Learning hidden elasticity with deep neural networks

Chun-Teh Chen[a] and Grace X. Gu[b,1]

[a]Department of Materials Science and Engineering, University of California, Berkeley, CA 94720; and [b]Department of Mechanical Engineering, University of California, Berkeley, CA 94720

Edited by David A. Weitz, Harvard University, Cambridge, MA, and approved June 28, 2021 (received for review February 9, 2021)

Elastography is an imaging technique to reconstruct elasticity distributions of heterogeneous objects. Since cancerous tissues are stiffer than healthy ones, for decades, elastography has been applied to medical imaging for noninvasive cancer diagnosis. Although the conventional strain-based elastography has been deployed on ultrasound diagnostic-imaging devices, the results are prone to inaccuracies. Model-based elastography, which reconstructs elasticity distributions by solving an inverse problem in elasticity, may provide more accurate results but is often unreliable in practice due to the ill-posed nature of the inverse problem. We introduce ElastNet, a de novo elastography method combining the theory of elasticity with a deep-learning approach. With prior knowledge from the laws of physics, ElastNet can escape the performance ceiling imposed by labeled data. ElastNet uses backpropagation to learn the hidden elasticity of objects, resulting in rapid and accurate predictions. We show that ElastNet is robust when dealing with noisy or missing measurements. Moreover, it can learn probable elasticity distributions for areas even without measurements and generate elasticity images of arbitrary resolution. When both strain and elasticity distributions are given, the hidden physics in elasticity—the conditions for equilibrium—can be learned by ElastNet.

deep learning | neural networks | elastography | elasticity theory | inverse problems

Being able to reconstruct mechanical property distributions of objects noninvasively has a broad range of applications in materials science (1), civil engineering (2), biomedical engineering (3), and clinical diagnosis (4, 5). As various diseases progress, the stiffness (elasticity) of cells, tissues, and organs are often altered (6). Palpation, a self-screening procedure for tumors, utilizes the difference in elasticity between healthy and cancerous tissues to distinguish them. The displacement (or strain) distribution of an elastic body under deformation can be acquired by a variety of imaging techniques (e.g., ultrasound, MRI, digital image correlation) (1, 3–5). If the stress distribution of the body is also known, the elasticity distribution can be calculated by the elastic constitutive relation (Hooke's law). However, there is currently no technique that can measure the stress distribution of a body in vivo. Therefore, in the applications of elasticity imagining (elastography), the stress distribution of a body is commonly assumed to be uniform, and the measured strain distribution is interpreted as a relative elasticity distribution. This approach is referred to as strain-based elastography and has the advantage of being easy to implement (7). The assumption of stress uniformity, however, is inaccurate. The stress field of a body can be distorted significantly near a hole, inclusion, or wherever the elasticity varies. This phenomenon, known as stress concentration, is of great interest in industry and academia (8–10). To mitigate this misinterpretation, a research field focusing on solving an inverse problem associated with elastography has been extensively investigated for decades (11–13). In this approach, referred to as model-based elastography, the elasticity distribution of a body can be, in principle, reconstructed by modeling the elastic behavior of the body and solving the inverse problem in elasticity.

To tackle this inverse problem, the predominant approaches in the literature are based on minimizing the difference between the measured and simulated displacements (7, 14–17). These approaches are computationally expensive as they need many iterations—each iteration requires solving a forward problem (e.g., using the finite element analysis) and conducting sensitivity analysis in order to update the prediction. It is possible to solve this inverse problem directly. In other approaches, known as direct approaches, measurements are considered as coefficients in the partial differential equations (PDEs) for equilibrium ($\nabla\boldsymbol{\sigma} = 0$) (18–20). While the direct approaches are computationally more efficient than the iterative approaches, they often perform poorly when the measurements contain large strain gradients or noise or when the elasticity distribution is not smooth (not differentiable). Moreover, the error from noise tends to propagate along the integration path when solving the PDEs, thus producing low-quality results (18). Due to these limitations, model-based elastography methods were mostly applied to simple elastography problems such as a uniform body containing a few circular inclusions.

Inverse problems arise in many scientific and engineering fields and are difficult to solve using conventional approaches (21–23). In recent years, much progress toward artificial intelligence and machine learning (ML) has been made and provided novel directions to solve inverse problems. For instance, ML techniques were applied to inverse problems in materials design (24–26), fluid mechanics (27, 28), and many others (29–34). To obtain useful information from an elasticity image (e.g., to identify potential tumors), the number of pixels (resolution) is typically on the order of 10^3 to 10^5. Due to the high-dimensional input space (strain distribution) and output space (elasticity distribution), the hidden correlation between them is difficult to capture by conventional supervised learning using labeled data. In principle, supervised learning may still work for this inverse problem if the number of possible elasticity distributions can be largely reduced. A simple way to do so, for instance, is to consider a uniform body containing

Significance

Elastography has been applied to medical imaging to detect cancerous tissues for decades. The conventional elastography based on the assumption of stress uniformity is prone to inaccuracies. Accurately reconstructing elasticity distributions of heterogeneous objects requires solving an inverse problem in elasticity, which is difficult due to its ill-posed nature. By implementing machine learning techniques, we have developed ElastNet, an elastography method for extracting the hidden elasticity of objects based on measured strain distributions. We show that by combining the theory of elasticity with a deep-learning approach, ElastNet can provide rapid and accurate predictions. We validate ElastNet by applying it to various elastography problems with complex elasticity distributions.

Author contributions: C.-T.C. and G.X.G. designed research; C.-T.C. performed research; C.-T.C. analyzed data; and C.-T.C. and G.X.G. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: ggu@berkeley.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2102721118/-/DCSupplemental.

Published July 29, 2021.

a few inclusions and constrain the possible shapes, sizes, locations, and elastic moduli of these inclusions (35). However, when an ML model is trained in this manner, adding such artificial constraints limits the application of the model in practice.

Results

Deep Learning for Elastography. We introduce ElastNet, a de novo elastography method to learn the hidden elasticity of a body with a deep neural network (DNN). The framework of ElastNet is summarized in Fig. 1. The DNN f_θ with learning parameters θ is not trained from labeled data—it is never shown the correct elasticity distribution. By contrast, it is supervised by the theory of elasticity, thus allowing it to escape the performance ceiling imposed by labeled data. The elastic constitutive relation ($\boldsymbol{\sigma} = \boldsymbol{C\varepsilon}$) and equilibrium equation ($\nabla\boldsymbol{\sigma} = 0$) for solving the inverse problem in elasticity are encoded in the DNN as prior knowledge. Biological tissues are mainly composed of water, and they are nearly incompressible (4). Here, all material points in a body of interest are assumed to be linear, isotropic, and incompressible. The elasticity at a material point can be described by either Young's modulus E or shear modulus G. Without loss of generality, Young's modulus E is used to quantify the elasticity in this work. The body is assumed to be a thin plate, and thus the nonzero stress components are σ_{xx}, σ_{yy}, and τ_{xy} (plane stress state). The information of position p and strain ε at all material points is converted to a data set. The stress σ at each material point is calculated by the encoded elastic constitutive relation based on the measured strain ε and predicted elasticity E (see *Materials and Methods*). The DNN f_θ takes the information of position p at each material point as an input and outputs its elasticity, $E = f_\theta(p)$. The information of strain ε at each material point is not involved in the forward-propagation for predicting the elasticity but is used in the backpropagation for updating the weights θ. When the entire data set is passed forward through the DNN, a predicted stress distribution $\boldsymbol{\sigma}$ is generated.

