


## Photonic Inverse Design with Neural Networks: The Case of Invisibility in the Visible

Arsen Sheverdin,<sup>1</sup> Francesco Monticone,<sup>2</sup> and Constantinos Valagiannopoulos<sup>1,\*</sup>

<sup>1</sup>*Department of Physics, Nazarbayev University, Nur-Sultan, KZ 010000, Kazakhstan*

<sup>2</sup>*School of Electrical and Computer Engineering, Cornell University, Ithaca, New York 14853, USA*

 (Received 12 May 2020; revised 11 July 2020; accepted 15 July 2020; published 19 August 2020)

Artificial intelligence is currently attracting unprecedented attention for its ability to tackle hard problems with huge datasets, which have been rendered tractable by the giant computational power and amount of training data available today. Photonic inverse design, in which one seeks to find objects of desired electromagnetic response, belongs to this class of complex problems that can greatly benefit from these ideas. In this work, artificial intelligence concepts are applied to advance the quest for invisible particles that do not perturb the background field; in particular, a fully connected neural network is proposed to address such a problem by learning the dynamics of visible-light interaction with low-scattering multilayered nanospheres. By swapping the roles between inputs and outputs, the same network can then be used for the inverse design of invisible nanoparticles, obtaining superior performance with respect to the best elements of the training set. The proposed approach can be generalized to approximate Maxwell interactions by simulating the electromagnetic response of more complicated optical configurations, and accomplish their inverse design directly, without successive iterations.

DOI: [10.1103/PhysRevApplied.14.024054](https://doi.org/10.1103/PhysRevApplied.14.024054)

### I. INTRODUCTION

A long-lasting goal of researchers across a broad range of disciplines has been to engineer machines able to operate not by following an explicit list of commands but by identifying patterns and deciding their response, based on their own “experience” and “training,” similarly to the operation of the human brain. The introduction of system models that adaptively improve themselves to recognize patterns not specified in advance [1], and devices that mimic humanlike functions as perception [2], were the first significant steps towards such an objective. These old concepts focused on enabling computing systems to perceive a well-described environment [3] and paved the way to the systematic formulation of multiple machine learning approaches, including artificial neural networks. The principle of their operation is based on the approximation of any arbitrary distribution by suitably predicted weighted sums of nonlinear basis functions [4,5]; it can be accomplished through a learning operator that allows the system to translate errors from task space into weights space. A major tool to decide the contribution of basis functions to the final sum is the so-called back-propagation algorithm [6], enabling the output to improve the utilized approximations. In this way, integrated neural networks that can “learn” from the dataset and generalize from examples

were proposed [7] and implemented successfully in document recognition, one of the main challenges in the early days of this field [8]. In recent years, neural networks have been experiencing an incredible popularity boost [9,10] and have attracted substantial funding interest [11], since the available computational power levels have rendered feasible the treatment of complex, real-world problems with huge datasets.

Increased computing speeds assisted by more efficient hardware have not only led to finer implementation of machine learning algorithms in traditional applications like speech recognition [12] and scene labeling [13], but also to the development of artificial neural networks for tackling of problems originating from diverse scientific domains [14]. Striking examples include deep learning concepts pertinent to medical image analysis [15] contributing to more reliable brain tumor localization [16], better identification of hidden structure within biological data [17], and more accurate computer-accommodated drug discovery and design [18]. The same techniques are also extremely useful in approaching computationally demanding problems in physics, especially those suffering from the so-called “curse of dimensionality” [19] like quantum many-body interactions [20], prediction of complex spatiotemporal states [21], and probabilistic Hamiltonian estimation [22]. Even market predictions and financial analyses [23] can benefit from recent machine learning advances, resulting in better portfolio management, fast

\*valagiannopoulos@gmail.com

pricing optimization, and improved estimation of macroeconomic quantities [24].

Photonics is not an exception in that global trend; indeed, numerous neural network-based solutions have been proposed to complicated problems, such as precise optimization of magneto-optic cooling [25] and accelerated solution of frequency-domain Maxwell's equations in complicated structures [26]. In particular, the inherently challenging inverse design process greatly benefited from machine learning approaches as exemplified by the successful guess of multilayers with the desired transmission spectra [27] or the derivation of meta-atom shapes for a metasurface with specific reflectivity response [28]. Indeed, metasurfaces, due to their ubiquitous operation, commonly employ deep learning technologies in applications like amplifying chiral optical responses [29], tailoring Fano resonances [30], and sharp transmissive sensing [31]. Alternatively, one can adopt a bottom-up optimization by inversely designing isolated nanoparticles prior to embedding them in lattice configurations supporting collective functionalities. Such a strategy has been followed to learn and simulate Maxwell interactions, leading to core-shell [32] and multilayered nanospheres [33] that optimally mimic a target scattering lineshape, even though the training is based on a different configuration [34].

A particularly challenging objective when designing photonic nanoparticles is to render them “invisible” across a wide frequency band by suppressing their scattering response. One of the most popular approaches to achieve invisibility is the so-called optical conformal mapping or transformation optics, which allows for designing material distributions that force light rays, or full electromagnetic fields, to propagate around an object without interacting with it [35,36]. In practice, such a tailoring of the space can be feasibly engineered with elliptical inclusions [37] and nonresonant elements [38]. Several alternative strategies have been proposed during the last two decades to serve similar cloaking purposes: from scattering cancellation with the use of plasmonic coatings [39,40] to transmission line waveguides [41] and homogenized analogues [42] that make the fields travel through the cloaked areas with a strongly suppressed scattering response. Furthermore, cloaking of bumps on surfaces has been realized by distorting the propagation space [43] or deploying homogeneous superstrates [44]. Finally, designs based on active media have been proposed [45,46] to suppress the scattering cross sections over broader bandwidths and for larger objects [47].

In this work, we propose and demonstrate the use of artificial intelligence to advance the quest for invisibility, namely to find spherical nanoparticles that do not significantly perturb the background electromagnetic fields. In particular, we train a fully connected neural network by feeding it with the scattering spectra of suitably selected

multilayered nanospheres, and accordingly its weights are optimally determined. The physical configuration of the particle comprises a few layers of predetermined materials and our goal is to suppress the scattering across frequency points in the visible range of the electromagnetic spectrum. As discussed in detail below, we find that the inverse process is performed better if one flips the network setup and treats the wavelength-dependent scattering efficiency as the true input and the layer thicknesses as the true outputs. In this way, several cloaked nanoparticles practically invisible under visible light are determined and the efficient inverse design for such a counterintuitive photonic operation is demonstrated.

