UC San Diego

UC San Diego Previously Published Works

Title

Shape prediction on the basis of spectrum using neural networks

Permalink

https://escholarship.org/uc/item/57b04648

Journal

Physical Review Research, 5(1)

ISSN

2643-1564

Authors

Zhao, Y
Fogler, MM

Publication Date

2023-02-01

DOI

10.1103/physrevresearch.5.013110

Copyright Information

This work is made available under the terms of a Creative Commons Attribution License, availalbe at https://creativecommons.org/licenses/by/4.0/

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

Shape prediction on the basis of spectrum using neural networks

Y. Zhao and M. M. Fogler
Department of Physics, University of California San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA

(Received 1 June 2022; accepted 20 December 2022; published 14 February 2023)

We have developed a deep neural network that reconstructs the shape of a polygonal domain given the first hundred of its Dirichlet Laplacian eigenvalues. Having an encoder-decoder structure, the network maps input spectra to a latent space and then predicts the discretized image of the domain on a square grid. Tested on randomly generated pentagons, the predictions prove to be highly accurate for both concave and convex pentagons. Our analysis indicates that the network has discovered fundamental properties of the Laplacian operator, the scaling rule, and the continuous rotational symmetry. Additionally, the latent variables are strongly correlated with Weyl's parameters (area, perimeter, and a certain function of the angles) of the test polygons.

DOI: 10.1103/PhysRevResearch.5.013110

I. INTRODUCTION

In 1966, Mark Kac raised the famous question "Can One Hear the Shape of a Drum?" [1]. He wondered if the frequencies of the fundamental vibrational modes, i.e., the Dirichlet eigenvalues for the Helmholtz equation, uniquely define the shape of a domain. This question pertains to the important yet difficult topic of inverse problems in physical sciences. It was not until 1992 when Gordon *et al.* finally gave a negative answer to Kac's question [2] presenting counterexamples such as ones shown in Fig. 1. However, only a few families of such isospectral nonisometric polygon pairs have been found during the subsequent decades of research [3]. A numerical study [4] suggested that the set of all counterexamples has measure zero. In other words, the answer to Kac's question is affirmative for a vast majority of shapes [5–10]. It is then natural to raise a follow-up question of how to reconstruct the domain shape from its spectrum, or, in more practical terms, from a finite number of its first Dirichlet eigenvalues. We call this the inverse Dirichlet problem (IDP).

The IDP is obviously not limited to acoustics but can be generalized to a wide variety of field and matter wave phenomena. Our own interest in IDP originates from the scanning near-field optical microscopy (SNOM) of subdiffractional resonators. In these types of experiments, one can measure both the phonon-polariton mode spectra and the real space images of small samples [11–13]. As their dimensions shrink, it becomes difficult to directly resolve the detailed shape of these nanostructures. However, this geometric information is still encoded in the mode spectra. We envision that tools developed to solve the IDP could be utilized to improve the resolution limits of SNOM and other scanned probe nanoscopy techniques.

Inverse problems are commonly solved by iterative numerical methods. For the IDP, one can employ the same strategy. For example, there exists an algorithm that iteratively morphs the embedding mesh of the domain until its Dirichlet eigenvalues match those of the ground truth (GT) [14,15]. Although this method seems to perform reasonably well for smooth and convex drum shapes, it is computationally intensive and generally requires a good initial guess (see Fig. 9 in the Appendix).

In recent years, deep neural networks (DNN) have gained popularity as a new tool for solving a variety of inverse problems. In physical sciences, examples include optimization of photonic crystals [16], data analysis in microscopy and spectroscopy [17,18], detection of quantum phases [19–21], and subatomic particles in collision experiments [22–24]. Some studies in computer vision used DNN to solve problems conceptually similar to IDP, e.g., reconstruction of three-dimensional (3D) geometries from two-dimensional (2D) images or vector encodings [25–28] and 2D images from text descriptions [29–31]. Recently, the DNN method has been used to solve the IDP for 3D shapes representing human facial expressions [32]. That work was motivated by industrial applications of computer vision techniques. In this work, we developed a DNN to solve the IDP for randomly generated pentagons. Additionally, we tried to analyze the physical and mathematical properties of its latent variables. Interpreting the knowledge gained by a DNN is in fact a trending topic in the machine learning community. For example, in computer vision studies, the DNN latent space has been shown to correlate with facial details and object symmetries [33,34]. In physical sciences, the DNN latent variables have been shown to contain conserved quantities in collisions [35], the heliocentric representation of orbits in astronomy [35], symmetries of quantum field models [36], and order parameters in phase transitions [37,38]. Here we find evidence that the DNN discovered the scaling and the rotational symmetry of the Laplacian operator. We also find a strong correspondence between the DNN latent

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

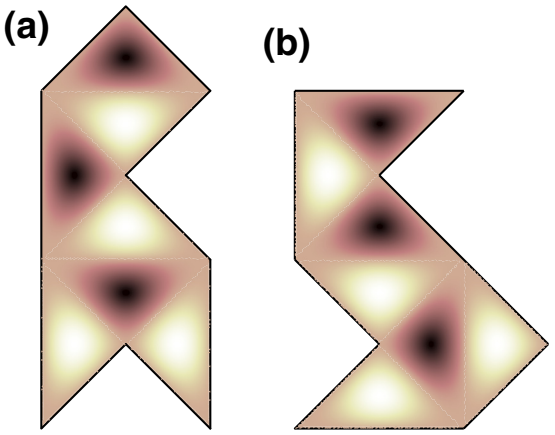

FIG. 1. A nonisometric isospectral pair. The two polygonal drums (a) "the hawk" and (b) "the seal" share the same spectrum of vibrational eigenfrequencies. The false color depicts the spatial distribution of the 9th eigenmodes of each polygon.

variables and the three Weyl parameters of the domain (area $\mathcal{A}$, perimeter $\mathcal{L}$, and a certain function $\mathcal{K}$ of the angles, see below).

The remainder of the paper is organized as follows. In Sec. I, we provide details on the data generation and the DNN design. In Sec. III we discuss the performance of the DNN and interpret the physics learned by it. We end with concluding remarks in Sec. IV.

