility verification of our machine learning method in Sec. II. Different performances of the CNN and the FCNN are also discussed. Section III is devoted to reveal the topological phase transition in the non-Hermitian SSH model by the CNN. In Sec. IV, we show the CNN can precisely predict the topological phase diagram of the non-Hermitian generalized AAH model. A short summary is finally presented in Sec. V.

II. LEARNING TOPOLOGICAL INVARIANTS IN HATANO-NELSON MODEL

Let us begin with the Hatano-Nelson model, which can be considered as the simplest single-band non-Hermitian model. The Hatano-Nelson model takes the following Hamiltonian in a one-dimensional lattice of length $L$

$$H_1 = \sum_{j}^{L} (t_r \hat{c}_j \hat{c}_{j+\mu} + t_l \hat{c}_j \hat{c}_{j+\mu} + V_j \hat{c}_j \hat{c}_j),$$

where $t_\mu \neq t^{*}_\mu$ denotes the amplitudes of non-reciprocal hopping, $\hat{c}_j$ is creation (annihilation) operator on the $j$-th lattice site, $V_j$ denotes the hopping length between two sites, and $V_j$ is the on-site energy in the lattice. The original Hatano-Nelson model takes the disorder potential with random $V_j$ and the nearest-neighbour hopping with $\mu = 1$, as shown in Fig. 1(a). Here we consider the clean case by setting $V_j = 0$ and take $\mu$ as a parameter in learning the topological phase transition with neural networks. Under the periodical boundary condition, the corresponding eigenenergies in this case are given by

$$E_1(k) = H_1(k) = t_r e^{-i\mu k} + t_l e^{i\mu k},$$

where $H_1(k)$ is the Hamiltonian in momentum space with the quasimomenta $k = 0, 2\pi/L, 4\pi/L, \cdots, 2\pi$.

Following Ref. [48], we can define the winding number in the complex energy place as a topological invariant in the Hatano-Nelson model:

$$w = \frac{1}{2\pi} \int_{0}^{2\pi} \frac{dk}{2\pi} \partial_k \ln \det H_1(k)$$

$$= \frac{1}{2\pi} \int_{0}^{2\pi} \frac{dk}{2\pi} \partial_k \arg E_1(k) = \left\{ \begin{array}{ll} \mu & |t_r| < |t_l|; \\ -\mu & |t_r| > |t_l|, \end{array} \right.$$

where $\arg$ denotes the principal value of the argument belonging to $[0, 2\pi)$. For discretized $E_1(k)$ with finite lattice site $L$, the complex-energy winding number reduces to

$$w = \frac{1}{2\pi} \sum_{n=1}^{L} \Delta \theta(n) = \frac{1}{2\pi} \sum_{n=1}^{L} [\theta(n) - \theta(n - 1)],$$

where $\theta(n) = \arg E_1(2\pi n/L)$. Note that for Hermitian systems ($t_r = t^{*}_r$), one has $w = 0$ due to the real energy spectrum with $\arg E_1(k) = 0, \pi$. According to this definition, a nontrivial winding number in this model gives the number of times the complex eigenenergy encircles the base point $E_B = 0$, which is unique to non-Hermitian systems. The complex eigenenergy windings of the two cases with $w = \pm 1$ for the original Hatano-Nelson are shown in Fig. 1(b). To examine whether the neural networks have the ability to learn the winding number in a general formalism, we enable the parameter $\mu$ to control the number of times of complex eigenenergy encircles the origin of the complex plane. When the loop winds around the origin $\mu$ times during the variation of $k$ from 0 to $2\pi$, the winding number is $\pm \mu$, where $\pm$ means the counterclockwise and clockwise windings, respectively.

We now build a supervised task for learning the winding number given by Eq. (4) based on neural networks. First, we need labeled data sets for training and evaluation. Since the winding number is intrinsically nonlocal and characterized by the complex energy spectrum, we feed neural networks with the normalized spectrum-dependent configurations $\mathbf{d}(n) = [\mathbf{d}_R(n), \mathbf{d}_I(n)]$ at $L$ points discretized uniformly from 0 to $2\pi$, where $\mathbf{d}_R(n) = \text{Re}[E_1(2\pi n/L)]$ and $\mathbf{d}_I(n) = \text{Im}[E_1(2\pi n/L)]$. Therefore, the input data is a $(L + 1) \times 2$-dimensional matrix of the form

$$[\mathbf{d}_R(0) \mathbf{d}_R(2\pi/L) \mathbf{d}_R(4\pi/L) \cdots \mathbf{d}_R(2\pi)]^T$$

