A Generative Deep Learning Approach for Shape Recognition of Arbitrary Objects from Phaseless Acoustic Scattering Data

Waqas W. Ahmed, Mohamed Farhat, Pai-Yen Chen, Xiangliang Zhang,* and Ying Wu*

1. Introduction

Wave propagation is widely used to solve the inverse scattering problem (e.g., retrieving the shape of arbitrary objects from scattering data) in a variety of real-life engineering applications, including remote sensing,[2] medical imaging,[3] tomography,[4] underwater robotics,[5] geophysical prospecting,[6] and detection of defects in oil and gas industry.[7] In the past decade, underwater activities such as biological research, target recognition, exploration of seabed resources, and monitoring of the underwater environment have increased the demand for subsea inspections. Compared to other waves, such as electromagnetic waves and visible light, sound waves have their own advantage in detecting targets in water in the long range because of their better transmissive characteristics. Several sonar devices have been developed to detect submarine objects using sound waves.[8] Sonar uses transmitter and sensor elements to send and receive acoustic waves. The waves are reflected by objects on the seabed and detected by the sensors; these measurements over time can be used to reconstruct sonar images with delay sum method to acquire underwater information. Despite their success in detecting certain underwater objects, this method, however, does not adapt well to the complex and constantly changing environment. In this way, the recognition of underwater targets is still primarily dependent on the decision of trained sonar operators,[9] which can be highly imprecise due to the need of continuous manual monitoring. As a result, an automatic and reliable recognition approach is desired to take over human tasks. In addition, scattering information is complex-valued radiated data, whose intensity and phase may not always be accurately measured. Especially it is difficult to retrieve the phase information from the scattered field data, and only the...
intensity information is generally available. Several analytical and numerical methods have been proposed to reconstruct the scatterer’s shape from the scattering data, such as regularization,[10] factorization,[31] linear sampling method,[12] and many others,[11–13] to name a few. Yet, these computation-driven and brute force optimization methods are time-consuming and require tedious hit-and-trial efforts in order to achieve the desired performance.

Recent advances in machine learning (ML) have made deep learning approaches an efficient way to solve forward and inverse scattering problems.[16,17] Deep learning networks can approximate the true solution to on-demand inverse designs, attributed to their ability to learn nonlinear mappings in datasets. Several studies focused on discriminative and generative design methods have shown excellent performance beyond the human capability.[18–20] Meng et al. presented a linear sampling method (LSM) with neural networks to reconstruct the shape of obstacles with acoustic far-field data.[21] LSM relies on selecting a contour line to obtain the shape information of an object. This solution is still somewhat imprecise and requires further improvement. Fan et al. used convolutional neural networks (CNNs) to determine the forward scattering properties for 2D convex prism geometries but did not address the inverse problem.[22] Using neural networks, a compressing sampling matching pursuit (CoSaMP) method is developed to reconstruct the shape of arbitrary objects, but it only works for scatterers with low contrast relative to the surrounding environment.[23] In addition, probability density-based deep learning approaches have also been developed to solve the inverse problem of the acoustic metasurfaces[24] and acoustic cloaks.[25,26] In both forward and inverse designs, the proposed solutions work well for simple geometries, but their performance decreases as the degree of freedom in the design space increases, making the scaling of complex models difficult. In this scenario, the generative models are used to reduce the dimensionality of the design space and efficiently learn the relations between design parameters and system responses.[19,27] For an arbitrary configuration, the design space is significantly enlarged, and the inverse design process becomes extremely challenging due to one-to-many structure–property mapping, allowing diverse predictions. Zhang et al. presented an inverse design method for random metasurfaces, but the method is unable to entirely eliminate nonunique space because patterns and electromagnetic responses do not have one-to-one correspondence.[28] Lai et al. studied the inverse scattering problem that determines the configuration of 2D rigid cylinders for given total scattering cross sections using Wasserstein generative adversarial networks but had limited success in reducing the nonunique predictions.[29] In some situations, these disparate predictions may make the implementation easier; however, nonunique solutions drastically decrease the performance of the shape recognition algorithm in the far-field scattering problem. Metaneural networks have also been proposed for real-time recognition of complex objects such as handwritten digits and misaligned orbital angular momentum (OAM) beams from acoustic scattering.[30] Different configurations of deep neural networks have been designed and tested to predict the far-field scattering from 2D and 3D arbitrary objects,[31,32] but the inverse scattering problem remains to be investigated.