## II. BASICS OF ARTIFICIAL NEURAL NETWORKS

Artificial neural networks are called so due to the qualitative similarity of their computational process with the way that the human brain operates. Each neural network is comprised of myriads of elementary modules, called neurons, whose sole job is to produce an output by applying a nonlinear function  $y = g(x)$  on their input. The major task of an artificial neural network is to mimic a system possessing multiple ( $D$ ) inputs and multiple ( $R$ ) outputs by employing several ( $L$ ) sequential layers, each of which contains numerous ( $U_l$  with  $l = 1, \dots, L$ ) neurons [48], as indicated by Fig. 1(a). The input of each neuron is the weighted sum of all the outputs from the previous layer, while the neurons of the same layer are not connected to each other. In particular, the input  $x_u^{(l)}$  of the  $u$ th neuron of the  $l$ th layer ( $u = 1, \dots, U_l$ ) reads  $x_u^{(l)} = \sum_{v=1}^{U_{l-1}} w_{vu}^{(l)} y_v^{(l-1)}$ , where the  $w_{vu}^{(l)}$  are weights to be determined and  $y_v^{(l-1)}$  is the output of the  $v$ th neuron located at the  $(l-1)$ th layer. The total number of weights to be found across the network is equal to

$$W = U_1 D + \sum_{l=1}^{L-1} U_l U_{l+1} + U_L R. \quad (1)$$

We use the symbol  $\mathbf{w}$  for the  $W \times 1$  vector of weights, as shown in Fig. 1(a).

A metric of how successful our network imitates the operation of the system, given a specific set of inputs, is obviously the difference between its own response and the actual output of the original system; accordingly, the mean square error from all the  $R$  outputs, denoted by  $G(\mathbf{w})$ , serves such a purpose. The weights are modified following the simple gradient descent rule

$$w_{vu}^{(l)}(\tau + 1) = w_{vu}^{(l)}(\tau) - \frac{\partial G}{\partial w_{vu}^{(l)}} \alpha, \quad (2)$$

where  $\tau$  is the update step number and  $\alpha > 0$  is the learning rate defining the impact of current error on the weight change. Because of the analytical nature of the involved

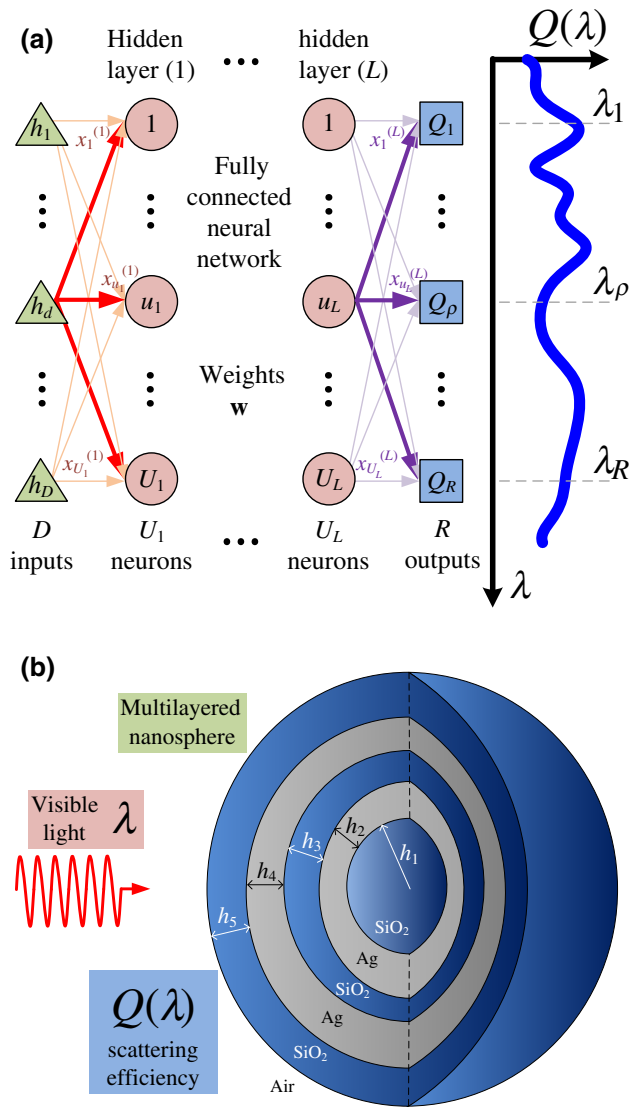


FIG. 1. (a) A typical setup of a fully connected neural network with  $D$  inputs and  $R$  outputs, which constitute a plot  $Q(\lambda)$  along variable  $\lambda$  with  $R$  samples and  $L$  hidden layers comprising  $U_l$  neurons each ( $l = 1, \dots, L$ ). (b) Physical configuration of the nanoparticle subject to optimization: a five-layered sphere scatters a plane wave of wavelength  $\lambda$  and is characterized by scattering efficiency  $Q(\lambda)$ .

functions, the derivative of  $G$  with respect to the weight  $w_{vu}^{(l)}$  is determined implicitly via the back-propagation algorithm by iteratively implementing the chain rule and evaluating the local sensitivities numerically [6]. It should be stressed that the activation function  $g$ , according to which any neuron operates, must be nonlinear; otherwise, all the  $L$  layers will effectively collapse to one and the approximation capability of the network diminishes to that of a linear regression. The number of nodes  $D$ ,  $R$ , and  $U_l$  for  $l = 1, \dots, L$ , the number of hidden layers  $L$ , and the learning ratio  $\alpha$  are characteristics of the setup and are called hyperparameters of the network.

The artificial neural network manages to mimic the overall system only after its weights  $\mathbf{w}$  are repeatedly updated with use of many data samples, namely, numerical sets pairing  $D$  inputs with  $R$  outputs. In order to perform a fair assessment of the network's predictive performance, we save a small portion of the data samples for validation purposes (validation set) and we work with the remaining  $M$  members. The learning process concerns the modification of weights  $w_{vu}^{(l)}$  according to rule (2), by applying it for each subset of population  $B$  from the total of  $M$  samples. The data are processed in random batches of size  $B$ , which is selected neither too small ( $B \cong 1$ ), making the learning process nonvulnerable to outliers [49], nor too large ( $B \cong M$ ), offering the necessary noisiness to escape local minima [50]. Note that each data sample interacts, on average, with our network not once but many ( $E$ ) times so that the impact of random selection and sample sequence is milder. More specifically, the training process, namely the evolution of  $w_{vu}^{(l)}(\tau)$ , stops not earlier than  $E \times \lfloor M/B \rfloor$  batch updates along the  $\tau$  axis. The integer  $E$  is the number of epochs and is decided by checking if the error on the validation set has reached a low threshold. Indeed, a huge number of epochs  $E$  would make our network overspecialized to specific classes of data and less able to generalize and react successfully to unseen inputs.