with a period of $2\pi$: $\mathbf{d}(n) = \mathbf{d}(n + 2\pi)$. In the following, we set $L = 32$, which is large enough to take discrete energy spectra as the input data of neural networks. La-
FIG. 2. (Color online) Schematic of machine learning workflow and the structure of neural networks for the Hatano-Nelson (denoted by HN) model, non-Hermitian SSH (denoted by NHSSH) model, and non-Hermitian generalized AAH (denoted by NHGAAH) model. The input data are represented by \((L + 1) \times 2\)-dimensional matrix for the CNN and \(2 \times (L + 1)\)-dimensional vector for the FCNN, respectively. Here \(d_R\) and \(d_I\) denote the real and imaginary parts of input data (complex eigenenergies), respectively.

The learning workflow is schematically shown in Fig. 2. For the Hatano-Nelson model with different \(\mu\), the output of the neural network is a real number \(\tilde{w}\), and the predicted winding number is interpreted as the integer that is closed to \(\tilde{w}\). We first train the neural networks with both complex-spectrum-dependent configurations and their corresponding true winding numbers. After the training, we feed only the complex-spectrum-dependent configurations to the neural networks and compare their predictions with the true winding numbers, from which we determine the percentage of the correct predictions as the accuracy. In this case, we consider two classes of neural networks: the CNN and FCNN, respectively. The neural networks are similar as those in Ref. [16] for calculating the winding number of the Bloch vectors in Hermitian topological bands.

The CNN has two convolution layers with 32 kernels of size \(1 \times 2 \times 2\) and 1 kernel of size \(32 \times 1 \times 1\), followed by a fully connected layer of 2 neurons before output layer. The total number of trainable parameters is 262. The FCNN has two hidden layers with 32 and 2 neurons, respectively. The total number of trainable parameters is 2213. The architecture of two classes of neural networks is shown in Fig. 2. All the hidden layers have rectified linear units \(f(x) = \max(0, x)\) as activation functions and the output layer has linear activation function \(f(x) = x\). The objective function to be optimized is defined by

\[
J_1 = \frac{1}{N} \sum_{i=1}^{N} (\tilde{w}_i - w_i)^2,
\]

where \(\tilde{w}_i\) and \(w_i\) are respectively the winding number of the \(i\)th complex eigenenergies predicted by the neural networks and the true winding number, and \(N\) is the total number of the training data set. We take \(6 \times 10^4\) training configurations, which consists of a ratio of \(1 : 1 : 1\) of them having winding number \(\{\pm 1, \pm 2, \pm 3\}\), respectively. Test set consists of some configurations with winding numbers \(w \in \{\pm 1, \pm 2, \pm 3\}\) that are not included in the training set and \(w \in \{\pm 4, \pm 5\}\) that are not seen by neural networks during the training. The number of configurations in each kind of winding number is \(4 \times 10^3\). The training details are given in the Appendix A.

After training, we test with other configurations and the predicted winding number \(\tilde{w}\) are shown in Fig. 3 (a). Note that the networks tend to produce \(\tilde{w}\) close to integers and thus we take each final winding number as the integer closed to \(\tilde{w}\). As shown in Fig. 3 (b), we plot the probability distribution of \(\tilde{w}\) predicted from the CNN on different test data sets. The test results of two neural networks are presented in Table. 1, which shows very high
accuracy (more than 98%) of the CNN and FCNN on test data set with the winding number \( w = \{ \pm 1, \pm 2, \pm 3 \} \). We can find that the CNN performs generally better than the FCNN. Surprisingly, the CNN works well even in the cases of \( w = \{ \pm 4, \pm 5 \} \), which consist of configurations with larger winding numbers not seen by neural networks during the training. On the contrary, the FCNN cannot predict the true winding number even though has more trainable parameters. These results indicate that the convolutional layer respects the translation symmetry of complex spectrum in the momentum space explicitly and convolutional layers can take local winding \( \Delta \theta \) explicitly through the \( 2 \times 2 \) kernels.

To further see the advantage of the CNN, we open up the black box of neural networks through finding the relationship between intermediate activation values and physical quantities, i.e. the winding angle \( \Delta \theta \). Based on the convolutional layers, we consider the activation value after two convolution should have a linear dependence on \( \Delta \theta \) at some extent and the following fully-connected layers use simple linear regression. We plot \( a_n \) versus \( \Delta \theta(n) \) with \( n = 1, \ldots, L \) and \( a_n \) being the \( n \)-th component of intermediate values after two convolution layers. As shown in Fig. 3 (c), the intermediate output is approximately linear with \( \Delta \theta \) within certain regions. A linear combination of these intermediate values with correct coefficients in the following fully-connected layers can then easily lead to the true winding number. In this way, the CNN realizes a calculation workflow that is equivalent to the winding angle \( \Delta \theta \) in Eq. (4).