A predicted stress distribution $\boldsymbol{\sigma}$ includes three stress images ($\sigma_{xx}, \sigma_{yy}, \tau_{xy}$). Before training, these strain images are unlikely to satisfy the conditions for equilibrium, as the initial elasticity distribution $\boldsymbol{E}$ is generated by the DNN with random weights θ. To evaluate how close the predicted stress distribution $\boldsymbol{\sigma}$ is from equilibrium, the stress images are passed forward through a convolutional layer in the DNN. Unlike other convolutional neural networks, in which the kernels need to be learned from labeled data, the kernels in our convolutional layer are encoded in such a way that the convolution operation is used to evaluate the conditions for equilibrium (explained in *Learning the Conditions for Equilibrium*). Unbalanced (residual) force maps are then generated by the convolution operation. The training procedure minimizes the norms of the residual forces with an additional physical constraint (to be discussed) and updates the weights θ using backpropagation. Consequently, the predicted elasticity distribution $\boldsymbol{E}$ is updated and then used in the next iteration of training. This training procedure repeats until the predicted elasticity distribution $\boldsymbol{E}$ is converged.

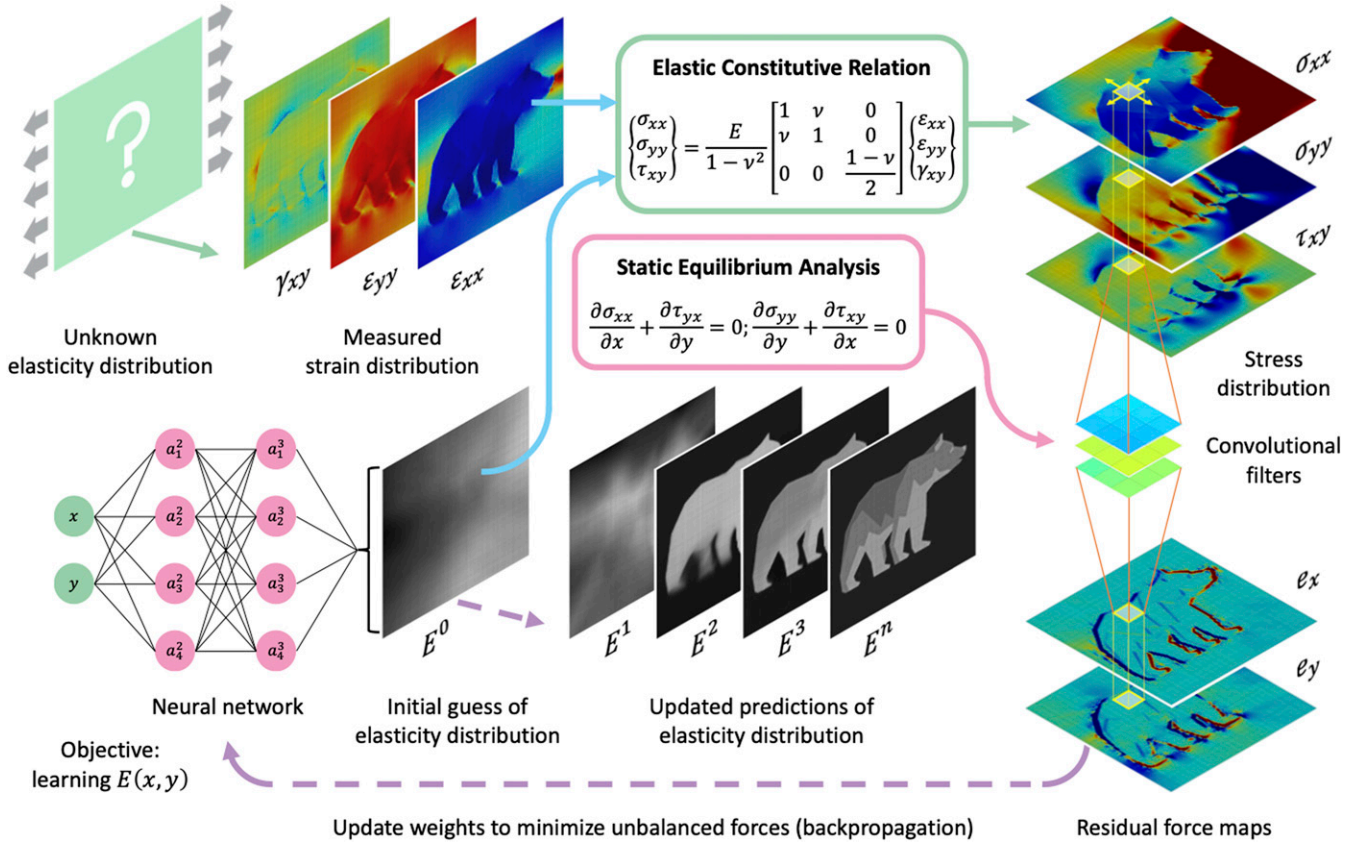

Fig. 1. Framework of ElastNet. The DNN f_θ takes the information of position p at each material point as an input and outputs its elasticity, $E = f_\theta(p)$. The stress σ at each material point is calculated by the encoded elastic constitutive relation ($\boldsymbol{\sigma} = \boldsymbol{C\varepsilon}$) based on the measured strain ε and predicted elasticity E. When the entire data set is passed forward through the DNN, a predicted stress distribution $\boldsymbol{\sigma}$ is generated. Residual force maps are then generated by the convolution operation. The training procedure minimizes the norms of the residual forces with an additional physical constraint and updates the weights θ using backpropagation. Consequently, the predicted elasticity distribution $\boldsymbol{E}$ is updated and then used in the next iteration of training. This training procedure repeats until the predicted elasticity distribution $\boldsymbol{E}$ is converged.