Once the weights  $\mathbf{w}$  are properly determined, our network can be used as a substitute to the actual system and give reliable responses ( $R$  outputs) to the corresponding excitations ( $D$  inputs). Therefore, as long as the network is trained, it can be employed for inverse problems by optimizing the inputs instead of the weights. Indeed, if instead of the error  $G$  between the exact and estimated outputs, we use the difference between a specific desired response and the neural network outcome, the system inputs can be updated in a similar repetitive fashion. Alternatively, we can train the neural network by using exactly the same dataset with the difference that the roles of the  $D$  inputs and  $R$  outputs are flipped. In this way, the new weights  $\mathbf{w}' \neq \mathbf{w}$  will give those  $D$  parameters that, if fed to the original system, will produce  $R$  outcomes, which, according to the network, will be as close as possible to the desired ones. We may say that the network provides an answer for any inverse design in a forward manner: directly and without iterations.

### III. INVISIBILITY IN THE VISIBLE

Inverse design problems span many disciplines and have a huge importance in optimizing the functionality of physical systems. The same applies to photonics, which require inverse design to determine the best structural and textural combinations that maximize the performance in a variety of wave-matter interaction setups. Clearly, the inherent complexity of inverse design, when the number of

involved parameters is increasingly large, creates a bottleneck (curse of dimensionality [19]) in the search of better structures and devices for fundamental optic operations. Therefore, the introduction of neural networks in photonic inverse design can substantially advance the quest for optimal photonic devices by reducing the computational complexity of the design process. A significant class of inverse problems in photonics pertains to determining a physical configuration that possesses a specific frequency response over a certain working band [51].

A representative example is the objective of this work; it considers a multilayered free-standing spherical nanoparticle composed of two alternating materials that should be optimized to have a suppressed scattering response across the visible. In particular, we pick two of the most commonly used materials in optics (silica and silver) and assume a SiO<sub>2</sub> core covered by four alternating layers filled with Ag and SiO<sub>2</sub>, as indicated by Fig 1(b). The radius of the core and the thickness of each layer constitute the input vector  $\mathbf{h} = [h_1 \ h_2 \ h_3 \ h_4 \ h_5]$  of the system, namely,  $D = 5$ ; we are interested in operational wavelengths within the visible range:  $\lambda_{\min} \equiv 400 \text{ nm} < \lambda < 700 \text{ nm} \equiv \lambda_{\max}$ . The frequency dispersion and the losses of silver are taken into account by using reliable experimental data [52]; however, the results may not be entirely accurate if the thickness of the layers  $h_\rho$  is very small, since additional losses due to surface effects at the boundaries will be present (Landau damping) [53]. In order to accurately approximate the frequency response of the particle, we sample it with a step of 2 nm and thus the number of outputs is given by  $R = 151$ . We point out that, due to the low dimensionality of the input vector  $\mathbf{h}$  in this example, the use of artificial neural networks can be substituted with conventional steepest descent approaches implemented directly on a suitable system's metric. Such a statement is not accurate because (i) the analytical expression of the field solutions, even for such a simple particle, are rather complicated, involving several multiplications of matrices and highly nonlinear functions; (ii) the number of outputs is large, making the metric even more complex; and, most importantly, (iii) once trained, our neural network can predict inverse solutions for alternative desired responses in a single step, with no extra computational effort.

The spherical particle is illuminated by the simplest possible source: a plane wave of electric field amplitude  $E_0$ . It is thus possible to define in the centralized spherical coordinate system  $(r, \theta, \varphi)$  the scattered field for  $r > (h_1 + h_2 + h_3 + h_4 + h_5) \equiv b$  as a sum of two fields (TE and TM) whose electric and magnetic vectors respectively are defined as follows:

$$\mathbf{E}_{\text{scat}}^{\text{TE}} = E_0 \sum_{n=1}^{+\infty} S_n^{\text{TE}} h_n(k_0 r) \{-\hat{\theta} \csc \theta P_n(\theta) \cos \varphi + \hat{\phi} P_n'(\theta) \sin \varphi\},$$

$$\mathbf{H}_{\text{scat}}^{\text{TM}} = \frac{E_0}{\eta_0} \sum_{n=1}^{+\infty} S_n^{\text{TM}} h_n(k_0 r) \{-\hat{\theta} \csc \theta P_n(\theta) \sin \varphi - \hat{\phi} P_n'(\theta) \cos \varphi\}.$$

Here  $P_n(\theta)$  is the Legendre polynomial of first order, degree  $n$ , and argument  $\cos \theta$ , and  $h_n$  is the spherical Hankel function of order  $n$  and second type that corresponds to spherical waves of different multipolar order; it should not be confused with a similar symbol used for the thickness of each layer of the nanoparticles. The symbols  $k_0 = 2\pi/\lambda$  and  $\eta_0$  stand for the wave number and the wave impedance into the vacuum. The coefficients  $S_n^{\text{TE/TM}}$  are complex dimensionless quantities determinable by imposing the necessary boundary conditions; they are dependent on the oscillation wavelength  $\lambda$  and input vector  $\mathbf{h}$ , but their exact expressions are not shown here for brevity [54]. The scattering efficiency  $Q$  is defined as the ratio of the total scattered power by the particle over the power of the incident field passing through the cross section of the sphere [55]:

$$Q(\lambda) = \frac{2}{(k_0 b)^2} \sum_{n=1}^{+\infty} \frac{n^2(n+1)^2}{2n+1} (|S_n^{\text{TE}}|^2 + |S_n^{\text{TM}}|^2). \quad (3)$$

As implied above, the  $R$  outputs of our system equal the values of  $Q(\lambda)$  at  $\lambda = \lambda_\rho = \lambda_{\min} + (\lambda_{\max} - \lambda_{\min})(\rho/R)$  for  $\rho = 0, \dots, R$ . If our aim is to find invisible particles in the visible then the perfect cloaking scenario reads  $Q(\lambda_\rho) = 0$  for  $\rho = 0, \dots, R$ . The average scattering can be defined via the integral

$$\bar{Q} \equiv \frac{1}{\lambda_{\max} - \lambda_{\min}} \int_{\lambda_{\min}}^{\lambda_{\max}} Q(\lambda) d\lambda. \quad (4)$$

Since we search for nanospheres that minimally scatter the incoming light across the visible part of the wavelength spectrum, our objective is to suppress the scalar quantity  $\bar{Q}$ .

#### IV. NUMERICAL DEMONSTRATION

In an attempt to find multilayered spherical objects that practically do not scatter the incoming illumination across the entire visible range, we employ an artificial neural network setup with  $L = 4$  hidden layers, each of which incorporates  $U_1 = U_2 = U_3 = U_4 = U$  neurons. Since our target is to detect particles that have particularly low  $Q(\lambda)$ , our learning set comprises  $M \cong 34\,500$  low-scattering nanospheres with  $\bar{Q} < 0.1$ , picked from a random pool of scatterers optically moderate in size [56] with layers  $\{h_1, h_2, h_3, h_4, h_5\}$  belonging to the interval  $5 \text{ nm} < h_d < 80 \text{ nm}$  for  $d = 1, \dots, D$ ; the same parameters are considered for the validation set that is taken four times smaller. To generate such a large number  $M$  of samples



with relatively suppressed scattering efficiency, we applied the SMOTE (synthetic minority over-sampling technique) algorithm [57] to an initial small set that was compatible with this demand. Note that if the desired response  $Q(\lambda)$  is different from zero, it is better to keep the learning set completely random [33].