### TABLE I. The accuracy of the CNN and FCNN on test data set with the winding number \( w = \{ \pm 1, \pm 2, \pm 3, \pm 4, \pm 5 \} \) in the Hatano-Nelson model with \( \mu = 1, 2, 3, 4, 5 \). The winding number \( w = \{ \pm 4, \pm 5 \} \) are not seen by the neural networks during the training.

|       | \( \pm 1 \) | \( \pm 2 \) | \( \pm 3 \) | \( \pm 4 \) | \( \pm 5 \) |
|-------|-------------|-------------|-------------|-------------|-------------|
| CNN Accuracy | 99.8% | 99.4% | 98.0% | 96.7% | 96.0% |
| FCNN Accuracy | 99.2% | 99.0% | 98.5% | 0.0% | 0.0% |

III. LEARNING TOPOLOGICAL TRANSITION IN NON-HERMITIAN SSH MODEL

Based on the accurate winding number calculated by the CNN, we further use similar CNN to study topological phase transitions in a non-Hermitian SSH model, as shown in Fig. 4(a). The considered model with nonreciprocal intra-cell hopping in the one-dimensional dimerized lattice of \( L \) unit cells can be described by the following Hamiltonian

\[
H_2 = \sum_{n=1}^{L} \left[ (t-\delta)a_n^\dagger \hat{b}_n + (t+\delta)b_n^\dagger a_n + t' a_{n+1}^\dagger \hat{b}_n + t' \hat{b}_n^\dagger a_{n+1} \right].
\]  

(6)
Here \(\hat{a}_n^\dagger\) and \(\hat{b}_n\) (\(\hat{a}_n, \hat{b}_n\)) denote the creation (annihilation) operators on the \(n\)-th \(A\) and \(B\) sublattices, \(t\) is the uniform intra-cell hopping amplitude, \(\delta\) is the non-Hermitian parameter, \(t'\) is the inter-cell hopping amplitude. When \(\delta = 0\), the model reduces to the Hermitian SSH model. Under the periodic boundary condition, the corresponding Hamiltonian in \(k\) space is given by

\[
H_2(k) = \begin{pmatrix}
0 & t' e^{-i k} + t - \delta \\
t' e^{i k} + t + \delta & 0
\end{pmatrix}.
\]

(7)

The two energy bands are then given by

\[
E_{\pm}(k) = \pm \sqrt{1 + t^2 - \delta^2 + 2t \cos k - i2 \delta \sin k}.
\]

(8)

Following Ref. [48-51] and considering the chiral symmetry, one can define an inter-band winding number

\[
w_{\pm} = \int_0^{2\pi} \frac{dk}{2\pi} \partial_k \arg(E_{+} - E_{-}) = \int_0^{2\pi} \frac{dk}{4\pi} \partial_k \arg E_{+}^2.
\]

(9)

For discretized \(E_{\pm}(k)\) with finite \(L\), it reduces to

\[
w_{\pm} = \frac{1}{4\pi} \sum_{n=1}^{L} [\theta'(n) - \theta'(n-1)]
\]

(10)

with \(\theta'(n) = \arg E_{+}^2(2\pi n/L)\) in this model. Notable, \(w_{\pm}\) is half of the summing the winding of number of \((t' e^{-i k} + t - \delta)\) and \((t' e^{i k} + t + \delta)\) around the origin of the complex plane as \(k\) is increased from 0 to \(2\pi\). The interband winding number \(w_{\pm}\) is quantized as \(\mathbb{Z}/2\) because the winding of \((t' e^{-i k} + t - \delta)\) and \((t' e^{i k} + t + \delta)\) are always integers due to periodicity [51]. We consider \(t' = 1\), \(t \in (-6, 6)\), and \(\delta \in (-6, 6)\) in our study.