Learning the Conditions for Equilibrium. The deep-learning framework in ElastNet is similar in spirit to the so-called physics-informed deep learning (28, 29, 31), in which physical laws are encoded into the loss function. Most physics-informed models used automatic differentiation (36) to solve the PDEs in physics. However, automatic differentiation may not be appropriate for the inverse problem in this work. As mentioned, measured strains naturally contain noise, and differentiating the strains can amplify the noise significantly. Moreover, this approach requires the elasticity distribution of a body to be differentiable, which is often not true in practice. Instead of using automatic differentiation, ElastNet uses a convolution operation to solve the PDEs for equilibrium. To elaborate on the idea, we consider a small cube with sides of length h. The equilibrium conditions for the cube can be expressed in finite difference forms as (see *Materials and Methods*):

$$\sigma_{xx}(x+h,y)-\sigma_{xx}(x,y)+\tau_{yx}(x,y+h)-\tau_{yx}(x,y)=0, \quad [1]$$

$$\sigma_{yy}(x,y+h)-\sigma_{yy}(x,y)+\tau_{xy}(x+h,y)-\tau_{xy}(x,y)=0.$$

We assume that the cube is made up of 3×3 material points as shown in Fig. 2*A*. Eq. **1** can be expressed in terms of the stress components at these material points:

$$\sum_{a=1}^{3}\sum_{b=1}^{3} w_{xx}(a,b)\sigma_{xx}(a,b)+w_{yy}(a,b)\sigma_{yy}(a,b)+w_{xy}(a,b)\tau_{xy}(a,b)=0, \quad [2]$$

where w_{xx}, w_{yy}, and w_{xy} are the convolution kernels for σ_{xx}, σ_{yy}, and τ_{xy}, respectively. By choosing proper values for the kernels, Eq. **2** can be used to describe the equilibrium conditions for the cube. Two sets of kernels are encoded in the DNN to describe the equilibrium conditions in the x-direction and y-direction, respectively (Fig. 2*A*). These kernels may therefore be viewed as "equilibrium detectors". Residual force maps (Fig. 1) can be generated by the convolution operation, in which each residual force is calculated by

$$e(i,j)=\sum_{a=1}^{3}\sum_{b=1}^{3}\left(w_{xx}(a,b)\sigma_{xx}(i+a-1,j+b-1)+w_{yy}(a,b)\sigma_{yy}(i+a-1,j+b-1)+w_{xy}(a,b)\tau_{xy}(i+a-1,j+b-1)\right)ht, \quad [3]$$

where t is the thickness of the cube. Here, we show that the conditions for equilibrium can be encoded in the DNN as domain knowledge for solving the inverse problem. Now, we consider the possibility of learning this domain knowledge from labeled data. To test this idea, an inhomogeneous body is modeled by the finite element method (FEM) with a 128×128 mesh. The elasticity field of the body is defined by a two-dimensional sinusoidal function, $E(x,y)=0.45\left(1+\cos\left(\frac{4\pi x}{L}\right)\cos\left(\frac{4\pi y}{L}\right)\right)+0.1$. The unit is megapascal (MPa). Thus, the maximum and minimum modulus values are 1 MPa and 0.1 MPa, respectively. This "sinusoidal" model is subjected to externally applied displacements along the x-direction on the boundary. An average normal strain (ε_{xx}) of 1% is introduced by the applied displacements. The details of the finite element analysis can be found in our previous work (26) and are summarized in *Materials and Methods*.

The elasticity and strain distributions of the sinusoidal model are shown in Fig. 2*B*. To learn the conditions for equilibrium, the elasticity and strain distributions are both fed into the DNN, and the loss function is defined as the mean absolute error (MAE) of the residual forces. From Eq. **3**, it can be seen that the kernels (w_{xx}, w_{yy}, and w_{xy}) cannot be uniquely determined by minimizing the norms of the residual forces. A trivial solution is to set all of the kernel values to zeros. Thus, to obtain physically meaningful kernels, additional information must be given. For instance, when the kernel for τ_{xy} to describe the equilibrium condition in the x-direction is given, the other two kernels, which are for σ_{xx} and σ_{yy}, can be learned. Similarly, when the kernel for τ_{xy} to describe the equilibrium condition in the y-direction is given, the other two kernels can be learned. The kernels learned by the DNN are shown in Fig. 2*B* and are almost identical to those derived mathematically shown in Fig. 2*A*. The results show that the kernels encoded in the DNN to describe the conditions for equilibrium are correct, as the same kernels can be learned from the hidden correlation between the elasticity and strain distributions by the DNN.

Learning Hidden Elasticity. A strain distribution alone does not provide sufficient information to generate a unique elasticity distribution; an additional physical constraint must be imposed (12, 19). We first investigate the effect of physical constraints on prediction accuracy (*SI Appendix*, Fig. S1 and Movies S1 and S2). We demonstrate that ElastNet can generate accurate predictions when imposing a proper physical constraint based on either the total applied force on the boundary or the mean elasticity. In practice, measured strains naturally contain a certain amount of noise. We also investigate the effect of noise in measurements on prediction accuracy (*SI Appendix*, Fig. S2 and Movies S3 and S4). We show that ElastNet is robust when dealing with noisy measurements. To evaluate the performance of ElastNet, we compare it with OpenQSEI (17), an iterative model-based elastography method using FEM. We show that ElastNet generates more accurate predictions and is computationally more efficient compared with OpenQSEI (*SI Appendix*, Figs. S3–S6 and Tables S1–S3).

Most elastography problems assume that measurements are known everywhere in a body. Here, we consider a more challenging problem in which some of the measurements in a body are missing. Noisy measurements may still provide useful information for learning the hidden elasticity. However, missing measurements not only provide no information for solving the inverse problem but also may cause the calculation to break down. Here, we consider a uniform body containing a soft inclusion with a shape similar to the University of California, Berkeley "Cal" logo. In this "Cal" model, the modulus values of the body and soft inclusion are 1 MPa and 0.1 MPa, respectively. The elasticity and strain distributions of the model are shown in Fig. 3*A*. In the first scenario, we assume that there is no missing data in the measured strains. In the second scenario, we assume that the measured strains in an arbitrary area are missing (set to zeros). The predicted elasticity distributions and relative error maps (compared with the ground-truth elasticity distribution) are shown in Fig. 3*B*, and the error over the training epochs is shown in Fig. 3*C*. In the first scenario, the predicted elasticity distribution is accurate with a mean relative error (MRE) of 3.01%. Larger errors occur on the boundary between the body and soft inclusion due to large elasticity differences. In the second scenario, the predicted elasticity distribution is still accurate with an MRE of 6.97%, given that the measurements on a squared area (corresponds to 6.25% of the total area) are missing (yellow-boxed area in Fig. 3*B*). Larger errors occur on the boundary between the body and soft inclusion and also in the area without data. If the area without data is excluded, the MRE is reduced to 2.95%, which is almost the same as the MRE observed in the first scenario with the full data (3.01%). The results show that, for ElastNet, missing measurements only reduce the prediction accuracy in the area without data but do not affect the prediction accuracy in the other areas. Moreover, the DNN is trained to learn the hidden elasticity as a function of positions (Fig. 1). Once such a function is learned, ElastNet can predict a probable elasticity distribution for the area even without measurements (inner figure of Fig. 3*C*). The learning processes for the first and second scenarios are shown in Movies