II. DATA GENERATION AND NETWORK STRUCTURE

We now describe the data generation and the structure of our DNN. The vertices of the pentagons were generated in polar coordinates with polar distances uniformly sampled between 0.5 and 2. To ensure the proper size of the pentagons, the corresponding polar angles were generated such that the included angles between neighboring radii have values between $\pi/10$ and π. The associated first 100 Dirichlet eigenvalues were calculated using a standard finite-element solver (Mathematica) [39]. It is important to notice that the computed eigenvalue spectrum is invariant under any translations, rotations, and reflections of the pentagon. However, in IDP these isometric transformations introduce ambiguity in the DNN output. To eliminate this "gauge freedom," we did the following. First, we shifted the centroids of each pentagon to the origin. Second, we rotated the pentagons such that the vertex with the smallest inner angle is located on the $x > 0$ semiaxis. Finally, the remaining reflection symmetry was integrated into the loss function to be discussed later in this section. After the above procedure, the pentagons were coarse-grained into bitmap images $I_{\rm GT}$ on a 41×41 grid obtained by dividing the square $-2 \leqslant x, y \leqslant 2$ into pixels of size 0.1×0.1. For these input images we restricted the pixel values to be $p_i = 0$ or 1. On the other hand, the output of the DNN are images I that we allowed to have fractional pixel values $0 \leqslant q_i \leqslant 1$.

To quantify the similarity between bitmap images, a commonly used metric is the Jaccard index, which is defined as the ratio of the areas of the intersection and the union of two

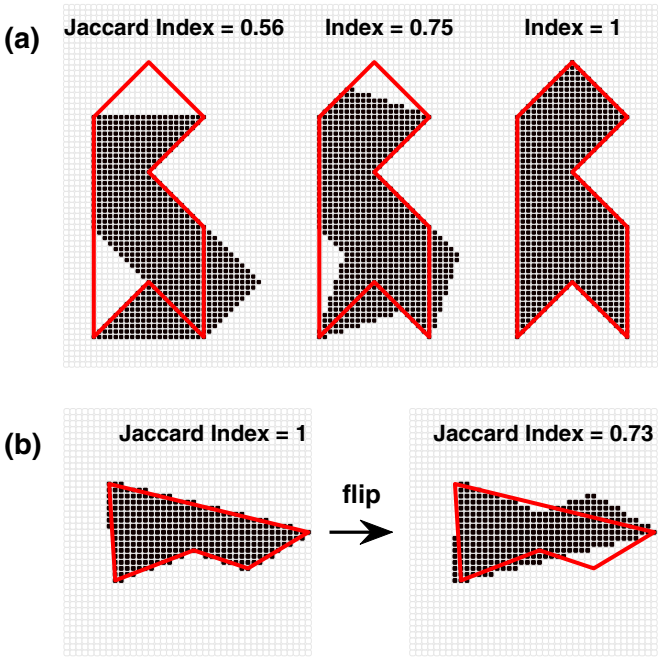

FIG. 2. Illustrations of the Jaccard index properties. (a) The Jaccard index for the hawk gradually morphing into the GT, the seal. (b) Reflection of an image with respect to a horizontal axis does not change its shape yet diminishes its Jaccard index. The GT is the red pentagon.

sets. As illustrated in Fig. 2(a), the Jaccard index increases as the "hawk" gradually morphs into the "seal." For grayscale images I in our study, a power-2 variant of the Jaccard index [40,41] is more suitable. This latter metric is defined by

$$J(I, I_{\rm GT}) = \frac{\sum_i p_i q_i}{\sum_i p_i^2 + q_i^2 - p_i q_i}. \tag{1}$$

It coincides with the Jaccard index if the pixel values p_i, q_i are restricted to $\{0, 1\}$ and ensures that $0 \leqslant J \leqslant 1$ if these values are allowed to be fractional. The maximum possible value of $J = 1$ is attained when I and $I_{\rm GT}$ are identical, $p_i = q_i$. The summation index i in Eq. (1) runs over the entire 41×41 image. However, the previously mentioned gauge freedom still compromises this metric. As demonstrated in Fig. 2(b), after a reflection, a perfect prediction deviates from the GT (the red pentagon) and therefore has the Jaccard index $J < 1$. To combat this problem we subject I to all possible symmetry operations π of the square grid (comprising the dihedral group D_4) and pick the one with the largest J. The corresponding loss function L is given by

$$L = \min_{\pi \in D_4} [1 - J(\pi(I), I_{\rm GT})]. \tag{2}$$

Our DNN has the encoder-decoder structure. Networks with such a structure have been shown to handle many complicated tasks including detecting abnormal quantum phases [42], approximating quantum state distributions [43], and parametrizing nonlinear dynamics [44–46]. Recent studies have also shown that encoder-decoder DNNs have the ability to discover key physical parameters of the problem [35,47]. As depicted in Fig. 1(a), the input to the DNN consists of

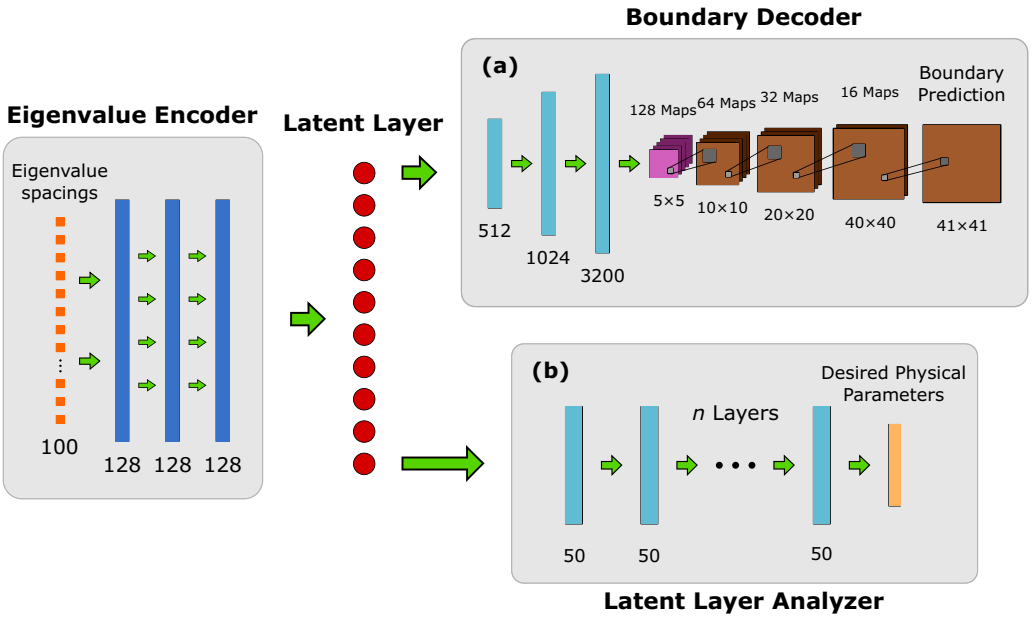

FIG. 3. Structure of the DNN developed in this study. The full architecture contains an eigenvalue encoder, a latent layer, a boundary decoder, and a latent layer analyzer. The encoder consists of three consecutive LSTM layers (dark blue) that extract information from the input eigenvalue spacings. This information is then compressed into the latent layer of 10 neurons. The decoder first expands the latent space via three dense layers (cyan), then reshapes (magenta) the result, then decodes the image using four 2D transpose convolutional layers (brown). The latent layer analyzer deciphers the latent space using n consecutive hidden dense layers. (a) The encoder and the latent layer are connected to the decoder for boundary prediction task. Here we train all the sections. (b) The latent layer analyzer is connected to the encoder and the latent layer for physical parameter extraction. In training, only the weights in the analyzer are adjusted.