This study aims at solving the inverse scattering problem with deep learning that “uniquely” determines the shape of the 2D object from solely its phaseless far-field information. To address the intrinsic one-to-many mapping problem,[33] we feed multifrequency far-field data into the training process, thereby eliminating the nonunique solution space in final predictions. The inverse design procedure is as follows. First, we encode the structural properties of the arbitrary shaped object in lower-dimensional latent vector, z, using the adversarial autoencoder whose decoder part acts as a generator, G, in the inverse design strategy (see Figure 1b). The designed adversarial autoencoder imposes the condition of Gaussian distribution on the latent space to each randomly generated geometry and expedite the learning process from the shape of the object to a given far-field profile. Second, we employ a forward neural network (FNN) that acts as a physics predictor to evaluate the inverse design process (see Figure 1c). In the final step, we probabilistically train an inverse neural network (INN) followed by the pretrained generator and forward simulator. After all networks are trained, the far-field patterns are fed as an input to determine the latent space which predicts the generated design to the corresponding geometry (see Figure 1c). This study presents a powerful method for modeling inverse far-field scattering that allows fast and accurate predictions of the random shape of objects from scattering data for various applications.

2. Results and Discussion

We consider a 2D arbitrary-shaped scatterer made of steel immersed in water. Our goal is to build a deep learning model that can instantaneously provide information on the shape of the scatterer when illuminated by an acoustic plane wave. The elastic scatterer in water environment, of dimension around 40 cm, is discretized into a pixel-based binary image where we assign the pixel value 0 (for water) or 1 (for steel). The neural network is fed a binary image of the scatterer’s shape as the input. The output of the network represents the directivity of the radiation pattern in the angular range from 0° to 360°. A neural network can be designed to learn the isomorphic relation between the input and the output of the considered acoustic system, as schematically described in Figure 1a.

To train the network, we need to prepare training samples. We randomly generate 20 000 geometries to ensure diversity in training datasets. The far-field radiation properties of these random geometries are simulated with COMSOL Multiphysics at five different frequencies \( f_1 = 1\, \text{kHz}, \ f_2 = 1.5\, \text{kHz}, \ f_3 = 2\, \text{kHz}, \ f_4 = 2.5\, \text{kHz}, \ \text{and} \ f_5 = 3\, \text{kHz}. \) The far-field dataset contains 87 discrete points equally distributed in the full angular range (from 0° to 360°). Each training sample consists of a pair of model input and the expected corresponding output, that is, the 2D geometry and the far-field radiation pattern. The left panel in Figure 1a illustrates an example of the structure with the red area representing steel and the corresponding radiation characteristics. The matrix of the structure is 64 × 64. The full dataset contains 16 000 (80%) training samples, 2000 (10%) validation samples, and the remaining 2000 (10%) testing samples. The validation dataset monitors the overfitting during training and helps to tune the hyperparameters of the network. The testing set evaluates the performance of the trained network.
The proposed deep learning model involves the training of adversarial autoencoder (AAE), forward neural network (FNN), and inverse neural network (INN). The AAE deals with the dimensionality reduction problem and learns the latent distribution of the arbitrary-shaped object to generate the geometric patterns. The FNN deals with the regression problem between the 2D object with the matrix of dimension $64 \times 64$ (equivalent to $1 \times 4096$-dimensional vector) and the multifrequency far-field scattering amplitudes with the dimension of $1 \times 435$. In fact, we found that the information contained in the total scattering cross section is not enough to determine unequivocally the shape of the object due to non-one-to-one correspondence. Therefore, we included angular-dependent scattering amplitude at more frequency points (i.e., radiative patterns or scattering amplitude, at different frequencies). The INN relies on the generator of AAE and FNN to predict the shape of an arbitrary structure of dimension $64 \times 64$ for the given multifrequency far-field amplitudes of dimension $1 \times 435$. We will discuss the training of these three networks in the following sections.