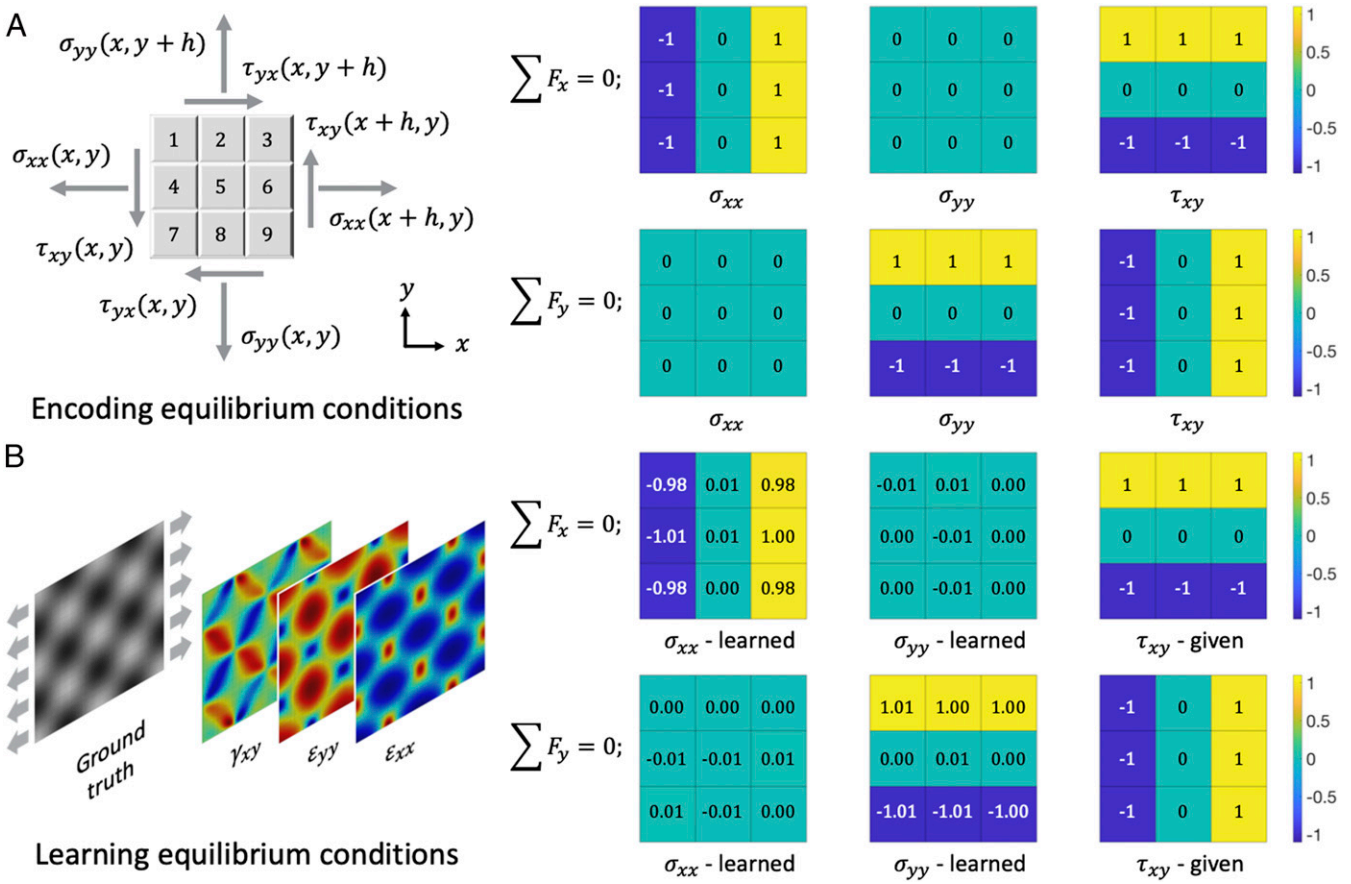

Fig. 2. Encoding and learning equilibrium conditions. (*A*) A representative cube consists of 3×3 material points. The equilibrium conditions for the cube can be expressed in terms of the stress components at these material points. Two sets of kernels are encoded in the DNN to describe the equilibrium conditions in the *x*-direction (upper three kernels) and *y*-direction (lower three kernels), respectively. (*B*) The sinusoidal model is subjected to externally applied displacements along the *x*-direction on the boundary. An average normal strain (ε_{xx}) of 1% is introduced by the applied displacements. To learn the conditions for equilibrium, the elasticity and strain distributions are both fed into the DNN, and the loss function is defined as the MAE of the residual forces. The kernels learned by the DNN are almost identical to those derived mathematically shown in *A*.

S5 and S6, respectively. Lastly, we compare ElastNet with DeepFill (37, 38), one of the state-of-the-art image inpainting methods. DeepFill is based on a variant of generative adversarial networks (39), named SN-PatchGAN, with gated convolution trained with millions of images. We use DeepFill to fill missing pixels of predicted elasticity images generated by ElastNet and show that the results obtained from both methods are of equivalent quality (*SI Appendix*, Figs. S7 and S8 and Table S4).

Superresolution Elasticity Imaging. For most elastography methods, the resolution of predicted elasticity distributions (elasticity images) is limited by the resolution of measurements. ElastNet can generate elasticity images of arbitrary resolution. Here, we apply ElastNet to generate a high-resolution elasticity image from low-resolution measurements. Inspired by the beauty and complexity of *Mona Lisa* by Leonardo da Vinci, we consider an inhomogeneous body with an elasticity distribution similar to the painting. In this "Mona Lisa" model, the maximum and minimum modulus values are 1 MPa and 0.1 MPa, respectively. The elasticity and strain distributions of the model are shown in Fig. 4*A*. The predicted elasticity distribution and relative error map are shown in Fig. 4*B*, and the error over the training epochs is shown in Fig. 4*C*. While the Mona Lisa model has an extremely complex elasticity distribution, the prediction accuracy is high with an MRE of 2.73%. No visible difference between the ground truth and prediction can be seen. To understand how ElastNet learns the hidden elasticity, intermediate predictions generated during the learning process are shown in the inner figures of Fig. 4*C*. Interestingly, it can be seen that ElastNet draws an outline first and then adds more details gradually. This process is similar to how an artist draws. The Mona Lisa model is discretized with a 128 × 128 mesh. Thus, the resolution of the measured strains is 128 × 128 (Fig. 4*A*). After learning the hidden elasticity from the measured strains, ElastNet can generate elasticity images of arbitrary resolution. Here, ElastNet is applied to generate an elasticity image of a higher resolution—512 × 512. For comparison, a crop of the ground-truth image and that of the superresolution image are shown in Fig. 4*D*. The superresolution image seems realistic and provides more details compared with the ground-truth image.