the first 100 eigenvalue spacings. The encoder portion is composed of three consecutive long-short term memory (LSTM) [48] layers of size 128 each. The latent layer directly after the encoder contains 10 neurons with linear activation. Even though seven independent parameters are enough to uniquely determine a pentagon, this slightly larger latent space helps to increase the DNN approximation capability. The decoder starts with four dense layers of sizes 50, 512, 1024, and 3200 with LeakyReLu activation (negative slope coefficient set to 0.3 and is the same for the rest of this section) [49]. Next, the output vector of size 3,200 is reshaped to 128 feature maps each of size 5×5. Four 2D Transpose Convolution (TransConv2D) layers with kernel sizes 3×3, 3×3, 3×3, and 1×1 and kernel stride lengths of 2, 2, 2, and 1, then construct the image up-sampling block. Through such a process, the number of feature maps reduces to 64, 32, 16, and 1, and the initial 5×5 image expands to the final prediction of size 41×41. The first three TransConv2D layers have the ReLu activation while the last one has sigmoid activation to constraint the predicted pixel values. Only the last TransConv2D layer utilizes zeropadding. During training, the loss function defined in Eq. (2) is applied and 20% of the samples are used for validation. A total of 1,100,000 training samples are fed into the network in 11 batches. The training epoch number for each batch decreases linearly from 50 to 10. We used the Adam optimization method with an initial learning rate of 10^{-3} and a decay rate of 10^{-5}. After training, a test data set containing 100,000 additional samples was used to measure the prediction accuracy. Our source codes are publicly available at Github [50].

III. RESULTS

The mean of the loss function L for the entire test dataset occurs at 0.069 with over 67% of the tested cases achieving a loss below this value. To better illustrate the relation between the loss function and the goodness of the DNN fit, we provide six examples in Fig. 4. These examples are arranged with increasing loss values, from Figs. 4(a) to 4(f) representing the lowest 10%, 25%, 40%, 57%, 75%, and 95% of all loss values, respectively. Their corresponding positions in the cumulative distribution function (CDF) of all loss values are also highlighted in the bottom subfigure in Fig. 4. Based on visual inspection, we consider the predictions in Figs. 4(a)–4(c) "good", Figs. 4(d) and 4(e) "average", and Fig. 4(f) "bad." The "good" predictions capture almost every detail of the true polygonal shapes (shown in red), with a sharp boundary and only a few pixel-level discrepancies present. In the "average" predictions, the pentagon edges become somewhat blurred. In the "bad" predictions, however, major discrepancies appear at multiple edges and vertices. We think that these failures are characteristic of shapes that were relatively rare instances in the training set. Overall, since our DNN captures the true geometry for the majority of the tested samples, we conclude that it can indeed "hear" the shape of a pentagon from its overtones.

Next, we address the question whether the DNN is capable of learning basic properties of the Dirichlet problem. We attempted to answer this question by investigating the scaling behavior of the DNN output. The eigenvalues scale inversely proportional to the square of the linear dimension of the drum.