For this model, we set the configuration of input data as \(d(n) = \{\text{Re}[E_{+}^2(2\pi n/L)], \text{Im}[E_{+}^2(2\pi n/L)]\}\). To learn the topological phase transition in this model, we treat it as a classification task assisted by neural networks. The output of neural network is the probabilities of different winding numbers. We define \(\{P_1, P_2, P_3\}\) as the output probabilities of winding number \(\hat{w}_{\pm}\) \(= \{0, 0.5, -0.5\}\), respectively. The predicted winding number is interpreted as the \(\hat{w}_{\pm}\), which has the highest probability. The architecture of the CNN is shown in Fig. 2, with some training details are given in the Appendix A. For our task, the objective function to be optimized is defined by

\[
J_2 = -\frac{1}{N} \sum_{i=1}^{N} \sum_{n=3}^{n_w} 1(w_{\pm}^{(i)} = \hat{w}_{\pm,j}) \log_2(P_j)),
\]

(11)

where \(w_{\pm}^{(i)}\) is the label of the \(i\)-th configuration, and the set \(\{\hat{w}_{\pm,1}, \hat{w}_{\pm,2}, ..., \hat{w}_{\pm,n_w}\}\) represents the winding number predicted by the neural networks. The expression \(1(w_{\pm}^{(i)} = \hat{w}_{\pm,j})\) means that it will take the value 1 when the condition \(w_{\pm}^{(i)} = \hat{w}_{\pm,j}\) is satisfied and 0 for the opposite case. In this model, \(n_w = 3\) and \(\{\hat{w}_{\pm,1}, \hat{w}_{\pm,2}, \hat{w}_{\pm,3}\}\) represent the winding number \(w = \{0, 0.5, -0.5\}\) correspondingly.

To see whether the CNN is a good tool to study topological phases transitions in this model, we define a Euclidean distance \(s\) between the configuration and the phase boundaries in parameter space of the Hamiltonian:

\[
s = \frac{|A\delta + Bt + C|}{\sqrt{A^2 + B^2}},
\]

(12)

where \(A\delta + Bt + C = 0\) (straight lines in parameters space about \(\delta\) and \(t\)) is the equations of phase boundaries with \(A, B, C\) being the parameters of the equation. In addition, we define a distance threshold \(T\). In the following, we choose \(T = 0.2\) as a demonstration and situation of \(0.2 < T \leq 0.6\) will be discussed later. Training data set consists of \(2.4 \times 10^4\) configurations satisfying \(s \geq T\) are sampled from different phases with different winding numbers.
We test the CNN with two different test data sets: (I) 6 × 10³ configurations satisfying s < T; (II) 300 configurations distributed uniformly in t = 0.5, δ = [−3, 3]. The data sets distribution and some training details are given in the Appendix A. After training, both test data set I and test data set II are evaluated by the CNN. We use the same training and test workflow for T = 0.3, 0.4, 0.5, 0.6. Fig. 4(b) shows the accuracy of the test data sets against the distance threshold T. We find that the CNN achieves high accuracy in different T, meaning that the CNN can detect the phase transitions precisely in these regions. Moreover, we locate the phase transition points from the crossing points of prediction probabilities, the phase transitions determined by this method is relatively accurate, as shown in the Fig. 4(c). At the deep phase, the probability for the true winding number w± stays at nearly 100%. On the other hand, the probability for w± rises straightly at the phase transitions. In a words, the CNN is a great supplementary tool to study the phase transitions when only phase properties in some confident regions (e.g. the deep phase) are provided.

IV. LEARNING TOPOLOGICAL PHASE DIAGRAM IN NON-HERMITIAN AAH MODEL

To show that our results can be generalized to other non-Hermitian topological models, we consider a generalized AAH model in a one-dimensional quasicrystal as shown in Fig. 5(a), with two kinds of non-Hermiticites arising from the non-reciprocal hopping [55] and complex on-site potential phase [56]. The Hamiltonian of such a non-Hermitian AAH model is given by [76]

\[ H_3(\Phi) = \begin{pmatrix} \Delta_1 & t_1^{(l)} & t_L^{(r)} e^{-i\Phi} \\ t_1^{(r)} & \Delta_2 & t_2^{(l)} \\ \vdots & \vdots & \vdots \\ t_L^{(l)} e^{i\Phi} & \Delta_{L-1} & t_{L-1}^{(r)} \end{pmatrix}. \]  

(15)

One can define the winding number with respect to \( \Phi \) and the energy base \( E_B \) [48, 55]:

\[ w_\Phi = \frac{1}{2\pi} \sum_{n=1}^{L-\Phi} [\theta''(n) - \theta''(n - 1)], \]

(16)

where \( \theta''(n) = \arg \det[H_3(2\pi n/\Phi) - E_B] \).