### 2.1. Adversarial Autoencoder (AAE)

We use an AAE\cite{35} to learn the distribution of random geometries whose training is based on traditional reconstruction and an adversarial regularization. The architecture of AAE is composed
of three fully connected coupled neural networks: encoder, generator, and discriminator, as shown in Figure 1b. In the AAE, the encoder transforms a given input geometry into a compressed continuous design space (a latent space); the generator reconstructs the real space geometry from a given latent space; and the discriminator forces the latent space to follow a specified prior distribution by an adversarial manner.

The training process of AAE includes both adversarial learning of generative adversarial network (GANs) and latent distribution learning of variational autoencoders (VAEs). In this sense, the AAE architecture incorporates the properties of VAEs and GANs. Compared to other deep generative networks, such as VAEs and GANs, AAE networks possess several advantages. The AAE is more flexible than VAEs because it offers the freedom to control the latent space distribution without imposing any restrictions. Unlike vanilla GANs that skip the representation learning in a latent space, AAE produces a dense (continuous) latent space that can be used to generate diverse designs. In addition, AAE networks are easier to train than GANs because the target of adversarial learning focuses on the representation in the latent space, rather than the generation output. The AAE training procedure involves updating autoencoder (encoder + generator) and discriminator network by optimizing the loss functions of each network. The weights of discriminator network are updated to minimize a binary cross-entropy (BCE) loss, so that it can distinguish whether the sample comes from the encoder’s latent space or a sampled distribution. Specifically, the loss function of discriminator to minimize is \( \mathcal{L}_D = -\log(\hat{y}_s) - \log(1 - \hat{y}_l) \), where \( \hat{y}_s \) is the prediction made by the discriminator for a sample from the prior distribution (expected to be 1), and \( \hat{y}_l \) is the prediction made by the discriminator for a sample from an encoded input in the latent space (expected to be 0). Alternately, the autoencoder weights are updated to minimize \( \mathcal{L}_A = -\log(\hat{y}_l) - \mathcal{L}_r \). The first term is to make the autoencoder produce sample-encoding vectors similar to samples from a prior distribution (fooling the discriminator to make prediction \( \hat{y}_l \) be close to 1). The second term measures how good the autoencoder can reconstruct the

---

![Figure 2](https://www.advancedsciencenews.com/)

**Figure 2.** Designed AAE to learn the latent space of arbitrary binary structures. a) The binary cross-entropy loss function of the generator over training epochs. b) The binary cross-entropy loss function of the discriminator over training epochs. c) The binary cross-entropy error of testing samples. d) The structure similarity index measure (SSIM) on the test samples where the red dashed line shows the average error. e–f) Representative examples of original and reconstructed shapes of arbitrary objects from trained AAE network.
arbitrary input object from the latent space. Concretely, the reconstruction error is defined based on a binary cross-entropy loss expressed as \( \mathcal{L}_r = \frac{1}{N} \sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \) where \( N \) is the size of input (also the size of output), \( y_i \) is the ground truth of the \( i \)th pixel in input, \( \hat{y}_i = 0 \) represents the \( i \)th pixel that belongs to water background and \( \hat{y}_i = 1 \) represents the one belonging to steel object and \( \hat{y}_i \in (0,1) \) is the predicted value of the \( i \)th pixel in the reconstruction output. When \( \hat{y}_i \) approaches 1, the \( i \)th pixel has a greater likelihood of belonging to a steel object.