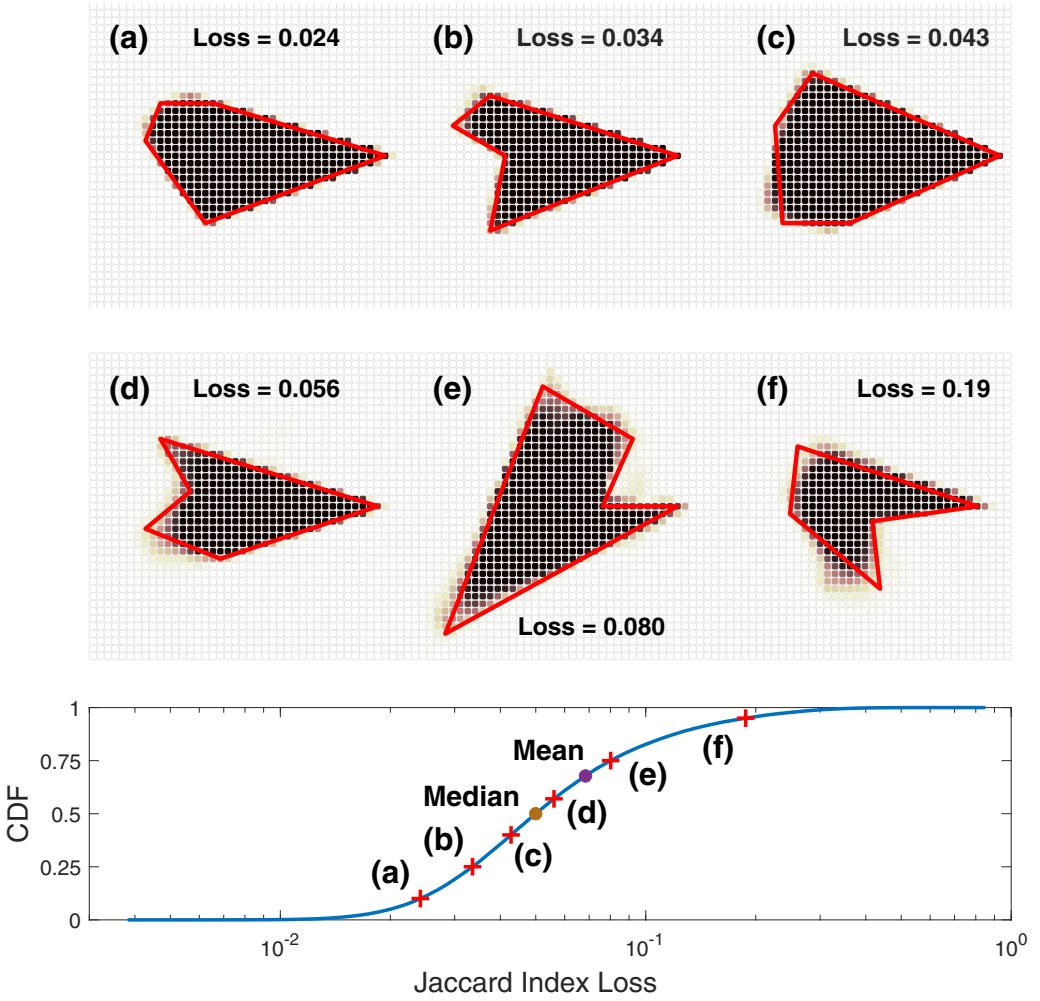

FIG. 4. Typical DNN predictions. The examples (a)–(f) are ordered according to their loss values. The bottom plot shows the cumulative distribution function (CDF) of the loss function for the entire test dataset. The red crosses mark the positions corresponding to the predictions (a)–(f). The brown and the purple dots indicate the statistical median and mean, respectively.

Therefore, scaling of the DNN input (level spacings) by a factor of S should make the predicted area $\mathcal{A}$ expand by a factor of $1/S$. To check if the DNN adheres to this rule, we repeated the simulations applying scaling factors $0.5 \leqslant S \leqslant 2.5$ to the previous dataset. The results are demonstrated in Fig. 5. In Fig. 5(a) we show the scaling behavior of a typical good prediction. In this example, the predicted polygons vary in size according to the expected scaling rule. The shape of the drum is maintained in both enlarged and shrunken images. Surprisingly, the same scaling rule is also exhibited by a bad prediction. As demonstrated in Fig. 5(b), the DNN changes the size of this faulty prediction while preserving its shape. To check the area scaling quantitatively, we defined the area $\mathcal{A}$ of the predicted pentagon to be the sum of all the pixel values. We observed that it has a power-law dependence on the eigenvalue scaling factor S with the exponent of -0.93, which is very close to the expected -1, see Fig. 5(c).

Another surprise is that the DNN also discovers the continuous rotational degree of freedom beyond the symmetry of the underlying square grid. As exemplified in Fig. 6(a), after a counterclockwise rotation of 150°, a seemingly fallacious prediction with $L = 0.35$ shows a much better agreement with the GT ($L = 0.13$). Such cases make up more than one quarter of the worst 3,000 predictions. Additional examples of such seemingly "bad" predictions include instances where L can be greatly reduced if a suitable rotation is combined with a reflection, see Fig. 6(b). It is remarkable that these predictions exist despite being penalized by the loss function. They suggest that the DNN must have captured fundamental information associated with the geometry of the drumheads.

Frequently discussed in the context of the IDP is Weyl's formula [3,51,52] for the average number of eigenvalues below a prescribed value E:

$$\mathcal{N}(E) \simeq \frac{\mathcal{A}}{4\pi}E - \frac{\mathcal{L}}{4\pi}\sqrt{E} + \mathcal{K}. \tag{3}$$

Here $\mathcal{L}$ is the perimeter of the polygon, constant $\mathcal{K}$ is given by

$$\mathcal{K} = \frac{1}{24}\sum_i \left(\frac{\pi}{\alpha_i} - \frac{\alpha_i}{\pi}\right), \tag{4}$$

and $0 < \alpha_i < 2\pi$ are the inner angles. We find evidence that the DNN discovers Weyl's expansion and stores information about the three parameters $\mathcal{A}$, $\mathcal{L}$, and $\mathcal{K}$ in the latent neurons. To do so we utilize the universal approximation property

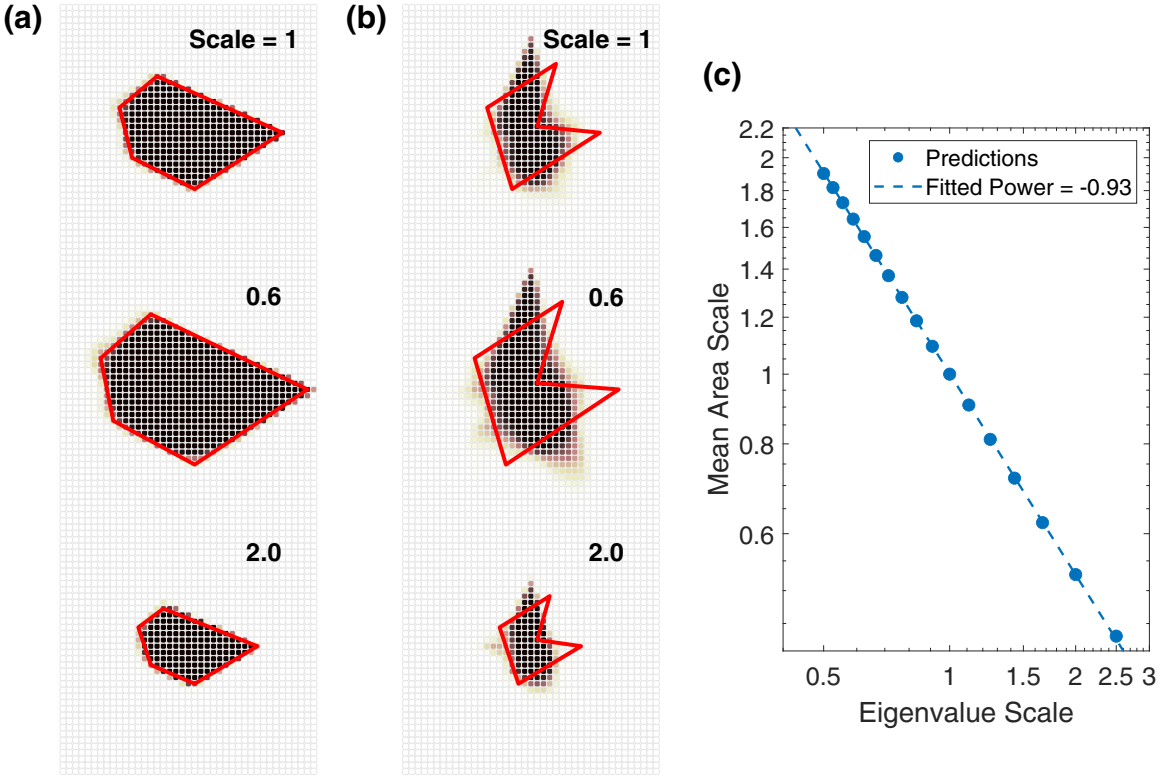

FIG. 5. The DNN discovers the scaling property of the Laplacian. (a) Examples of the scaling behavior of the predicted images with the corresponding eigenvalue scaling factors indicated. The GTs are shown in red for comparison. (b) The scaling law holds even for a poor prediction. (c) The predicted area as a function of the scaling factor, fitted to a powerlaw (dashed line) with exponent −0.93. The statistical error is smaller than the symbol size.