A conventional deep-learning model is trained on a data set and can be applied to predict an elasticity distribution based on a new strain distribution without retraining the model (35). ElastNet, on the other hand, is not supervised by labeled data, and thus its performance is not limited by the amount, distribution, and accuracy of the data. However, ElastNet needs to be retrained for different elastography problems. Therefore, the computational efficiency of ElastNet is essential for elasticity imaging in practice. To make an accurate prediction for the Mona Lisa model (Fig. 4*B*), ElastNet takes about 80 min for training 800,000 epochs on a workstation using a single graphics processing unit (GPU). After about 3,000 epochs (~20 s) of training, an intermediate prediction with much detail can already be obtained (inner figures of Fig. 4*C*). The learning

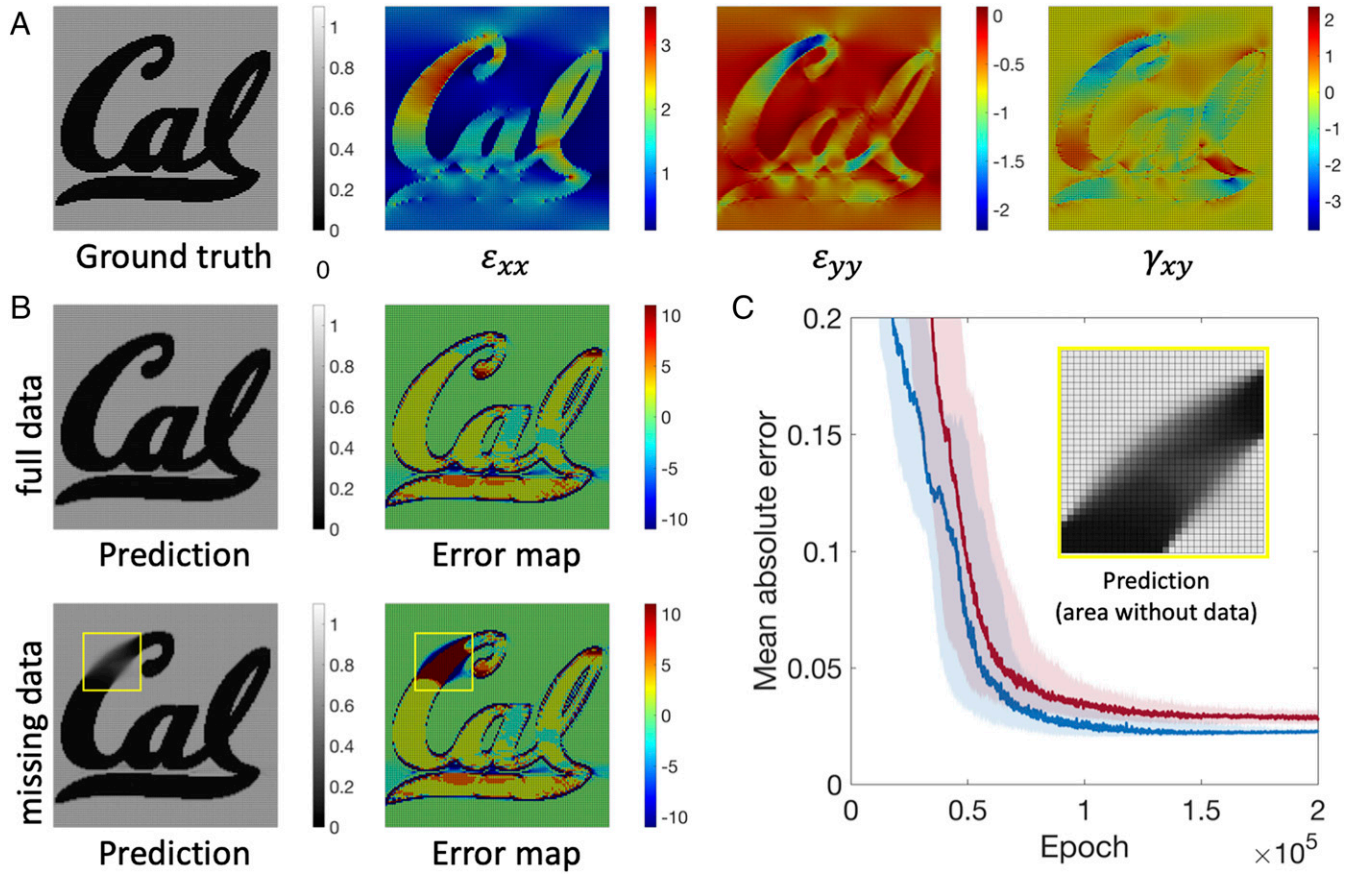

Fig. 3. Effect of missing measurements. (*A*) The ground-truth elasticity distribution (in grayscale) and strain images (in color) of the "Cal" model. The unit in the elasticity distribution is MPa and that in the strain images is percentage (%). (*B*) The predicted elasticity distributions (in grayscale) and relative error maps (in color) for the scenario with the full data and that with missing data. The unit in the error maps is percentage (%). The yellow-boxed area represents the area without data (corresponds to 6.25% of the total area). (*C*) The blue and red shaded lines represent the MAE over the training epochs for the scenario with the full data and that with missing data, respectively. The DNNs are trained 100 times with different initial weights. The line and shading represent the median and interquartile range of the 100 predictions, respectively. The inner figure shows a probable elasticity distribution for the area without data predicted by ElastNet.

process is shown in Movie S7. With further improvements on the DNN architecture, ElastNet may be applied in an environment requiring real-time elasticity imaging with broad implications in various fields. Lastly, we compare ElastNet with ESRGAN (40), which won first place in the 2018 Perceptual Image Restoration and Manipulation Challenge on Perceptual Image Superresolution. We apply ESRGAN to produce a high-resolution version of a predicted elasticity image generated by ElastNet and show that the results obtained from both methods are comparable (*SI Appendix*, Fig. S9 and Table S5).

Discussion

Conventional model-based elastography using FEM represents an elasticity image as a set of pixels. In this work, we consider an elasticity image as a mathematical function. Given a large enough DNN, any complex elasticity image can be approximated by the DNN. This approach ensures that a target elasticity distribution can be represented as a function and allows a learning algorithm to gradually update the function. Here, all material points in a body of interest are assumed to be linear, isotropic, and incompressible. However, ElastNet can be extended to consider compressible materials if necessary. For instance, instead of setting Poisson's ratio at each material point to 0.5, it can be represented as an unknown variable. Thus, the DNN f_θ will take the information of position p at each material point as an input and output its elasticity and Poisson's ratio simultaneously, $(E, \nu) = f_\theta(p)$. We show that, by combining the theory of elasticity with a deep-learning approach, ElastNet can generate rapid and accurate predictions. The prediction accuracy depends on the complexity of the hidden elasticity. A higher prediction accuracy may be expected when the elasticity distribution is simpler or smoother. We also show that ElastNet is robust when dealing with noisy measurements. For measurements with missing data, only the prediction accuracy in the area without data will be compromised and the prediction accuracy in the other areas will not be affected. Once the function of an elastic image is learned, the DNN can predict probable elasticity distributions for areas even without measurements and generate elasticity images of arbitrary resolution. To obtain better performance, incorporating other DNNs specifically trained for image inpainting or single-image superresolution into ElastNet is recommended. With prior knowledge from the theory of elasticity, ElastNet does not require any labeled data for training, and thus no artificial constraint on possible elasticity distributions needs to be imposed. This advantage allows ElastNet to be applied to a broad range of elastography problems in which no prior knowledge is available on the hidden elasticity.