When the AAE network is trained, the generator will be able to generate designs in real space based on the latent vector. The trained generator is extracted from AAE and then used in the inverse design process. The training and prediction results for AAE network are shown in Figure 2. Figure 2a,b show the loss of the generator and discriminator network during the training process respectively. The loss function of the generator smoothly converges as the number of epochs increase; however, the loss of the discriminator fluctuates due to adversarial learning. Ideally, the discriminator loss should be close to 0.5 since it distinguishes between real and fake images during training. In order to evaluate the performance of the trained AAE, we apply it on testing samples and show the BCE error and the structure similarity index measure (SSIM) of testing samples in Figure 2c,d respectively. The details of the SSIM are provided in Supporting Information. The mean average values of BCE and SSIM are found to be 0.02 and 0.94, respectively. Two representative examples of the reconstructed geometries are shown in Figure 2e,f. The results clearly indicate the excellent prediction performance of the trained network.

2.2. Forward Neural Network (FNN)

Forward modeling consists of predicting the response of the given physical system. Different from the traditional approach,
the data-driven approaches predict the acoustic response of a given structure without explicitly solving the acoustic wave equations. To be more specific, we design a deep neural network that solves the regression problem deterministically by determining the far-field radiation patterns for a given arbitrary structure. The network learns the complex relation between arbitrary 2D binary structures and their associated multifrequency far-field radiation patterns. These patterns are invariant under translation of the object, so each binary structure is flattened to feed the fully connected network as an input.

For the training process, mean squared error (MSE) is used as the loss function that computes the average squared difference between the actual and the estimated values, that is, \( \mathcal{L} = \frac{1}{N} \sum_i (F_i - \hat{F}_i)^2 \) where \( N \) is size of output spectrum and \( F_i \) and \( \hat{F}_i \) are the ground truth and the predicted far-field pattern, respectively. The architecture of FNN contains 4096 nodes and the details of the training process and network hyper parameters are provided in Supporting Information. [38] The learning behavior as function of different epochs is shown in Figure 3a. In order to verify the validity and accuracy of the designed network, we use the test dataset to calculate the relative absolute error as follows: \( e = \sum_i |F_i - \hat{F}_i|/F_i \) where \( F_i \) and \( \hat{F}_i \) correspond to the target and the predicted far-field response, respectively. Figure 3b illustrates the distribution of error, and the corresponding mean relative error of the test set predictions is below 0.04, as indicated by the dashed vertical red line. A typical example shows a good match between the COMSOL Multiphysics simulation and the FNN prediction results (see Figure 3c). An accurate training of FNN is crucial because the following inverse design approach uses the trained FN as an essential component for accurate and instant prediction of far-field patterns for a given object.

### 2.3. Inverse Neural Network (INN)

The next step extends deep learning to solve the inverse design problem, which is essentially the inverse process of the forward modeling. While the forward modeling deals with a one-to-one mapping between a physical system and the resultant response, the inverse design is facing the difficulty induced by the nonunique solution spaces. Therefore, a single discriminative network is not able to learn the complex relation in the inverse design. To circumvent this problem, auxiliary training approaches, generative models, and optimization strategies are combined in the inverse design process. [39–44] The most common method to overcome the nonuniqueness issue consists of exploiting a tandem network that incorporates the forward modeling network into the inverse design DNN architecture. In conventional tandem networks, direct learning of the pixel-binary structure is not trivial due to the high degree of freedom in parameter design. In addition, the conventional tandem

![Figure 4](https://www.advancedsciencenews.com/figure4.png) **Figure 4.** Designed INN for shape recognition from given multifrequency far-field patterns. a) Evolution of the loss function of the generative inverse model over training epochs. b) The distribution of absolute prediction error for reconstructed far-field patterns on testing samples. c) Binary cross-entropy error of testing samples. d) Structure similarity index on predicted binary structures of testing samples. The red dashed lines in (b–d) indicate the average errors.
networks do not guarantee the complete elimination of nonunique solution space. Here, we propose a new solution of generative tandem network which learns the mapping between the far-field patterns and the latent distribution of a binary structure by incorporating an additional pretrained network, that is, the shape generator extracted from the trained AAE, into our inverse design strategy. The INN translates the input far-field patterns into the latent representation $z$, sampled from the Gaussian prior distribution parametrized by mean $\mu$ and standard deviation $\sigma$, to approximate the latent space of an arbitrary geometry as illustrated in Figure 1c.