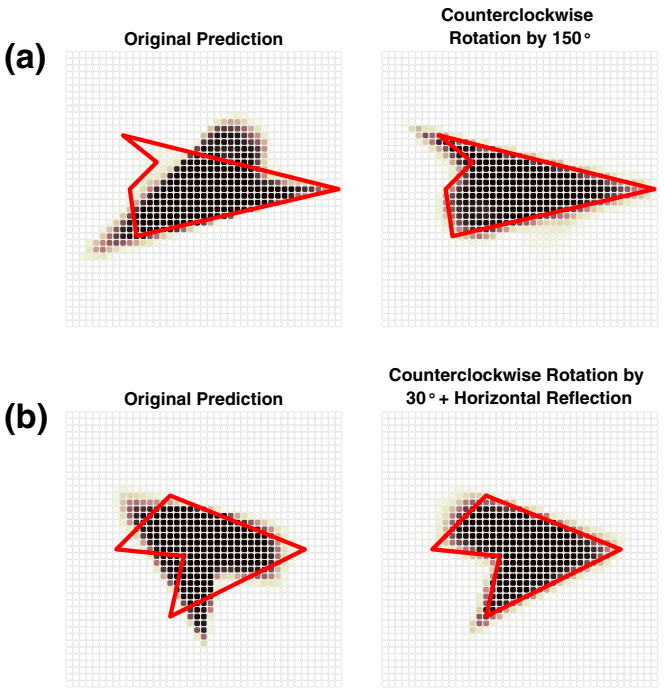

FIG. 6. The DNN discovers spatial symmetries beyond the limitations of the underlying square grid. (a) The rotational symmetry (b) a rotation combined with a reflection. The GTs are shown in red.

[53] of neural networks to establish a mapping from the latent neurons to Weyl's parameters. Specifically, we detached the encoder and the latent layer from the original DNN and connected it instead to a small latent layer analyzer network whose outputs were Weyl's parameters (or other parameters of interest, see below). To control the complexity of the mapping function, we varied the number of hidden layers in the small network but fixed the size of each layer to 50 neurons, see Fig. 3(b). We kept the output layer activationfree and hence, the only source of nonlinearity in the small network was the hidden layers where the LeakyReLu activation function was applied. During training, we allowed modifications of the small network parameters only, using the mean squared error (MSE) as the loss function. In a similar way, we constructed two other networks, one predicting Cartesian coordinates of the polygon vertices and another one predicting the edge lengths and inner angles of the polygons. These more traditional ways to define shapes were included for comparison.

We used 1,100,000 samples to train the mapping from the latent space to each of these three sets of boundary descriptors. The same optimizer with identical learning rate and decay policy was applied. The testing results from 100,000 samples are demonstrated in Fig. 7(a), where the mean predicted percentage errors for every parameter set are graphed as functions of the number of the hidden layers. For Weyl's parameters, the error is as low as 7% for a simple linear mapping. When nonlinearity is later introduced, the error plateaus at 1.6%. In

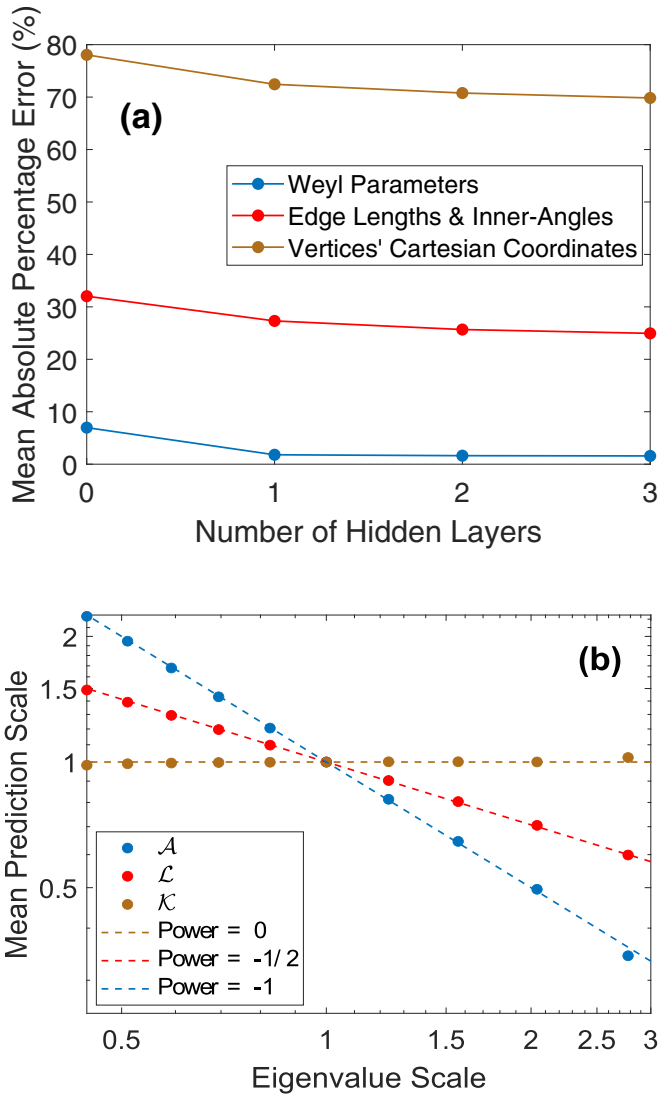

FIG. 7. Latent space analysis. (a) The prediction accuracy for different sets of geometric parameters (Weyl's parameters, blue; the edge lengths and inner angles, red; the Cartesian coordinates of the vertices, brown) as a function of the number of hidden layers in the analyzer. (b) Scaling behavior of the Weyl's parameters (area, red; perimeter, blue; inner angle characteristic, brown) predicted by the analyzer with a single hidden layer. The dashed lines represent the proper scaling laws. The test set contained 10,074 samples that remained within the grid boundary in the full range of the eigenvalue scaling factors shown.

contrast, even equipped with the highest level of nonlinearity, the network struggles to construct the mapping functions for the remaining two sets of descriptors. The prediction accuracy hardly surpass those of random guess. We conclude that out of these three sets, the latent representation has the closest relation to Weyl's parameters.