The search for perfectly cloaked nanoparticles is an inverse process and thus we can employ the two approaches mentioned above: (i) the repetitive approach with  $D$  inputs and  $R$  outputs, where the minimization of error between the predicted and desired responses is performed; and (ii) the direct approach with  $R$  inputs and  $D$  outputs, where the network is fed by the target and provides the optimal design. We tried each approach and the training procedure ran smoothly by readily determining the optimal weights, based on the minimal validation error. However, we found that the second direct approach, apart from the fact that it does not require further updates, provided us with much better inverse solutions at a faster time; such a finding may be related to the huge dimensional asymmetry between the input and output sets of the considered system ( $R \gg D$ ). Therefore, for the rest of the paper, we consider the case in which our inputs are  $R$  and concern the frequency response  $Q(\lambda)$  of the scatterer and our outputs are just the  $D$  thicknesses of the nanoparticle layers. The adopted inverse approach can directly give the best, according to the network, solution  $\mathbf{h}$  for any pattern  $Q(\lambda)$  of low values. The weights  $\mathbf{w}$  and  $\mathbf{w}'$  are updated by the PyTorch [58] built-in implementation of the Adam algorithm [59], allowing us to stochastically change the weights and making them converge to the local minima. Such software is primarily developed and supported by Facebook’s artificial intelligence (AI) research laboratory [60] in a free and open-source format [61].

In all the following numerical implementations we use training batches of size  $B = 64$ ; furthermore, in order to improve the convergence speed of the training, we utilize the batch normalization technique [62], with the default PyTorch algorithmic parameters. In addition, the dropout regularization technique [63] is employed, whose job is to randomly drop the units, together with their connections during the training, preventing the network from becoming too specialized on the features of the training set (over fitting).

In Fig. 2(a) we use the exact formulas (3) and (4) to compute the mean relative error (MRE) of the neural network prediction when tested on the validation set for various hardware configurations with different numbers of nodes  $U$  per hidden layer ( $L = 4$ ) and different learning rates  $\alpha$ . We directly note that a small learning rate  $\alpha$  facilitates a better training (smaller MRE); it seems that the system evades good solutions due to the large correction steps in the weights  $\mathbf{w}$  parametric space if  $\alpha$  is substantial. Note that the performance worsens further if a high  $\alpha$  is combined with an excessive number of neurons  $U$  since the

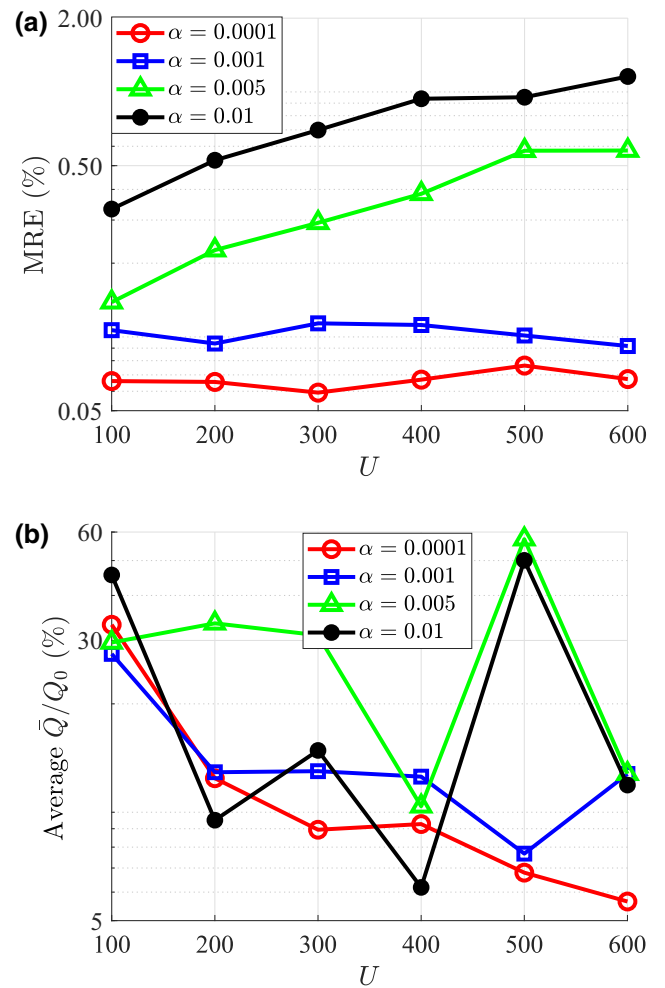


FIG. 2. (a) The mean relative error (MRE) of the neural network applied to the validation set, and (b) the average performance  $\bar{Q}$  of optimal particles corresponding to different data partitions, normalized by the performance of the most weakly scattering particle of the dataset  $Q_0 \cong 0.08$ , as a function of the number of neurons for each hidden layer  $U$  for various learning rates  $\alpha$ . The plot parameters are  $L = 4$ ,  $D = 151$ ,  $R = 5$ .

same poor training process is applied to a more complex setup (higher number of optimizable weights). In addition, an increased number of nodes  $U$  allows the system to more successfully mimic the training set, but may inflict “over fitting,” namely suppressing the ability of the network to generalize predictions.

It should be stressed that the random partition of data makes the response of the neural network somewhat dependent on it. Therefore, to evaluate the success of the implementation for each combination of  $(U, \alpha)$ , we train the network multiple times based on different partitions of the dataset, namely for different mutually exclusive validation sets. As a result, the weights  $\mathbf{w}$  are not identical each time and thus the question “which particle has the weakest scattering efficiency?” is answered differently each time.

In Fig. 2(b) we show the average  $\bar{Q}$  over all the proposed weakest scatterers for different partitions of the dataset and for networks with the same hyperparameter values as in Fig. 2(a). This quantity is normalized by the minimal scattering efficiency  $Q_0$  of the particles comprising the dataset and thus we can understand how many times better a proposed design is, compared to the best sample that we had available beforehand. A less reactive setup again exhibits a more desirable response since it tends to be more stable and robust for a sufficient population of nodes  $U$ ; on the contrary, the performance of a system with larger  $\alpha$  seems heavily dependent on  $U$ . As far as the absolute values of the ratio  $\bar{Q}/Q_0$  are concerned, in most cases they are much smaller than unity, demonstrating that the network has learned something about the system and can produce features it was not trained on.