The loss function for the INN is constructed by absolute loss for reconstruction of far-field patterns, $\mathcal{L}_{\text{recon}}$ and Kullback–Leibler (KL) divergence $\mathcal{L}_{\text{KL}}$, between the latent space.

![Figure 5](https://example.com/figure5.png)

**Figure 5.** Examples of generated shapes from given multifrequency far-field patterns using INN. i) Target structure. ii) Inversely generated structure from far-field patterns. iii–vii) Comparison of the simulated (solid black curve) and predicted far-field pattern (dashed red curves) from the generated geometries in (a,b).
distribution and the prior Gaussian distribution, \( N(0,1) \). To be specific, the training of INN is to minimize the following loss function expressed as

\[
\mathscr{L} = \frac{1}{N} \sum_{n=1}^{N} \left( \mathscr{L}_{\text{recon}}^{(n)} + \alpha \mathscr{L}_{\text{KL}}^{(n)} \right)
\]  

(1)

where \( N \) is the total number of training samples, \( n \) is the \( n \)th data sample, and \( \alpha \) is the relative weight between deterministic and generative learning. \( \mathscr{L}_{\text{recon}} \) is the reconstruction loss computed as \( \mathscr{L}_{\text{recon}}^{(n)} = \sum_{k=1}^{K} \left| f_k^{(n)} - \hat{f}_k^{(n)} \right| \) where \( K \) is the length of input far-field pattern, \( f_i \) and \( \hat{f}_i \) are the ground truth and the predicted far-field pattern from the generated object, respectively. KL divergence for Gaussian latent distribution is expressed as \( \mathscr{L}_{\text{KL}}^{(n)} = -\frac{1}{2} \sum_{j=1}^{f} \left( 1 + \left( \sigma_j^{(n)} \right)^2 - \left( \mu_j^{(n)} \right)^2 - \log(\sigma_j^{(n)})^2 \right) \), where \( f \) is the length of latent vector and \( \sigma \) and \( \mu \) are the standard deviation and mean of the distribution, respectively. The details about the stochastic tuning of the model are provided in the Experimental Section.

The INN architecture is composed of eight layers with each layer having 435 – 800 – 800 – 500 – 500 – 500 – 400 – 100 nodes. The intermediate latent space layer acts as a design layer, which is then passed into the generator and forward modeling part to calculate the corresponding far-field pattern. The learning curve is shown in Figure 4a. The rapidly decreasing behavior of learning curve demonstrates that training is highly effective. To evaluate the performance of the INN, we calculate the prediction errors on inversely generated far-field patterns, that is, relative absolute error, and on the predicted binary structure, that is, binary cross entropy error, and on the predicted angular structure similarity index measure. The mean percent error of inverse-designed far-field patterns is around 0.03 (see Figure 4b), while the average BCE error and SSIM are 0.077 and 0.85, respectively (see Figure 4c,d). In the inverse process, the learning behavior of the network depends on the intrinsically degenerate solution spaces. When degenerate solution spaces are reduced, prediction accuracy of the network to determine the unique arbitrary object is increased. A similarity metric of 0.85 over the test data indicates that the nonunique solution space has been reduced dramatically and that the target object is generated effectively.

Figure 5 illustrates two examples to showcase how the trained inverse network accurately determines the shape of an object for the given multifrequency far-field patterns. The predicted far-field patterns from the generated shapes are provided to compare with the given far-field patterns. As we can see, inversely designed objects (generated) for the given far-field patterns match well with the target objects, which confirms the efficacy of our method. The small variations around the edges barely alter the corresponding far-field information (see Figure 5a,b: (iii)–(viii)).