At this point it still remains a question whether the combined network understands the meaning of $\mathcal{A}$, $\mathcal{L}$, and $\mathcal{K}$. Since no additional physical information, e.g., measurement units, was provided during training, it is possible that the learned mapping treated the Weyl's parameters as unitless and hence overfitted them. To check if that is the case we again

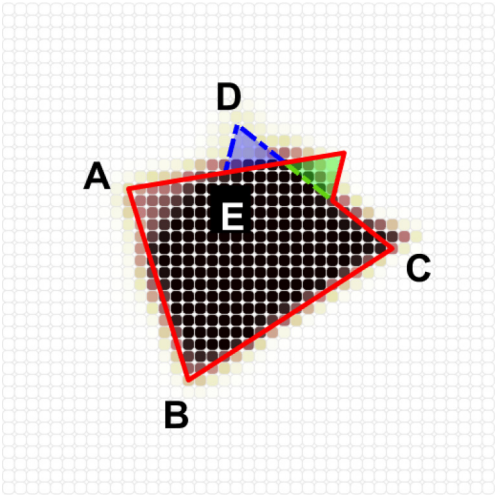

FIG. 8. Weyl's parameters preservation in an "average" prediction. The prediction is approximated by a polygon ABCDE which has the same Weyl's parameters as the red polygon (the GT rotated by a small angle). The blue and green triangles are congruent.

investigated the scaling behavior of the outputs. Unlike the DNN predictions of the images, the output Weyl's parameters should obey different scaling laws: $\mathcal{A}$ should scale as $1/S$, $\mathcal{L}$ as $1/\sqrt{S}$, while $\mathcal{K}$ should be scale-invariant. As demonstrated in Fig. 7(b), for a network with one hidden layer, the average $\mathcal{A}$, $\mathcal{L}$, and $\mathcal{K}$ predictions indeed follow the correct scaling laws. In this figure we included only the samples and scaling factors S for which the scaled shapes remain within the training range (for the entire data set, see Fig. 10 in the Appendix). From these results, we infer that Weyl's parameters are stored in the latent space quasilinearly, and so the prediction of the drum shapes by the encoder-decoder network may be aided by these parameters. Yet another evidence in favor of this interpretation is obtained by visually examining the "average" predictions. As demonstrated in Fig. 8, one such prediction misses the upper right corner of the pentagon, as outlined by the green triangle but includes an extra triangular region depicted in blue. These two regions, the missing one and the added one, are approximately isometric, and so the predicted image preserves Weyl's parameters of the GT. The results from the latent layer analyzer network further validate the above analysis: the predicted $\mathcal{A}$, $\mathcal{L}$, and $\mathcal{K}$ for this example are only 0.64%, 1.2%, and 3.2% off the GT. This may be why the DNN has a difficulty handling this particular case.

IV. CONCLUSIONS AND OUTLOOK

In this paper, we have discussed an encoder-decoder DNN that solves the IDP for simple shapes such as pentagons with an exceptional accuracy. We presented evidence that this DNN has learned the scaling properties of the Laplacian operator and that it stores information about Weyl's parameters (area $\mathcal{A}$, perimeter $\mathcal{L}$, and inner-angle characteristic $\mathcal{K}$) in the latent representation. The DNN has discovered a continuous rotational symmetry of the Dirichlet problem beyond the limitations of the square grid. Note that the latent space must also contain information about other boundary descriptors, such as the Cartesian coordinates of the vertices. However, we

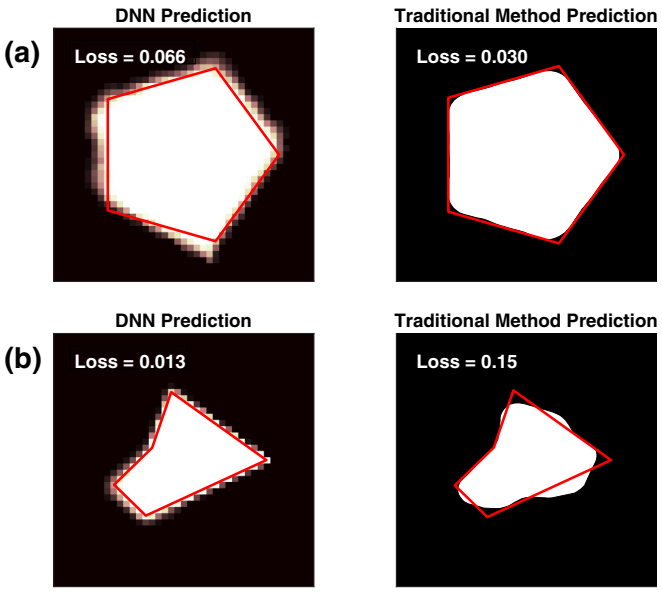

FIG. 9. Reconstructions from the DNN and the traditional method [15] for (a) a regular pentagon and (b) a concave pentagon.

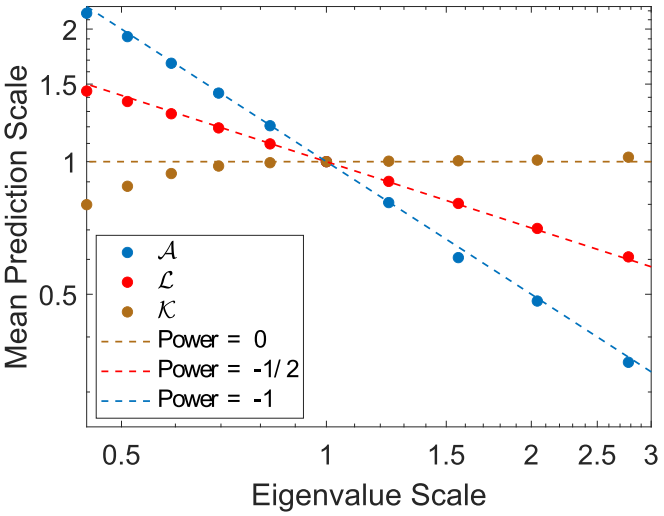

FIG. 10. Similar to Fig. 7 of the main text but for the entire 100,000 samples data set.

found extracting these more readily interpretable parameters challenging. We think that one could potentially achieve that by implementing a mutually independent latent space [35] and more extensive training. Our DNN should in principle be able to handle more complicated nonpolygonal shapes if the size of the output layer, i.e., the resolution of the pixel grid, is increased further.