Materials and Methods

Finite Element Analysis. A body of interest is discretized by four-node quadrilateral elements with a 128×128 mesh. The modulus of each element is determined based on the elasticity distribution of the body. To simulate an incompressible material, Poisson's ratio of each element is set to 0.5. The boundary and loading conditions of the body are shown in *SI Appendix*, Fig. S10. The top and bottom boundaries are free. The left boundary is fixed in

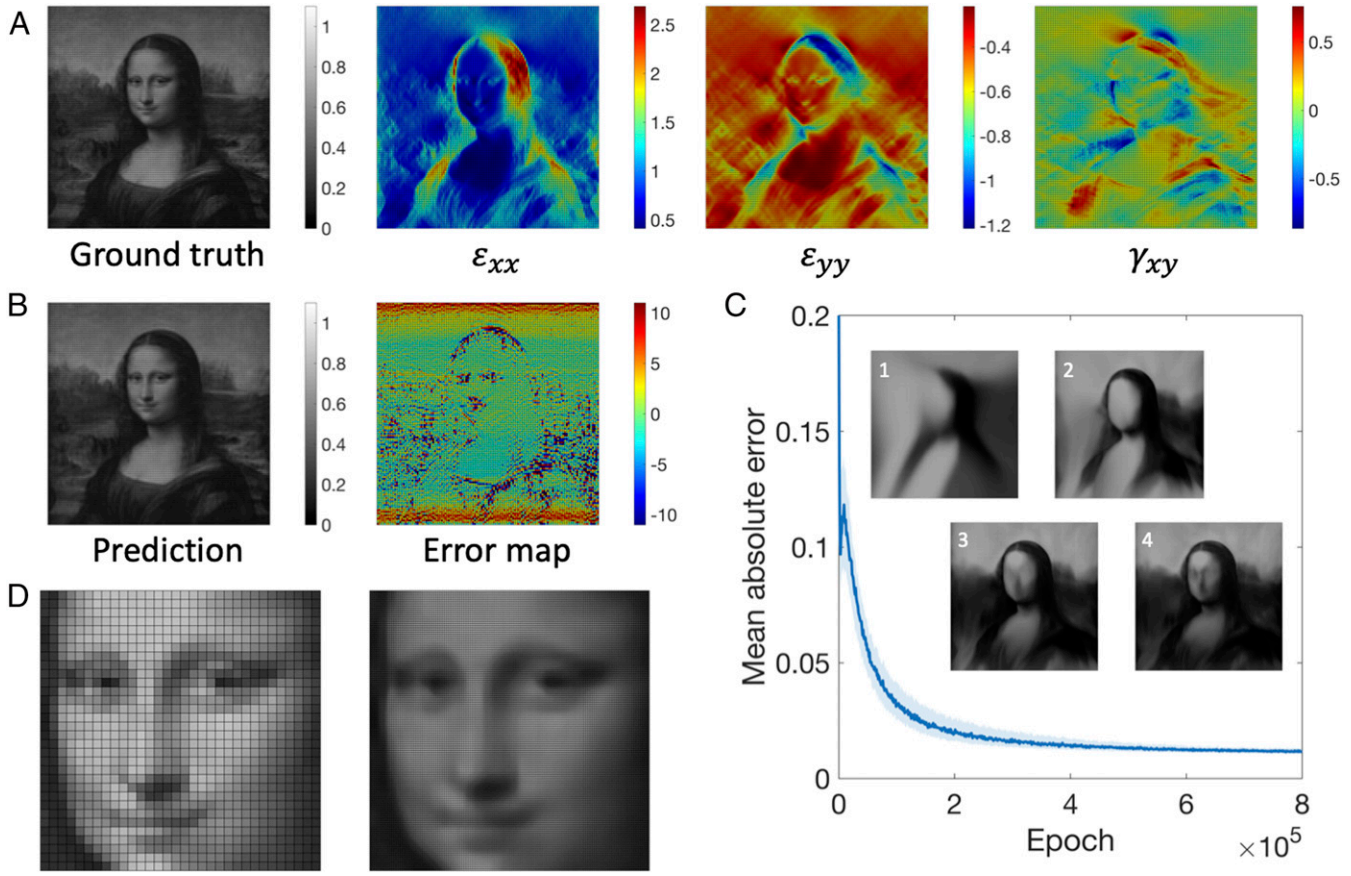

Fig. 4. Superresolution elasticity imaging. (*A*) The ground-truth elasticity distribution (in grayscale) and strain images (in color) of the Mona Lisa model. The unit in the elasticity distribution is MPa and that in the strain images is percentage (%). (*B*) The predicted elasticity distribution (in grayscale) and relative error map (in color). The unit in the error maps is percentage (%). (*C*) The blue shaded line represents the MAE over the training epochs. The DNN is trained 100 times with different initial weights. The line and shading represent the median and interquartile range of the 100 predictions, respectively. The inner figures show the intermediate predictions after 280, 1,200, 2,100, and 3,000 epochs, respectively. (*D*) A crop of the ground-truth image (128×128) and that of the superresolution image (512×512) generated by ElastNet.

such a way that the movements are not allowed along the horizontal direction (*x*-direction) but are allowed along the vertical direction (*y*-direction). To obtain a nonzero strain distribution of the body, external displacements along the *x*-direction are applied on the right boundary. The applied displacements are 1% of the body length. Thus, an average normal strain along the *x*-direction (ε_{xx}) of 1% can be generated. The strain distribution is calculated from the displacement distribution.

Encoded Domain Knowledge of Elasticity. For a two-dimensional body, the relation between the strain and displacement with respect to the Cartesian axes is given by

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial v}{\partial y} \\ \dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x} \end{Bmatrix}, \quad [4]$$

where $\boldsymbol{\varepsilon}$ is the strain vector and u and v are the horizontal and vertical components of the displacement, respectively. The constitutive elasticity relation for a linear elastic isotropic material in plane stress is given by

$$\boldsymbol{\sigma} = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix}, \quad [5]$$

where $\boldsymbol{\sigma}$ is the stress vector, E is Young's modulus, and ν is Poisson's ratio. The equilibrium conditions are typically written in differential forms:

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} = 0 \quad [6]$$

and

$$\frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} = 0.$$

These PDEs carry the derivatives of the stress field, which are the functions of the derivatives of the strain field and elasticity field. The derivatives of the strain field can be calculated from measured strains. However, measured strains naturally contain noise, and the calculation of the derivatives can amplify the noise significantly. The derivatives of the elasticity field cannot be calculated accurately when the derivatives of the strain field are inaccurate. To mitigate this potential problem, we rewrite the equilibrium conditions in finite difference forms:

$$\frac{\sigma_{xx}(x+\Delta x, y) - \sigma_{xx}(x,y)}{\Delta x} + \frac{\tau_{yx}(x, y+\Delta y) - \tau_{yx}(x,y)}{\Delta y} = 0 \quad [7]$$

$$\frac{\sigma_{yy}(x, y+\Delta y) - \sigma_{yy}(x,y)}{\Delta y} + \frac{\tau_{xy}(x+\Delta x, y) - \tau_{xy}(x,y)}{\Delta x} = 0.$$

From these equations, the equilibrium conditions for a small cube with sides of length h can be expressed in Eq. **1**.