Below we show that the generalization ability enables the CNN to precisely obtain topological phase diagrams of this non-Hermitian generalized AAH model, even though we only use non-reciprocal-hopping configurations in the training. To do this, we treat the problem as a classification task and set the configuration in this case as \( d(n) = \{ \Re \det[H_3(n)], \Im \det[H_3(n)] \} \) with \( \tilde{H}_3(n) = H_3(2\pi n/\Phi) - E_B \). The architecture of the CNN is similar to that for the non-Hermitian SSH model, but the output layer now becomes two neurons for two kinds of winding number. We define \( \{ P_1, P_2 \} \) as the output probabilities of the winding numbers \( w_\Phi = \{0, -1\} \), respectively. The objective function in this case is similar to that in Eq. (11) and is given by

\[ J_3 = -\frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{n_w=2} 1[\tilde{w}_{\Phi,j} = w_{\Phi,j}] \log_2(P_j)], \]

(18)

where \( \tilde{w}_{\Phi,1}, \tilde{w}_{\Phi,2} \) (with \( n_w = 2 \)) represent \( \tilde{w}_{\Phi,1} = \{0, -1\} \), respectively.

To test the generality of the neural network, we train the neural network with configurations corresponding the model Hamiltonians with \( h = 0 \), and test it with configurations corresponding Hamiltonians with both non-reciprocal hopping amplitudes (\( \alpha \neq 0 \)) and complex potentials (\( h \neq 0 \)). Training data set includes configurations with \( \alpha \in [0.1, 1.0] \) and the interval \( \Delta \alpha = 0.1 \), each consists of 3.2 × 10³ configurations corresponding Hamiltonians sampled from the two-dimensional parameter space spanned by \( V_1 \in [0, 4] \times V_2 \in [0, 2] \). Test data
set includes 110 pairs of parameters, which consist of $\alpha$ from $\alpha = 0.15$ to $\alpha = 1.95$ with the interval $\Delta \alpha = 0.2$ and $h$ from $h = 0.0$ to $h = 2.0$ with the interval $\Delta h = 0.2$. We sample $3.2 \times 10^3$ configurations corresponding Hamiltonians from the region $V_1 \in [0,4] \times V_2 \in [0,2]$ for each pair of parameters.

After the training, we find that the CNN performs well even without a knowledge of the complex on-site potential ($h = 0$) during the training process. Fig. 5(b) shows the test accuracy table with respect to the two non-Hermiticity parameters $\alpha$ and $h$, with the accuracy more than 99% in the whole parameter region. Moreover, we present the topological phase diagrams predicted by the CNN, which is with respect to $V_1$ and $V_2$ as shown in Fig. 5(c). It is clear that the CNN performs excellently at the deep phase with only a little struggle near the topological phase transitions. We attribute the high accuracy in this learning task to two reasons. First, normalizing data enable both the training and test data distributing in the complex unit, which is important for the generality of the neural network. Second, the topological transitions in this model is consistent with the real-complex transitions in the energy spectrum [76], which reduces the complexity of problem when input data is dependent on complex spectrum.

V. CONCLUSIONS

In summary, we have demonstrated that the artificial neural networks can be used to predict the topological invariants and the associated topological phase transition and topological phase diagrams in three different non-Hermitian models with high accuracy. The winding numbers of the Hatano-Nelson model are presented as a demonstration of our non-Hermitian machine learning method. The CNN trained by the data set within the deep phases has been shown to correctly detect the phase transition near each boundary of the non-Hermitian SSH model. We have investigated the non-Hermitian generalized AAH model with non-reciprocal hopping and complex quasiperiodic potential. It is found that the topological phase diagram in the 2D non-Hermiticity parameter space predicted by CNN has high accuracy with the theoretical one. Our results have shown the generality of machine learning based method on classifying topological properties for both single- and multi-band models.

Note added. After the completion of this work, we noticed a complementary work on machine learning non-Hermitian topological phases [77], which focused on the winding number of the Hamiltonian vectors.

Appendix A: Training details

We first describe some training details for the Hatano-Nelson model. We use the deep learning framework pytorch [78] to construct and train the neural network. Weights are randomly initialized to a normal distribution with Xavier algorithm [79] and the biases are initialized to 0. We use Adam optimizer[80] to minimize the output of the neural network $\tilde{w}$ with true $w$. We
set initial learning rate is 0.001 and use ReduceLROnPlateau algorithm [78] to lower by 10 times when the improvement of the validation loss stops for 20 epochs. All hyper-parameters are set to be default, unless mentioned otherwise. In order to prevent neural overfitting, $L^2$ regularization with strength $10^{-4}$ and early stop [81] are used during the training. We use a mini-batch training with the batch size to be 64 and a validation set to confirm there is no overfitting during training. We use a mini-batch training with the batch size to be 64 and a validation set to confirm there is no overfitting during training. Among $4 \times 10^3$ configurations consist of 1 : 1 : 1 of them having winding numbers $w = \pm \{1, 2, 3\}$, respectively. Typical loss during a training instance of the CNN and FCNN is shown in Fig. 6(b), from which one can see that there is no sign of overfitting.