To decipher the entangled latent space of our DNN we utilized an auxiliary neural network. By changing the number of hidden layers and the type of activation functions, we varied the complexity of the approximation function and established the correspondence between the latent neurons and Weyl's parameters. Without hidden layers, our method performs similar to the principal component analysis. However, with additional hidden layers, our method should be more powerful as it can model nonlinear relations.

As mentioned in the introduction, there are examples of eigenvalue spectra [2] for which the solution of the IDP is not unique (Fig. 1). These examples are rare (of measure zero) but very interesting from the theoretical point of view [3]. We envision that an improved DNN architecture could discover new families of such nonisometric isospectral domain pairs, which may further guide human intuition in solving mathematical problems [54].

Our study was originally motivated by scanning near-field optical microscopy experiments that measured collective mode spectra of subdiffractional nanostructures [13]. Methods we developed to tackle IDP could potentially assist with inferring shapes of such nanostructures when their size becomes smaller than the real-space resolution limit. Implementation of such a super-resolution imaging could be an interesting subject for a future work. We hope that our study would stimulate broad applications of DNNs to theoretical and practical aspects of solving inverse spectral problems in science and technology.

ACKNOWLEDGMENT

We thank Yi-Zhuang You for valuable suggestions on improving the network design and Mengkun Liu and Xinzhong Chen for useful comments and discussions.

APPENDIX

In Fig. 9 we compare the performance of the DNN and the traditional method in [15] when predicting two pentagons. For both examples, the initial guess for the traditional method is an oval. In Fig. 9(a), the traditional method provides a slightly more accurate reconstruction than the DNN. However, the computation time exceeds two mins while that for the DNN is within a fraction of a second. In Fig. 9(b), the traditional method does not converge while the DNN gives an excellent prediction. Figure 10 shows the scaling behavior of Weyl's parameters for the entire data set. The small discrepancy in $\mathcal{K}$ at low S is caused by the samples which exceed the 41×41 image range when boosted. Removing such samples restores the ideal scaling behavior, see Fig. 7.

[1] M. Kac, Can one hear the shape of a drum? Am. Math. Mon. **73**, 1 (1966).
[2] C. Gordon, D. L. Webb, and S. Wolpert, One cannot hear the shape of a drum, Bull. Am. Math. Soc. **27**, 134 (1992).
[3] O. Giraud and K. Thas, Hearing shapes of drums: Mathematical and physical aspects of isospectrality, Rev. Mod. Phys. **82**, 2213 (2010).
[4] M. Panine and A. Kempf, Towards spectral geometric methods for Euclidean quantum gravity, Phys. Rev. D **93**, 084033 (2016).

[5] S. Zelditch, Spectral determination of analytic bi-axisymmetric plane domains, Math. Res. Lett. **6**, 457 (1999).

[6] S. Zelditch, Spectral determination of analytic bi-axisymmetric plane domains, Geometric & Functional Analysis GAFA **10**, 628 (2000).

[7] A. Iantchenko, J. Sjoestrand, and M. Zworski, Birkhoff normal forms in semi-classical inverse problems, Math. Res. Lett. **9**, 337 (2002).

[8] S. Zelditch, Inverse spectral problem for analytic domains I: Balian-bloch trace formula, Commun. Math. Phys. **248**, 357 (2004).

[9] S. Zelditch, Inverse resonance problem for z_2-symmetric analytic obstacles in the plane, in *Geometric Methods in Inverse Problems and PDE Control*, edited by C. B. Croke, M. S. Vogelius, G. Uhlmann, and I. Lasiecka, The IMA Volumes in Mathematics and its Applications (Springer, New York, 2004), pp. 289–321.

[10] S. Zelditch, Inverse spectral problem for analytic domains, II: z_2-Symmetric domains, Ann. Math. **170**, 205 (2009).

[11] J. D. Caldwell, A. V. Kretinin, Y. Chen, V. Giannini, M. M. Fogler, Y. Francescato, C. T. Ellis, J. G. Tischler, C. R. Woods, A. J. Giles, M. Hong, K. Watanabe, T. Taniguchi, S. A. Maier, and K. S. Novoselov, Sub-diffractional volume-confined polaritons in the natural hyperbolic material hexagonal boron nitride, Nat. Commun. **5**, 5221 (2014).

[12] Z. Sun, Á. Gutiérrez-Rubio, D. N. Basov, and M. M. Fogler, Hamiltonian optics of hyperbolic polaritons in nanogranules, Nano Lett. **15**, 4455 (2015).

[13] L. V. Brown, M. Davanco, Z. Sun, A. Kretinin, Y. Chen, J. R. Matson, I. Vurgaftman, N. Sharac, A. J. Giles, M. M. Fogler, T. Taniguchi, K. Watanabe, K. S. Novoselov, S. A. Maier, A. Centrone, and J. D. Caldwell, Nanoscale mapping and spectroscopy of nonradiative hyperbolic modes in hexagonal boron nitride nanostructures, Nano Lett. **18**, 1628 (2018).

[14] D. Aasen, T. Bhamre, and A. Kempf, Shape from Sound: Toward New Tools for Quantum Gravity, Phys. Rev. Lett. **110**, 121301 (2013).

[15] L. Cosmo, M. Panine, A. Rampini, M. Ovsjanikov, M. M. Bronstein, and R. Rodola, Isospectralization, or how to hear shape, style, and correspondence, in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Long Beach, CA, 2019), pp. 7521–7530.

[16] T. Christensen, C. Loh, S. Picek, D. Jakobović, L. Jing, S. Fisher, V. Ceperic, J. D. Joannopoulos, and M. Soljaćić, Predictive and generative machine learning models for photonic crystals, Nanophotonics **9**, 4183 (2020).

[17] N. Borodinov, S. Neumayer, S. V. Kalinin, O. S. Ovchinnikova, R. K. Vasudevan, and S. Jesse, Deep neural networks for understanding noisy data applied to physical property extraction in scanning probe microscopy, npj Comput. Mater. **5**, 25 (2019).

[18] X. Chen, R. Ren, and M. Liu, Validity of Machine Learning in the Quantitative Analysis of Complex Scanning Near-Field Optical Microscopy Signals Using Simulated Data, Phys. Rev. Appl. **15**, 014001 (2021).

[19] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, Nat. Phys. **13**, 435 (2017).