Error Measurements in Predictions. The error over the training epochs (Figs. 3C and 4C and *SI Appendix*, Figs. S1C and S2C) is quantified by the MAE, defined as

$$\mathrm{MAE} = \frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}\mathrm{error}_{\mathrm{absolute}}(i,j), \quad [8]$$

where n is the dimension of the elasticity image in both x-direction and y-direction. $\mathrm{error}_{\mathrm{absolute}}$ is the absolute error of a prediction at each material point, defined as

$$\mathrm{error}_{\mathrm{absolute}}(i,j) = \left|E_{\mathrm{pred}}(i,j) - E_{\mathrm{truth}}(i,j)\right|, \quad [9]$$

where E_{pred} is the predicted elasticity and E_{truth} is the ground-truth elasticity. The MAE can be used to compare the performances of different learning algorithms on the same model. However, the MAE may not be an ideal quantity to compare the accuracies between different models. A larger MAE can be expected when the mean elasticity of the model is larger. Therefore, we use the MRE to compare the accuracies between different models. The MRE is defined as

$$\mathrm{MRE} = \frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}\left|\mathrm{error}_{\mathrm{relative}}(i,j)\right|, \quad [10]$$

where $\mathrm{error}_{\mathrm{relative}}$ (unit: %) is the relative error of a prediction at each material point, defined as

$$\mathrm{error}_{\mathrm{relative}}(i,j) = 100\times\left(E_{\mathrm{pred}}(i,j) - E_{\mathrm{truth}}(i,j)\right)\big/E_{\mathrm{truth}}(i,j). \quad [11]$$

The relative error is used to generate the error maps (Figs. 3*B* and 4*B* and *SI Appendix*, Figs. S1*B*, S2*B*, and S6–S8).

ElastNet Training. The DNN consists of 16 fully connected hidden layers with 128 neurons per layer. The rectified linear unit is adopted as the activation function. The input of the DNN is a vector of two variables (x, y) representing the position p of a material point, and the output is the elasticity E of the point. The stress σ at a material point is calculated by the elastic constitutive relation based on the measured strain ε and predicted elasticity E. Full-batch learning is used when training the DNN. A predicted stress distribution σ is generated when the entire data set is passed forward through the DNN. A convolutional layer consisting of 6 filters of kernel size 3×3 with stride 1 is used to generate residual force maps. In these 6 filters, 3 of them are encoded for evaluating the equilibrium condition in the x-direction and the other 3 are for the y-direction (Fig. 2*A*). The Adam optimizer (41) is adopted to minimize the loss function of the DNN. The loss function consists of two parts: one is from the residual forces and the other is from a physical constraint. The residual forces can be measured by the MAE. However, larger residual forces can be expected in areas with larger elasticity. Therefore, the accuracy (relative error) in areas with smaller elasticity will be compromised when using the MAE to measure the residual forces. To make the relative error maps more uniform, the normalized MAE is adopted to measure the residual forces when training the DNN to learn the hidden elasticity. The normalized MAE is defined as

$$\mathrm{loss}_{\mathrm{force}} = \frac{1}{m^2}\sum_{i}^{m}\sum_{j}^{m}\frac{|e(i,j)|}{\hat{E}_{\mathrm{pred}}(i,j)}, \quad [12]$$

where m is the dimension of the residual force map in both x-direction and y-direction, e is the residual force, and $\hat{E}_{\mathrm{pred}}$ is the sum of the predicted elasticity values in a cube containing 3×3 material points. $\hat{E}_{\mathrm{pred}}$ is calculated as

$$\hat{E}_{\mathrm{pred}}(i,j) = \sum_{a=1}^{3}\sum_{b=1}^{3} w_E(a,b)E_{\mathrm{pred}}(i+a-1, j+b-1), \quad [13]$$

where w_E is a convolution kernel of ones. Two types of physical constraints are considered. One is based on the total applied force on the boundary, in which three force boundary conditions (BCs) can be added to the loss function. The penalty terms for these BCs are defined as

$$\mathrm{loss}_{\mathrm{BC1}} = \sum_{i}^{m}\left|\left(\sum_{j}^{m}\sigma_{xx}(i,j)ht\right) - F\right| \quad [14]$$

$$\mathrm{loss}_{\mathrm{BC2}} = \sum_{i}^{m}\left|\sum_{j}^{m}\tau_{xy}(i,j)ht\right|$$

$$\mathrm{loss}_{\mathrm{BC3}} = \sum_{j}^{m}\left|\sum_{i}^{m}\sigma_{yy}(i,j)ht\right|,$$

where F is the total applied force on the right boundary, which is the same as the total reaction force (along the x-direction) on the left boundary (*SI Appendix*, Fig. S10). These penalty terms can be applied to constrain the distribution of the internal stresses. The equilibrium conditions for these BCs are shown in the inner figure of *SI Appendix*, Fig. S1*C*. The other type of physical constraints is based on the mean elasticity. The penalty term for the mean elasticity is defined as

$$\mathrm{loss}_{\mathrm{elast}} = \left|\frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}E_{\mathrm{pred}}(i,j) - \frac{1}{n^2}\sum_{i}^{n}\sum_{j}^{n}E_{\mathrm{truth}}(i,j)\right|. \quad [15]$$

For the inclusion model (*SI Appendix*, Fig. S1), sinusoidal model (*SI Appendix*, Fig. S2), and the "Cal" model (Fig. 3), the DNN is trained for 200,000 epochs. For the Mona Lisa model (Fig. 4), due to the extremely complex hidden elasticity, the DNN is trained for 800,000 epochs. As the weights in the DNN are randomly initialized before training, the predicted elasticity distribution cannot be exactly the same when training with different initial weights. To better evaluate the performance of ElastNet, the predictions reported in this work are the average values after training the DNN 100 times with different initial weights. The DNN is trained using TensorFlow (42) with a single GPU (NVIDIA Tesla V100 or Titan V).

Data Availability. All study data are included in the article and/or supporting information.

ACKNOWLEDGMENTS. We acknowledge support from 3M and the NVIDIA GPU Seed Grant. This work used the Extreme Science and Engineering Discovery Environment, which is supported by the NSF Grant No. ACI-1548562.