We then provide some training details for the non-Hermitian SSH model. In this case, the CNN has two convolution layers with 32 kernels of size $1 \times 2 \times 2$ and 1 kernel of size $32 \times 1 \times 1$, followed by a fully connected layer of 16 neurons before output layer. In this model, the output layer consists of three neurons for three different inter-band winding numbers. All the hidden layers have ReLU as activation functions and the output layer has softmax function $f(x)_i = \exp x_i / \sum_{j=1}^{n} \exp x_j$. The exact topological phase diagram in the parameter space spanned by $t$ and $\delta$ is show in Fig. 7(a). Training data set satisfied $s \geq T$ with $T = 0.2$ here and test data set satisfied $s < T$ are randomly sampled from the parameter space. Data set distribution is shown in Fig. 7(b). The number of configurations in training data set, validation data set, and test data set are about $2.4 \times 10^4$, $6 \times 10^3$, and $6 \times 10^3$, respectively. Typical loss during training instances of the CNN for different training data sets is plotted in Fig. 6(b), which clearly shows the neural networks converge quickly without overfitting.

Finally, we present briefly some details for the non-Hermitian generalized AAH model. In this case, the validation set consists of $8 \times 10^3$ configurations corresponding non-reciprocal-hopping Hamiltonians (with $h = 0$) that are not included in the training data set. Typical loss is shown in Fig. 6(c) with the networks converging quickly without overfitting.
ACKNOWLEDGMENTS

We thank Dan-Bo Zhang for helpful discussions. This work was supported by the NSAF (Grants No. U1830111 and No. U1801661), the Key-Area Research and Development Program of Guangdong Province (Grant No. 2019B030330001), and the Science and Technology Program of Guangzhou (Grants No. 201804020055 and No. 2019050001).