[20] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, Nat. Phys. **13**, 431 (2017).

[21] X.-Y. Dong, F. Pollmann, and X.-F. Zhang, Machine learning of quantum phase transitions, Phys. Rev. B **99**, 121104(R) (2019).

[22] P. Baldi, P. Sadowski, and D. Whiteson, Searching for exotic particles in high-energy physics with deep learning, Nat. Commun. **5**, 4308 (2014).

[23] P. Baldi, P. Sadowski, and D. Whiteson, Enhanced Higgs Boson to $\tau^+\tau^-$ Search with Deep Learning, Phys. Rev. Lett. **114**, 111801 (2015).

[24] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, and D. Whiteson, Jet flavor classification in high-energy physics with deep neural networks, Phys. Rev. D **94**, 112002 (2016).

[25] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling, in *Advances in Neural Information Processing Systems* (Curran Associates, 2016), Vol. 29.

[26] A. Sinha, A. Unmesh, Q. Huang, and K. Ramani, SurfNet: Generating 3D shape surfaces using deep residual networks, in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Honolulu, HI, 2017), pp. 791–800.

[27] T. Aumentado-Armstrong, S. Tsogkas, A. Jepson, and S. Dickinson, Geometric disentanglement for generative latent shape models, in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (IEEE, Seoul, Korea (South), 2019), pp. 8180–8189.

[28] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, Expressive body capture: 3D hands, face, and body from a single image, in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Long Beach, CA, 2019), pp. 10967–10977.

[29] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, Generative adversarial text to image synthesis, in *Proceedings of the International Conference on Machine Learning* (PMLR, New York, 2016), pp. 1060–1069.

[30] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, StackGAN++: Realistic image synthesis with stacked generative adversarial networks, IEEE Trans. Pattern Anal. Mach. Intell. **41**, 1947 (2019).

[31] A. El-Nouby, S. Sharma, H. Schulz, R. D. Hjelm, L. E. Asri, S. E. Kahou, Y. Bengio, and G. Taylor, Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction, in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (IEEE, Seoul, Korea (South), 2019), pp. 10303–10311.

[32] R. Marin, A. Rampini, U. Castellani, E. Rodolà, M. Ovsjanikov, and S. Melzi, Spectral shape recovery and analysis via data-driven connections, Int. J. Comput. Vis. **129**, 2745 (2021).

[33] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, Beta-VAE: Learning basic visual concepts with a constrained variational framework, in *Proceedings of the 5th International Conference on Learning Representations, ICLR 2017* (Toulon, France, 2017).

[34] H.-Y. Hu, D. Wu, Y.-Z. You, B. Olshausen, and Y. Chen, RG-Flow: A hierarchical and explainable flow model based on renormalization group and sparse prior, Mach. Learn.: Sci. Technol. **3**, 035009 (2022).

[35] R. Iten, T. Metger, H. Wilming, L. del Rio, and R. Renner, Discovering Physical Concepts with Neural Networks, Phys. Rev. Lett. **124**, 010508 (2020).

[36] H.-Y. Hu, S.-H. Li, L. Wang, and Y.-Z. You, Machine learning holographic mapping by neural network renormalization group, Phys. Rev. Res. **2**, 023369 (2020).

[37] S. J. Wetzel, Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders, Phys. Rev. E **96**, 022140 (2017).

[38] S. V. Kalinin, O. Dyck, S. Jesse, and M. Ziatdinov, Exploring order parameters and dynamic processes in disordered systems via variational autoencoders, Sci. Adv. **7**, eabd5084 (2021).

[39] Wolfram Research, Inc., Mathematica, Version 13.2, Champaign, IL (2022).

[40] S.-H. Cha, Comprehensive survey on distance/similarity measures between probability density functions, Int. J. Math. Model. Meth. Appl. Sci. **1**, 300 (2007).

[41] D. Duque-Arias, S. Velasco-Forero, J. Deschaud, F. Goulette, A. Serna, E. Decenciàre, and B. Marcotegui, On power Jaccard losses for semantic segmentation, in *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications Volume 5: VISAPP* (Setubal, Portugal, 2021), pp. 561–568.

[42] K. Kottmann, P. Huembeli, M. Lewenstein, and A. Acín, Unsupervised Phase Discovery with Deep Anomaly Detection, Phys. Rev. Lett. **125**, 170603 (2020).

[43] A. Rocchetto, E. Grant, S. Strelchuk, G. Carleo, and S. Severini, Learning hard quantum distributions with variational autoencoders, npj Quantum Inf. **4**, 1 (2018).

[44] B. Lusch, J. N. Kutz, and S. L. Brunton, Deep learning for universal linear embeddings of nonlinear dynamics, Nat. Commun. **9**, 4950 (2018).

[45] N. Takeishi, Y. Kawahara, and T. Yairi, Learning koopman invariant subspaces for dynamic mode decomposition, in *Advances in Neural Information Processing Systems* (Curran Associates, 2017), Vol. 30.

[46] S. E. Otto and C. W. Rowley, Linearly recurrent autoencoder networks for learning dynamics, SIAM J. Appl. Dyn. Syst. **18**, 558 (2019).

[47] D. Zheng, V. Luo, J. Wu, and J. B. Tenenbaum, Unsupervised learning of latent physical properties using perception-prediction networks, arXiv:1807.09244.

[48] S. Hochreiter and J. Schmidhuber, Long short-term memory, Neural Comput. **9**, 1735 (1997).

[49] A. L. Maas, A. Y. Hannun, and A. Y. Ng, Rectifier nonlinearities improve neural network acoustic models, in *Proceedings of the 30th International Conference on Machine Learning*, JMLR: W&CP, Vol. 28 (Atlanta, Georgia, 2013).

[50] https://github.com/diffractiongoo/hearing-shape-of-a-drum.

[51] H. Weyl, Ueber die asymptotische Verteilung der Eigenwerte, Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse **1911**, 110 (1911).

[52] K. Stewartson and R. T. Waechter, On hearing the shape of a drum: further results, Math. Proc. Camb. Philos. Soc. **69**, 353 (1971).

[53] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, Neural Networks **2**, 359 (1989).

[54] A. Davies, P. Veličković, L. Buesing, S. Blackwell, D. Zheng, N. Tomašev, R. Tanburn, P. Battaglia, C. Blundell, A. Juhász, M. Lackenby, G. Williamson, D. Hassabis, and P. Kohli, Advancing mathematics by guiding human intuition with AI, Nature (London) **600**, 70 (2021).