It would also be relevant to identify the behavior of the average prediction error of our network when it is tested on unknown samples for various training sets and as a function of epochs  $E$ , namely the number of times that each member of the training set is, on average, fed to the network. In particular, we consider three distinct training sets: (i) the aforementioned  $M = 34\,500$  low-scattering nanospheres (targeted training set); (ii) a pool of particles whose thicknesses  $\mathbf{h}$  are randomly selected within the limits indicated by the objects with suppressed scattering, without checking their actual  $\bar{Q}$  scores (targeted random training set); and (iii) initial random pool of nanospheres with layer thicknesses selected uniformly within the ranges  $5\text{ nm} < h_\rho < 80\text{ nm}$  for  $\rho = 1, \dots, R$  with  $R = 5$  after the mirrorlike flip of the network (random training set).

In Fig. 3(a) we show the distribution of the overall scattering efficiency  $\bar{Q}$  for the three different training sets by plotting the corresponding relative frequency of occurrence  $0 < f(\bar{Q}) < 1$ . The red distribution concerns the aforementioned carefully selected particles based on their low-scattering performance; as indicated above, all members of this set respect the constraint  $\bar{Q} < 0.1$ . On the contrary, the nanospheres picked randomly in the parametric neighborhood of  $\mathbf{h}$  of the targeted training set (blue color) mostly give  $\bar{Q} > 0.1$  and violate the imposed inequality (blue distribution). As far as the totally random dataset (green color) is concerned, it gives tiny frequencies  $f$  for low  $\bar{Q}$  and tends to be mainly comprised of particles of substantial  $\bar{Q}$ , which is anticipated given the optically large average size of spheres  $5\text{ nm} < h_\rho < 80\text{ nm}$  for  $\rho = 1, \dots, R$  at  $400\text{ nm} < \lambda < 700\text{ nm}$ .

In Fig. 3(b) we represent the MRE on the (same) validation set for the aforementioned three training procedures as a function of the number of epochs; naturally, a declining trend with respect to  $E$  is recorded. The best, most efficient response is recorded for the targeted training set; such a result is expected since validation and training sets contain very similar samples. The estimation error is worse, but still very small, in the case of a targeted random

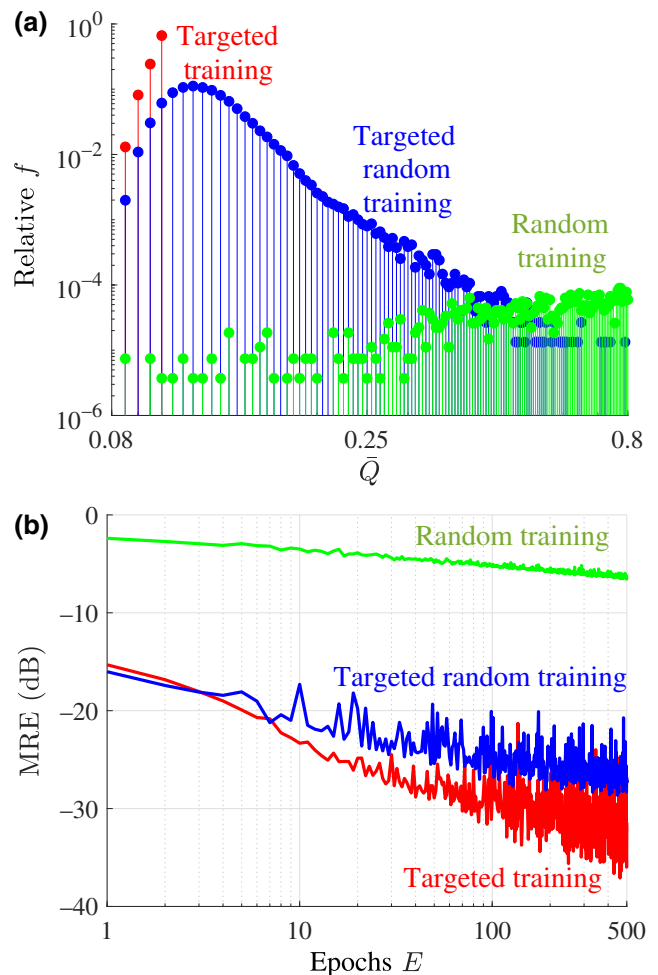


FIG. 3. (a) Relative frequency  $f$  of the samples belonging to three different training sets: totally random (green curve), targeted, comprised exclusively of low-scattering samples (red curve), and random around the average thicknesses  $\mathbf{h}$  of the targeted set (blue curve) in proportion to their overall scattering efficiency  $\bar{Q}$ . (b) Mean relative error in decibels as a function of the number of epochs  $E$  for the three aforementioned training sets (random, green; targeted, red; random targeted, blue). The testing set is the same in all cases and involves samples with suppressed  $\bar{Q}$ .

training set. Indeed, the low- $\bar{Q}$  training set suggests the existence of a “parametric neighborhood” for  $\mathbf{h}$ , where one can consistently find nanoparticles with suppressed scattering response; in other words, by operating locally, it is very likely to find objects of similar behavior. Finally, we show that  $\text{MRE} = \text{MRE}(E)$  for the random training set; the MRE is unacceptably high because the validation set occupies an extremely small and coherent part of the training set. This clearly illustrates why we need to use a carefully selected group of low- $\bar{Q}$  scatterers to train our network for predicting cloaked objects (red curve).