[1] M. I. Jordan and T. M. Mitchell, Science 349, 255 (2015).
[2] Y. LeCun, Y. Bengio, and G. Hinton, Nature 521, 436 (2015).
[3] I. J. Goodfellow, Y. Bengio, and A. Courville, Deep Learning (MIT Press, Cambridge, MA, USA, 2016) http://www.deeplearningbook.org.
[4] G. Carleo, I. Cirac, K. Cramer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Rev. Mod. Phys. 91, 045002 (2019).
[5] R. Biswas, L. Blackburn, J. Cao, R. Essick, K. A. Hodge, E. Katsavounidis, K. Kim, Y.-M. Kim, E.-O. Le Bigot, C.-H. Lee, J. J. Oh, S. H. Oh, E. J. Son, Y. Tao, R. Vaulin, and X. Wang, Phys. Rev. D 88, 062003 (2013).
[6] B. S. Rem, N. Kämig, M. Tarnowski, L. Astoria, N. Fläschner, C. Becker, K. Sengstock, and C. Weitenberg, Nat. Phys. 15, 917 (2019).
[7] G. Kasieczka, T. Plehn, A. Butter, K. Cramer, D. Debnath, B. M. Dillon, M. Fairbairn, D. A. Faroughy, W. Fedorko, C. Gay, L. Gouskos, J. F. Kamenik, P. Komiske, S. Leiss, A. Lister, S. Macaluso, E. Metodiev, L. Moore, B. Nachman, K. Nordström, J. Pearkes, H. Qu, Y. Rath, M. Rieger, D. Shih, J. Thompson, and S. Varma, SciPost Physics 7 (2019), 10.21468/scipostphys.7.1.014.
[8] L. Wang, Phys. Rev. B 94, 195105 (2016).
[9] J. Carrasquilla and R. G. Melko, Nat. Phys. 13, 431 (2017).
[10] Y. Zhang and E.-A. Kim, Phys. Rev. Lett. 118, 216401 (2017).
[11] D.-L. Deng, X. Li, and S. Das Sarma, Phys. Rev. B 96, 195145 (2017).
[12] P. Huembeli, A. Dauphin, P. Wittek, and C. Gogolin, Phys. Rev. B 99, 104106 (2019).
[13] X.-Y. Dong, F. Pollmann, and X.-F. Zhang, Phys. Rev. B 99, 121104 (2019).
[14] E. P. Van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Nat. Phys. 13, 435 (2017).
[15] D. Carvalho, N. A. García-Martínez, J. L. Lado, and J. Fernández-Rossier, Phys. Rev. B 97, 115453 (2018).
[16] P. Zhang, H. Shen, and H. Zhai, Phys. Rev. Lett. 120, 060401 (2018).
[17] N. Sun, J. Yi, P. Zhang, H. Shen, and H. Zhai, Phys. Rev. B 98, 085402 (2018).
[18] P. Huembeli, A. Dauphin, and P. Wittek, Phys. Rev. B 97, 134109 (2018).
[19] Y.-H. Tsai, M.-Z. Yu, Y.-H. Hsu, and M.-C. Chung, Phys. Rev. B 102, 054512 (2020).
[20] Y. Ming, C.-T. Lin, S. D. Bartlett, and W.-W. Zhang, npj Comput. Mater. 5, 1 (2019).
[21] J. F. Rodriguez-Nieva and M. S. Scheuer, Nat. Phys. 15, 790 (2019).
[22] N. L. Holanda and M. A. R. Griffith, Phys. Rev. B 102, 054107 (2020).
[23] T. Ohtsuki and T. Mano, J. Phys. Soc. Jpn. 89, 022001 (2020).
[24] Y. Zhang, P. Ginsparg, and E.-A. Kim, Phys. Rev. Research 2, 023282 (2020).
[25] Y. Long, J. Ren, and H. Chen, Phys. Rev. Lett. 124, 185501 (2020).
[26] M. S. Scheurer and R.-J. Slager, Phys. Rev. Lett. 124, 253601 (2020).
[27] B. S. Rem, N. Kaging, M. Tarnowski, L. Astria, N. Fläschner, C. Becker, K. Sengstock, and C. Weitenberg, Nature Physics 15, 917 (2019).
[28] W. Lian, S.-T. Wang, S. Lu, Y. Huang, F. Wang, X. Yuan, W. Zhang, X. Ouyang, X. Wang, X. Huang, L. He, X. Chang, D.-L. Deng, and L. Duan, Phys. Rev. Lett. 122, 210503 (2019).
[29] S. Diehl, A. Micheli, A. Kantian, B. Kraus, H. Bäichler, and P. Zoller, Nat. Phys. 4, 878 (2008).
[30] S. Malzard, C. Poli, and H. Schomerus, Phys. Rev. Lett. 115, 200402 (2015).
[31] T. E. Lee, Phys. Rev. Lett. 116, 133903 (2016).
[32] S. Yao and Z. Wang, Phys. Rev. Lett. 121, 086803 (2018).
[33] S. Yao, F. Song, and Z. Wang, Phys. Rev. Lett. 121, 136802 (2018).
[34] F. Song, S. Yao, and Z. Wang, Phys. Rev. Lett. 123, 246801 (2019).
[35] F. K. Kunst, E. Edvardsson, J. C. Budich, and E. J. Bergholtz, Phys. Rev. Lett. 121, 026808 (2018).
[36] K. Takata and M. Notomi, Phys. Rev. Lett. 121, 213902 (2018).