Next, we study the effect of feeding the multifrequency data for “unique” shape recognition. We investigate the performance of the INN by starting with single-frequency far-field data, 1 kHz, and then continuously increasing the number of frequencies in discrete steps of 0.5 kHz. For brevity, this study limits the INN results to five frequencies. The INN results are shown in Figure 6. We observe that the performance of the network to uniquely identify a structure is improved as the number of frequencies considered for the far-field data increases. Figure 6a clearly shows that the binary cross-entropy error used to identify an arbitrary shape sharply decreases when the second frequency is added and then slowly saturates as more frequencies are taken into account. A similar trend is observed with SSIM, where more frequencies make the SSIM approach toward unity (see Figure 6b). Thus, we can conclude that by adding more frequencies in the network training, the network gives better results. We found five frequencies are sufficient for an optimal performance (further increasing the frequencies number does not noticeably improve the performance). The detailed results for the trained FNN and INN for multifrequency data are provided in Supporting Information.[38]

In reality, it may not be feasible to measure multifrequency far-field data over the entire angular range from 0° to 360°. Therefore, analyzing the performance of the network with partial angular field data is crucial. For the sake of demonstration, we provide far-field data during training in a reduced angular range from 0° to 180° (see inset in Figure 7a), and the results of training the network are shown in Figure 7. Although the average error of

![Figure 6. Comparison of INN performance with multifrequency far-field data. a) Frequency-dependent Binary cross entropy error and b) structure similarity index measure (SSIM). BCE error decreases and SSIM increases with feeding multifrequency far-field data to the neural network where \( f_1 = 1 \) kHz, \( f_2 = 1.5 \) kHz, \( f_3 = 2 \) kHz, \( f_4 = 2.5 \) kHz, and \( f_5 = 3 \) kHz. The accuracy of INN to recognize the shape is significantly improved with multifrequency data.](image-url)
prediction is slightly compromised because of reduced information used for learning in the training, the proposed method can still accurately identify the key features of the considered arbitrary objects. The detailed results are shown in Supporting Information.

3. Conclusion

The inverse scattering problem of retrieving the shape from the far-field information is highly nonlinear and extremely challenging to solve with conventional approaches due to nonuniqueness issues. Here, we propose a generative deep learning approach as a practical design tool, to uniquely determine the shape of an arbitrary object with multifrequency phaseless data. We exploit generative adversarial learning to encode the true features of objects into the latent space through adversarial autoencoder and further integrate its generator into the inverse design process to create random shapes. The forward network is designed to learn the relation between a given structure and the corresponding far-field profile for instant and accurate predictions. The inverse design strategy is based on a generative encoder-decoder-like architecture, where the encoder (i.e., inverse network) is trained while fixing the decoder, consisting of the pretrained forward network and generator. We study the influence of feeding multifrequency far-field data on shape recognition and show that multifrequency data rules out the nonunique solution spaces in the inverse architecture. In addition, we demonstrate that half-angular far-field data (i.e., from 0° to 180°) is still capable of uniquely determining the shape of the arbitrary object. To accelerate inverse learning, the proposed approach uses a latent distribution rather than a higher-dimensional object, which differs from conventional tandem architectures. The designed network instantly predicts the shape of the object for the given far-field information with unprecedented speed, reducing the design time by four orders of magnitude over traditional methods. It is important to emphasize that all our simulations are scalable, and the choice of frequencies depends on

Figure 7. FNN and INN results for multifrequency half-plane far-field data. The distribution of absolute relative error to predict the far-field through trained a) FNN and b) INN. The inset in (a) shows the half-plane multifrequency far-field patterns for an arbitrary-shaped object. The binary cross-entropy error and structure similarity index measure of INN to recognize the shape are shown in (c,d), respectively. e–f) Examples of generated geometries for the given far-field patterns. (See the Supporting Information for more details).
For instance, in our previous work,[25] we predicted the shape based on a particular material label. As a result, the output will predict the shape along with label values 0, 1, 2 to each material instead of bulk modulus or density. As a result, the output will predict the shape along with label values 0, 1, 2 to each material instead of bulk modulus or density. As long as the material contrast with the host medium (water in this case) is not significantly large such that the scattering patterns can be distinguished from other materials, the method can learn the material properties from the scattering features. Moreover, a specific material set can be treated as a classification problem to predict the shape based on a particular material label. For instance, in our previous work,[25] we predicted materials within a bounded modulus and density range. However, if we specify the material set (such as aluminum, steel, rubber, etc.), then we can treat it as a classification problem by assigning label values 0, 1, 2 to each material instead of bulk modulus or density. As a result, the output will predict the shape along with the target label (0, 1, 2, ...) corresponding to real materials.