In Fig. 4(a) we test the inverse design capabilities of

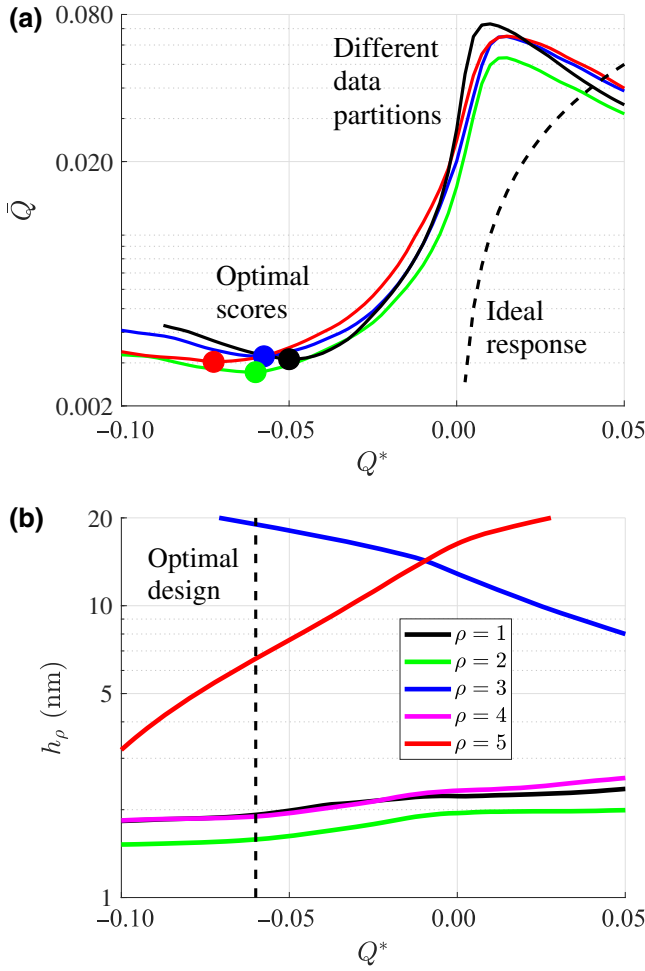


FIG. 4. (a) The scattering efficiency metric  $\bar{Q}$  of the proposed particle as a function of the desired flat output level  $Q^*$ . The dashed line corresponds to  $\bar{Q} = Q^*$ . (b) Optimal thicknesses of layers  $h_\rho$  for  $\rho = 1, \dots, R$  for the multilayered nanospheres of Fig. 4(a) (green curve) as a function of the selected  $Q^*$ . The dashed line corresponds to the design of minimal  $\bar{Q}$ . The plot parameters are  $L = 4$ ,  $D = 151$ ,  $R = 5$ .

our implementation with the hyperparameters of the network selected based on the results of Fig. 2. In particular, we show the exact  $\bar{Q}$ , based on Eqs. (3) and (4), for the samples answers of the neural network when the target response is flat,  $Q(\lambda) = Q^*$ , for the considered wavelength band. On the horizontal axis, we represent exactly this (small) constant level  $Q^* = Q(\lambda) = \bar{Q}$  and, ideally, the response of a perfect predicting system would be given by the dashed line of Fig. 4(a) along which  $\bar{Q} = Q^*$ . Interestingly, we find that the best results are obtained for negative values of the desired output  $Q^*$ ; such a selection of  $Q^* < 0$  is of course unnatural, but this does not bother our physics-agnostic neural network. On the other hand, we find that if  $Q^*$  is chosen to be very negative, the system is unable to predict acceptable values for shell thicknesses  $h_\rho > 0$ ,  $\rho = 1, \dots, 5$ ; thus, the curves in Fig. 4(a) possess a single

minimum, which is our network's answer. In Fig. 4(a) we show several of such answers, indicated by filled circles, each corresponding to a different partition of the dataset. It is evident that, regardless of the way we split the data in learning and validation sets, the network converges to well-cloaked particles with very low  $\bar{Q}$  scores. Note that the influence of the additional randomness introduced by the batch selection becomes unimportant due to the large number of epochs  $E$ .

In Fig. 4(b) we pick one of the partitions of Fig. 4(a) (green curve) and show the evolution of the layer thicknesses  $h_\rho > 0$  for  $\rho = 1, \dots, 5$  that the network predicts for various levels of flat target response  $Q(\lambda) = Q^*$ . Because of the higher absolute value of silver permittivity

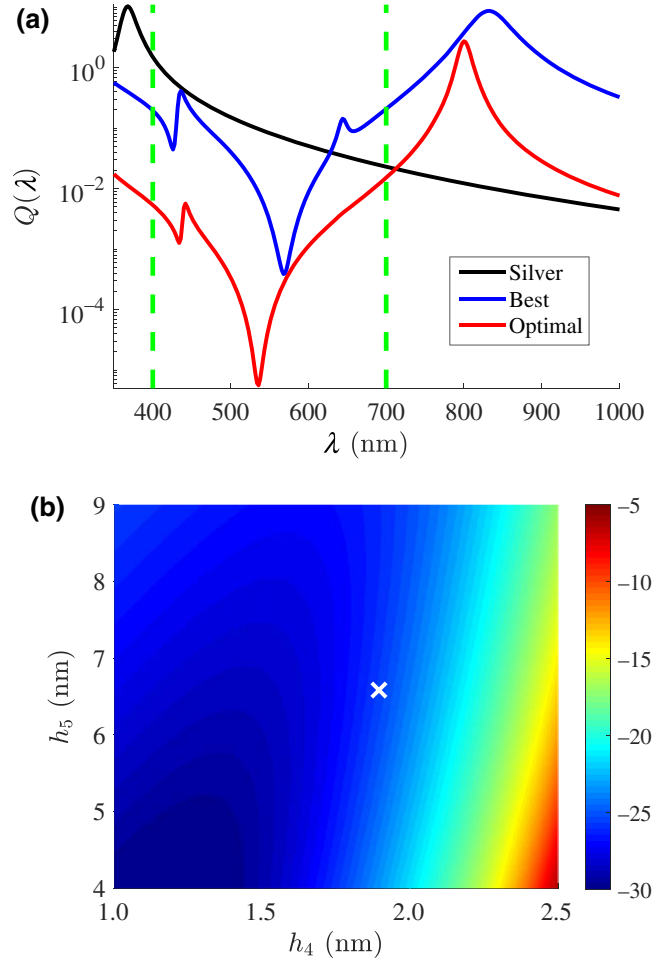


FIG. 5. (a) The scattering efficiency  $Q(\lambda)$  as a function of the operational wavelength  $\lambda$  for a silver nanosphere equal in size to our optimal particle (silver), the best-cloaked particle from the training set (best), and the best prediction of our neural network [optimal design of Fig. 4(b)]. Vertical dashed lines denote the visible part of the wavelength spectrum. (b) The performance metric  $\bar{Q}$  (in decibels) of the optimal nanosphere [white cross, shown in Figs. 4(b) and 5(a)] when the physical dimensions of its two external layers ( $h_4, h_5$ ) are perturbed.

compared to silica in the visible, the corresponding layers ( $\rho = 2, 4$ ) are substantially thinner compared to the dielectric ones ( $\rho = 3, 5$ ), while the silica core ( $\rho = 1$ ) is small since it does not interact much with the external illumination. As far as the thicknesses of the layers are concerned, they decrease as one approaches the optimal design (namely the optimal level of  $Q^* < 0$ ) with the exception of the intermediate dielectric mantle; accordingly the proposed nanoparticle gets physically smaller. It should be stressed, however, that the diameter of the cloaked particle, with the tiny  $\bar{Q}$  indicated by Fig. 4(a) is not negligible compared to the operational wavelength band; indeed,  $2b > 60$  nm with  $400$  nm  $< \lambda < 700$  nm.

In Fig. 5(a) we demonstrate the effectiveness of the proposed neural network-based inverse design technique by considering the optimal design given by Fig. 4(b) and represent its scattering efficiency  $Q(\lambda)$  together with two other designs: a silver sphere of the same size and the best cloaked nanoparticle from our dataset. It is apparent that the result of our optimization is superior to all those designs we started with, outperforming them across the entire visible spectrum (denoted by dashed vertical lines). It seems that the quest for cloaked objects at a specific frequency window leads to solutions that scatter substantially outside of this band; such a feature is consistent with the finding that the total scattering (over all wavelengths) of a passive linear particle is an increasing function of its own size [64]. For this reason, and contrary to the frequency response of a silver nanosphere with the same external radius, the deep minimum at the center of the visible band is followed by a sharp peak outside of it, giving the well-known asymmetric lineshape of Fano resonance [65–67].