[37] H. Wang, J. Ruan, and H. Zhang, Phys. Rev. B 99, 075130 (2019).
[38] Q.-B. Zeng, S. Chen, and R. Lü, Phys. Rev. A 95, 062118 (2017).
[39] L.-J. Lang, Y. Wang, H. Wang, and Y. D. Chong, Phys. Rev. B 98, 094307 (2018).
[40] R. Hamasaki, K. Kawabata, and M. Ueda, Phys. Rev. Lett. 123, 090603 (2019).
[41] L. Jin and Z. Song, Phys. Rev. B 99, 081103 (2019).
[42] K. Kawabata, S. Higashikawa, Z. Gong, Y. Ashida, and M. Ueda, Nat. Commun. 10, 1 (2019).
[43] T. Liu, Y.-R. Zhang, Q. Ai, Z. Gong, K. Kawabata, M. Ueda, and F. Nori, Phys. Rev. Lett. 122, 076801 (2019).
[44] C. H. Lee, L. Li, and J. Gong, Phys. Rev. Lett. 123, 016805 (2019).
[45] K. Yamamoto, M. Nakagawa, K. Adachi, K. Takasan, M. Ueda, and N. Kawakami, Phys. Rev. Lett. 123, 123601 (2019).
[46] N. Hatano and D. R. Nelson, Phys. Rev. Lett. 77, 570 (1996).
[47] N. Hatano and D. R. Nelson, Phys. Rev. B 56, 8651 (1997).
[48] Z. Gong, Y. Ashida, K. Kawabata, K. Takasan, S. Higashikawa, and M. Ueda, Phys. Rev. X 8, 031079 (2018).
[49] A. Ghatak and T. Das, J. Phys.: Condens. Matter 31,
D. Leykam, K. Y. Bliokh, C. Huang, Y. D. Chong, and F. Nori, Phys. Rev. Lett. 118, 040401 (2017).
H. Shen, B. Zhen, and L. Fu, Phys. Rev. Lett. 120, 146402 (2018).
G.-Q. Zhang, D.-W. Zhang, Z. Li, Z. D. Wang, and S.-L. Zhu, Phys. Rev. B 102, 054204 (2020).
D.-W. Zhang, L.-Z. Tang, L.-J. Lang, H. Yan, and S.-L. Zhu, Sci. China Phys. Mech. Astron. 63, 1 (2020).
X.-W. Luo and C. Zhang, arXiv:1912.10652v1.
H. Jiang, L.-J. Lang, C. Yang, S.-L. Zhu, and S. Chen, Phys. Rev. B 100, 054301 (2019).
S. Longhi, Phys. Rev. Lett. 122, 237601 (2019).
T. Liu, H. Guo, Y. Pu, and S. Longhi, Phys. Rev. B 102, 024205 (2020).
H. Wu and J.-H. An, Phys. Rev. B 102, 041119 (2020).
Q.-B. Zeng, Y.-B. Yang, and Y. Xu, Phys. Rev. B 101, 020201 (2020).
Q.-B. Zeng and Y. Xu, Phys. Rev. Research 2, 033052 (2020).
L.-Z. Tang, L.-F. Zhang, G.-Q. Zhang, and D.-W. Zhang, Phys. Rev. A 101, 063612 (2020).
H. Liu, Z. Su, Z.-Q. Zhang, and H. Jiang, Chinese Physics B 29, 050502 (2020).
D.-W. Zhang, Y.-L. Chen, G.-Q. Zhang, L.-J. Lang, Z. Li, and S.-L. Zhu, Phys. Rev. B 101, 235150 (2020).
Z. Xu and S. Chen, Phys. Rev. B 102, 035153 (2020).
T. Liu, J. J. He, T. Yoshida, Z.-L. Xiang, and F. Nori, arXiv:2001.09475v2.
W. Xi, Z.-H. Zhang, Z.-C. Gu, and W.-Q. Chen, arXiv:1911.01590v4.
E. Lee, H. Lee, and B.-J. Yang, Phys. Rev. B 101, 121109 (2020).
T. Yoshida, K. Kudo, and Y. Hatsugai, Sci. Rep. 9 (2019), 10.1038/s41598-019-53253-8.
D. S. Borgnia, A. J. Kruchkov, and R.-J. Slager, Phys. Rev. Lett. 124, 056802 (2020).
N. Okuma, K. Kawabata, K. Shiozaki, and M. Sato, Phys. Rev. Lett. 124, 086801 (2020).
K. Zhang, Z. Yang, and C. Fang, arXiv:1910.01131v3.
W. P. Su, J. R. Schrieffer, and A. J. Heeger, Phys. Rev. Lett. 42, 1698 (1979).
P. G. Harper, Proc. Phys. Soc. Sect. A 68, 874 (1955).
S. Aubry and G. Andre, Ann. Israel Phys. Soc. 3, 133 (1980).
F. Liu, S. Ghosh, and Y. D. Chong, Phys. Rev. B 91, 014108 (2015).
L.-Z. Tang and et al, In prepareration.
B. Narayan and A. Narayan, arXiv:2009.06476 (2020).
A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, in Advances in Neural Information Processing Systems 32 (Curran Associates, Inc., 2019) pp. 8026–8037.
X. Glorot and Y. Bengio, in Proceedings of Machine Learning Research, Vol. 9, edited by Y. W. Teh and M. Titterington (JMLR Workshop and Conference Proceedings, Chia Laguna Resort, Sardinia, Italy, 2010) pp. 249–256.
D. Kingma and J. Ba, arXiv:1412.6980 (2014).
Y. Yao, L. Rosasco, and A. Caponnetto, Constructive Approximation 26, 289 (2007).