4. Experimental Section

Adversarial Learning Method: In AAE, the encoder takes a real image $x \in \mathbb{R}^d$ as an input, compresses it into a $p$-dimensional latent space $z \in \mathbb{R}^p$ (where $p < d$), and then the generator reconstructs the image $\hat{x} \in \mathbb{R}^d$. Let $q(z)$ be the posterior distribution of latent space, $z$, in autoencoder and $p(z)$ be user-specified prior distribution, which is assumed to be Gaussian distribution. The encoder generates the latent space from the posterior distribution $z = p(z)$ and the discriminator $D$ distinguishes $z$ to be a real from the prior distribution $p(z)$ or generated by the encoder as follows.

$$
D(z) = \begin{cases} 
1 & \text{if } z \approx p(z) \\
0 & \text{if } z \approx q(z) 
\end{cases}
$$

(2)

AAE attempts to generate $z$ analogous to the real latent space from the prior distribution via adversarial learning.

Variational Inference Method: The inverse network exploits the variational inference method to learn the distribution of far-field response and its mapping with the latent space, $z$, of the arbitrary object. The reconstruction loss is the absolute error between the input far-field patterns and the reconstructed patterns by the INN. The learned distribution $q(z|x)$ equivalent to the predefined distribution $p(z)$ is ensured through KL divergence term in loss function expressed as

$$
\mathcal{L}_{\text{KL}}^{(n)}(z) = \frac{1}{2} \sum_p \left( \sigma_p^2 + \mu_p^2 - \log(\sigma_p^2) + 1 \right)
$$

(4)

where $\mu$ and $\sigma$ are the mean and the standard deviation of the generated latent space distribution, respectively. The reparameterization trick helps to determine $z = \mu_p + \sigma_p \epsilon$, where $\epsilon$ is a sample from prior Gaussian distribution, during the optimization process. The reconstruction term is expressed as the MSE between the input coding matrix and the reconstructed coding matrix output by the decoder.

Numerical Modeling: The finite-element method was used to perform the full wave simulations with COMSOL Multiphysics. We assumed water as the background medium in our simulation setup with mass density $\rho = 1000 \text{ kg m}^{-3}$ and bulk modulus $\kappa = 2.19 \text{ GPa}$. The arbitrary object was made of steel with mass density $\rho = 7850 \text{ kg m}^{-3}$ and bulk modulus $\kappa = 201 \text{ GPa}$. The steel object in water produces shear waves so we modeled the steel with solid mechanics modules and background medium with acoustic module. The integration of acoustic module and solid mechanics module was done with Multiphysics module for full wave modelling, via the acoustic–structure boundary. The arbitrary object was excited with an acoustic plane wave of unit amplitude to determine the far-field response at different operating frequencies. (See Supporting Information for details on the derivation and governing equations).

Supporting Information
Supporting Information is available from the Wiley Online Library or from the author.

Acknowledgements

The work described in here was supported by King Abdullah University of Science and Technology (KAUST) Artificial Intelligence Initiative Fund, KAUST Office of Sponsored Research (OSR) under grant no. OSR-2020-CRG9-4374, and KAUST Baseline Research Fund no. BAS/1/1626-01-01.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

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

Keywords

acoustic scattering, deep learning, inverse scattering problems, shape recognition

Received: September 5, 2022
Revised: December 21, 2022
Published online: January 20, 2023

[1] D. Colton, R. Kress, Inverse Acoustic and Electromagnetic Scattering Theory, 3rd ed., Springer, Berlin 2013.