In Fig. 5(b) we represent the average scattering efficiency  $\bar{Q}$  (expressed in decibels) as a function of the physical thicknesses of the two external layers (silver layer with  $h_4$  and silica layer with  $h_5$ ), which are mostly exposed to the external illumination. It is clear that the design's response is more sensitive to changes in the plasmonic layer's thickness compared to the size of the dielectric shell. One may also directly infer that in the parametric vicinity of the given optimal designs, there are particles with even better cloaking performances (lower  $\bar{Q}$ ) and thus the predictions from the neural network can serve well as initial points for further optimization.

## V. CONCLUSIONS

One challenging objective in photonic design is to render an object, of non-negligible optical size, invisible for all colors of the visible spectrum, with extensive applications in sensing, wave steering, and stealth technologies. We demonstrate that this challenge can be tackled by using artificial neural networks that optimize nanospheres composed of multiple concentric shells. The weights for its

nodes are found through gradient descent optimization and back propagation after excluding from the training set the strongly scattering particles; after the training, the network learns to approximate the optical response for such a class of scatterers. Most importantly, we find that the inverse design is much more successful if one swaps the position of inputs and outputs and performs the deep learning process by feeding the network with scattering responses that correspond to certain particle configurations and not the opposite. This modification in the architecture allows the neural network to provide an answer to any inverse problem of this type in a forward manner and without iterations.

The proposed technique can be generalized to treat alternative purposes if the training set is altered accordingly. Moreover, the performed mirror flip can be refined by successive implementations of the inverse process with similar targets in order to provide reliable and unique predictions. Finally, our approach shows great potential in inversely designing complex photonic nanostructures with very large parameter landscapes, which cannot be treated with more conventional optimization approaches.

## ACKNOWLEDGMENTS

This work was partially supported by Nazarbayev University under Grants No. 090118FD5349 and No. SST2017031. Funding from MES RK state-targeted program under Grant No. BR05236454 is also acknowledged. The corresponding author (C.V.) would like to thank Professor Alexandros Dimakis (University of Texas at Austin, TX, USA) and Professor Atakan Varol (Nazarbayev University, Kazakhstan) for useful feedback.

- 
- [1] O. G. Selfridge, *Pandemonium: in Proc. of the Symp. on Mech. of Thought Proc.* (National Physical Laboratory, London, UK, 1958), p. 513.
  - [2] F. Rosenblatt, *The Perceptron – A Perceiving and Recognizing Automaton*, Technical Report 85-460-1, Cornell Aeronautical Laboratory, Ithaca, New York, USA (1957).
  - [3] P. H. Winston, PhD Dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, 1970.
  - [4] G. Cybenko, Approximation by superpositions of a sigmoidal function, *Math. Control Signals Syst.* **2**, 303 (1989).
  - [5] K. Hornik, M. Stinchcombe, and H. White, Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks, *Neural Netw.* **3**, 551 (1990).
  - [6] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, *Nature* **323**, 533 (1986).
  - [7] D. F. Specht, A general regression neural network, *IEEE T. Neural Netw.* **2**, 568 (1991).



- [8] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE* **86**, 2278 (1998).
- [9] I. Goodfellow and Y. Bengio and A. Courville, *Deep Learning* (MIT Press, Cambridge, Massachusetts, USA, 2016). Online available at: <http://www.deeplearningbook.org>.
- [10] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *Nature* **521**, 436 (2015).
- [11] Air Force Research Laboratory (AFRL) Announcement. Online available at: <https://community.afml.org/wg/afosr/m/afosr/240890/download>.
- [12] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition, *IEEE Signal Process. Mag.* **29**, 82 (2012).
- [13] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, Learning hierarchical features for scene labeling, *IEEE Trans. Pattern. Anal. Mach. Intell.* **35**, 1915 (2013).
- [14] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Machine learning and the physical sciences, *Rev. Mod. Phys.* **91**, 045002 (2019).
- [15] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, A survey on deep learning in medical image analysis, *Med. Image Anal.* **42**, 60 (2017).
- [16] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. C. Courville, Y. Bengio, C. J. Pal, P.-M. Jodoin, and H. Larochelle, Brain tumor segmentation with deep neural networks, *Med. Image Anal.* **35**, 18 (2017).
- [17] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, Deep learning for computational biology, *Mol. Syst. Biol.* **12**, 878 (2016).
- [18] E. Gawehn, J. A. Hiss, and G. Schneider, Deep learning in drug discovery, *Mol. Inform.* **35**, 3 (2016).
- [19] J. Carrasquilla and R. Melko, Machine learning phases of matter, *Nat. Phys.* **13**, 431 (2017).
- [20] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural network, *Science* **355**, 602 (2017).
- [21] G. D. Barmparis, G. Neofotistos, M. Mattheakis, J. Hizanidis, G. P. Tsironis, and E. Kaxiras, Robust prediction of complex spatiotemporal states through machine learning with sparse sensing, *Phys. Lett. A* **384**, 126300 (2020).
- [22] J. Wang, S. Paesani, R. Santagati, S. Knauer, A. A. Gentile, N. Wiebe, M. Petruzzella, J. L. O'Brien, J. G. Rarity, A. Laing, and M. G. Thompson, Experimental quantum Hamiltonian learning, *Nat. Phys.* **13**, 551 (2017).
- [23] B. G. Buchanan, Artificial intelligence in finance, Zenodo (2019).
- [24] S. Mullainathan and J. Spiess, Machine learning: An applied econometric approach, *J. Econ. Perspect.* **31**, 87 (2017).
- [25] A. D. Tranter, H. J. Slatyer, M. R. Hush, A. C. Leung, J. L. Everett, K. V. Paul, P. Vernaz-Gris, P. K. Lam, B. C. Buchler, and G. T. Campbell, Multiparameter optimisation of a magneto-optical trap using deep learning, *Nat. Commun.* **9**, 4360 (2018).
- [26] R. Trivedi, L. Su, J. Lu, M. F. Schubert, and J. Vucković, Data-driven acceleration of photonic simulations, *Sci. Rep.* **9**, 19728 (2019).
- [27] D. Liu, Y. Tan, E. Khoram, and Z. Yu, Training deep neural networks for the inverse design of nanophotonic structures, *ACS Photonics* **5**, 1365 (2018).
- [28] W. Ma, F. Cheng, Y. Xu, Q. Wen, and Y. Liu, Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy, *Adv. Mater.* **31**, 1901111 (2019).
- [29] W. Ma, F. Cheng, and Y. Liu, Deep-learning-enabled on-demand design of chiral metamaterials, *ACS Nano* **12**, 6326 (2018).
- [30] Y. Kiarashinejad, M. Zandehshahvar, S. Abdollahramezani, O. Hemmatyar, R. Pourabolghasem, and A. Adibi, Knowledge discovery in nanophotonics using geometric deep learning, *Adv. Intell. Syst.* **2**, 1900132 (2020).
- [31] I. Malkiel, M. Mrejen, A. Nagler, U. Arieli, L. Wolf, and H. Suchowski, Plasmonic nanostructure design and characterization via deep learning, *Light: Sci. Appl.* **7**, 60 (2018).
- [32] S. So, J. Mun, and J. Rho, Simultaneous inverse design of materials and structures via deep learning: Demonstration of dipole resonance engineering using core-shell nanoparticles, *ACS Appl. Mater. Interfaces* **11**, 24264 (2019).
- [33] Nanophotonic particle simulation and inverse design using artificial neural networks, *Sci. Adv.* **4**, eaar4206 (2018).
- [34] Y. Qu, L. Jing, Y. Shen, M. Qiu, and M. Soljačić, Migrating knowledge between physical scenarios based on artificial neural networks, *ACS Photonics* **6**, 1168 (2019).
- [35] U. Leonhardt, Optical conformal mapping, *Science* **312**, 1777 (2006).
- [36] J. B. Pendry, D. Schurig, and D. R. Smith, Controlling electromagnetic fields, *Science* **312**, 1780 (2006).
- [37] W. Cai, U. K. Chettiar, A. V. Kildishev, and V. M. Shalaev, Optical cloaking with metamaterials, *Nat. Photonics* **1**, 224 (2007).
- [38] R. Liu, C. Ji, J. J. Mock, J. Y. Chin, T. J. Cui, and D. R. Smith, Broadband ground-plane cloak, *Science* **323**, 366 (2009).
- [39] A. Alù and N. Engheta, Achieving transparency with plasmonic and metamaterial coatings, *Phys. Rev. E* **72**, 016623 (2005).
- [40] C. A. Valagiannopoulos, P. Alitalo, and S. A. Tretyakov, On the minimal scattering response of PEC cylinders in a dielectric cloak, *IEEE Antennas Wirel. Propag. Lett.* **13**, 403 (2014).
- [41] P. Alitalo and S. A. Tretyakov, Broadband electromagnetic cloaking realized with transmission-line and waveguiding structures, *Proc. IEEE* **99**, 1646 (2011).
- [42] C. A. Valagiannopoulos and P. Alitalo, Electromagnetic cloaking of cylindrical objects by multilayer or uniform dielectric claddings, *Phys. Rev. B* **85**, 115402 (2012).
- [43] J. Li and J. B. Pendry, Hiding under the Carpet: A New Strategy for Cloaking, *Phys. Rev. Lett.* **101**, 203901 (2008).
- [44] C. A. Valagiannopoulos, N. L. Tsitsas, and A. H. Sihvola, Hiding a bump on a PEC plane by using an isotropic lossless dielectric layer, *IEEE Trans. Antennas Propag.* **62**, 5706 (2014).

- [45] D. L. Sounas, R. Fleury, and A. Alù, Unidirectional Cloaking Based on Metasurfaces with Balanced Loss and Gain, *Phys. Rev. Appl.* **4**, 014005 (2015).
- [46] A. Chen and F. Monticone, Active scattering-cancellation cloaking: Broadband invisibility and stability constraints, *IEEE Trans. Antennas Propag.* **68**, 1655 (2020).
- [47] M. Selvanayagam and G. V. Eleftheriades, Experimental Demonstration of Active Electromagnetic Cloaking, *Phys. Rev. X* **3**, 041011 (2013).
- [48] S. Kim, P. Lu, S. Mukherjee, M. Gilbert, L. Jing, V. Čeperić, and M. Soljačić, Integration of neural network-based symbolic regression in deep learning for scientific discovery, arXiv:1912.04825v1, (2019).
- [49] D. R. Wilson and T. R. Martinez, The general inefficiency of batch training for gradient descent learning, *Neural Netw.* **16**, 1429 (2003).
- [50] R. Ge, F. Huang, C. Jin, and Y. Yuan, Escaping from saddle points – online stochastic gradient for tensor decomposition, *J. Mach. Learn. Res.: Workshop Conf. Proc.* **40**, 1 (2015).
- [51] S. Molesky, Z. Lin, A. Y. Piggott, W. Jin, J. Vucković, and A. W. Rodriguez, Inverse design in nanophotonics, *Nat. Photonics* **12**, 659 (2018).
- [52] K. M. McPeak, S. V. Jayanti, S. J. P. Kress, S. Meyer, S. Iotti, A. Rossinelli, and D. J. Norris, Plasmonic films can easily be better: Rules and recipes, *ACS Photonics* **2**, 326 (2015).
- [53] J. Khurgin, W.-Y. Tsai, D. P. Tsai, and G. Sun, Landau damping and limit to field confinement and enhancement in plasmonic dimers, *ACS Photonics* **4**, 2871 (2017).
- [54] C. A. Valagiannopoulos and N. L. Tsitsas, Linearization of the T-matrix solution for quasi-homogeneous scatterers, *J. Opt. Soc. Am. A* **26**, 870 (2009).
- [55] C. A. Valagiannopoulos, Single-series solution to the radiation of loop antenna in the presence of a conducting sphere, *Prog. Electromagn. Res.* **72**, 277 (2007).
- [56] C. A. Valagiannopoulos, An overview of the Watson transformation presented through a simple example, *Prog. Electromagn. Res.* **75**, 137 (2007).
- [57] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, *J. Artif. Intell.* **16**, 321 (2002).
- [58] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshein, 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, PyTorch: An imperative style, high-performance deep learning library, *NeurIPS* **32**, 8026 (2019).
- [59] D. P. Kingma and J. Ba, Adam: a method for stochastic optimization, arXiv:1412.6980v9, (2017).
- [60] Facebook Artificial Intelligence Website, <https://ai.facebook.com/>.
- [61] PyTorch Official Website, <https://pytorch.org/>.
- [62] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, *Proc. ICML, PMLR* **37**, 448 (2015).
- [63] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* **15**, 1929 (2014).
- [64] F. Monticone and A. Alù, Do Cloaked Objects Really Scatter Less? *Phys. Rev. X* **3**, 041005 (2013).
- [65] A. E. Miroshnichenko, S. Flach, and Y. S. Kivshar, Fano resonances in nanoscale structures, *Rev. Mod. Phys.* **82**, 2257 (2010).
- [66] A. Sheverdin and C. Valagiannopoulos, Core-shell nanospheres under visible light: Optimal absorption, scattering, and cloaking, *Phys. Rev. B* **99**, 075305 (2019).
- [67] F. Monticone and C. Argyropoulos and A. Alù, Multilayered Plasmonic Covers for Comblike Scattering Response and Optical Tagging, *Phys. Rev. Lett.* **110**, 113901 (2013).