1. J. Blaber, B. Adair, A. Antoniou, Ncorr: Open-source 2D digital image correlation matlab software. *Exp. Mech.* **55**, 1105–1122 (2015).
2. S. Sony, S. Laventure, A. Sadhu, A literature review of next-generation smart sensing technology in structural health monitoring. *Struct. Contr. Health Monit.* **26**, e2321 (2019).
3. B. F. Kennedy, P. Wijesinghe, D. D. Sampson, The emergence of optical elastography in biomedicine. *Nat. Photonics* **11**, 215–221 (2017).
4. J.-L. Gennisson, T. Deffieux, M. Fink, M. Tanter, Ultrasound elastography: Principles and techniques. *Diagn. Interv. Imaging* **94**, 487–495 (2013).
5. K. J. Parker, M. M. Doyley, D. J. Rubens, Imaging the elastic properties of tissue: The 20 year perspective. *Phys. Med. Biol.* **56**, R1–R29 (2011).
6. H. Zhi *et al*., Comparison of ultrasound elastography, mammography, and sonography in the diagnosis of solid breast lesions. *J. Ultrasound Med.* **26**, 807–815 (2007).
7. M. M. Doyley, S. Srinivasan, S. A. Pendergrass, Z. Wu, J. Ophir, Comparative evaluation of strain-based and model-based modulus elastography. *Ultrasound Med. Biol.* **31**, 787–802 (2005).
8. G. X. Gu, C.-T. Chen, D. J. Richmond, M. J. Buehler, Bioinspired hierarchical composite design using machine learning: Simulation, additive manufacturing, and experiment. *Mater. Horiz.* **5**, 939–945 (2018).
9. H. Gao, B. Ji, I. L. Jäger, E. Arzt, P. Fratzl, Materials become insensitive to flaws at nanoscale: Lessons from nature. *Proc. Natl. Acad. Sci. U.S.A.* **100**, 5597–5600 (2003).
10. C.-T. Chen, G. X. Gu, Effect of constituent materials on composite performance: Exploring design strategies via machine learning. *Adv. Theory Simul.* **2**, 1900056 (2019).
11. M. M. Doyley, Model-based elastography: A survey of approaches to the inverse elasticity problem. *Phys. Med. Biol.* **57**, R35–R73 (2012).
12. P. E. Barbone, J. C. Bamber, Quantitative elasticity imaging: What can and cannot be inferred from strain images. *Phys. Med. Biol.* **47**, 2147–2164 (2002).
13. J. F. Greenleaf, M. Fatemi, M. Insana, Selected methods for imaging elastic properties of biological tissues. *Annu. Rev. Biomed. Eng.* **5**, 57–78 (2003).
14. A. A. Oberai, N. H. Gokhale, G. R. Feijóo, Solution of inverse problems in elasticity imaging using the adjoint method. *Inverse Probl.* **19**, 297 (2003).
15. A. A. Oberai, N. H. Gokhale, M. M. Doyley, J. C. Bamber, Evaluation of the adjoint equation based algorithm for elasticity imaging. *Phys. Med. Biol.* **49**, 2955–2974 (2004).
16. M. M. Doyley, P. M. Meaney, J. C. Bamber, Evaluation of an iterative reconstruction method for quantitative elastography. *Phys. Med. Biol.* **45**, 1521–1540 (2000).
17. D. Smyl, S. Bossuyt, D. Liu, OpenQSEI: A MATLAB package for quasi static elasticity imaging. *SoftwareX* **9**, 73–76 (2019).
18. P. E. Barbone, A. A. Oberai, Elastic modulus imaging: Some exact solutions of the compressible elastography inverse problem. *Phys. Med. Biol.* **52**, 1577–1593 (2007).
19. P. E. Barbone *et al*., Adjoint-weighted variational formulation for the direct solution of inverse problems of general linear elasticity with full interior data. *Int. J. Numer. Methods Eng.* **81**, 1713–1736 (2010).

20. O. A. Babaniyi, A. A. Oberai, P. E. Barbone, Direct error in constitutive equation formulation for plane stress inverse elasticity problem. *Comput. Methods Appl. Mech. Eng.* **314**, 3–18 (2017).
21. B. Sanchez-Lengeling, A. Aspuru-Guzik, Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **361**, 360–365 (2018).
22. C.-T. Chen, D. C. Chrzan, G. X. Gu, Nano-topology optimization for materials design with atom-by-atom control. *Nat. Commun.* **11**, 3745 (2020).
23. A. Zunger, Inverse design in search of materials with target functionalities. *Nat. Rev. Chem.* **2**, 1–16 (2018).
24. J. Peurifoy *et al*., Nanophotonic particle simulation and inverse design using artificial neural networks. *Sci. Adv.* **4**, eaar4206 (2018).
25. K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, A. Walsh, Machine learning for molecular and materials science. *Nature* **559**, 547–555 (2018).
26. C. T. Chen, G. X. Gu, Generative deep neural networks for inverse materials design using backpropagation and active learning. *Adv. Sci. (Weinh.)* **7**, 1902607 (2020).
27. S. L. Brunton, B. R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **52**, 477–508 (2020).
28. M. Raissi, A. Yazdani, G. E. Karniadakis, Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science* **367**, 1026–1030 (2020).
29. M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **378**, 686–707 (2019).
30. B. Zheng, G. X. Gu, Machine learning-based detection of graphene defects with atomic precision. *Nano-Micro Lett.* **12**, 181 (2020).
31. A. D. Jagtap, K. Kawaguchi, G. E. Karniadakis, Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J. Comput. Phys.* **404**, 109136 (2020).
32. J. Sirignano, K. Spiliopoulos, DGM: A deep learning algorithm for solving partial differential equations. *J. Comput. Phys.* **375**, 1339–1364 (2018).
33. Z. Zhang, G. X. Gu, Finite element based deep learning model for deformation behavior of digital materials. *Adv. Theory Simul.* **3**, 2000031 (2020).
34. L. Lu *et al*., Extraction of mechanical properties of materials through deep learning from instrumented indentation. *Proc. Natl. Acad. Sci. U.S.A.* **117**, 7052–7062 (2020).
35. B. Ni, H. Gao, A deep learning approach to the inverse problem of modulus identification in elasticity. *MRS Bull.* **46**, 19–25 (2021).
36. A. G. Baydin, B. A. Pearlmutter, A. A. Radul, J. M. Siskind, Automatic differentiation in machine learning: a survey. *J. Mach. Learn. Res.* **18**, 5595–5637 (2017).
37. J. Yu *et al*., "Generative image inpainting with contextual attention" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Institute of Electrical and Electronics Engineers, 2018), pp. 5505–5514.
38. J. Yu *et al*., "Free-form image inpainting with gated convolution" in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (Institute of Electrical and Electronics Engineers, 2019), pp. 4471–4480.
39. I. Goodfellow *et al*., "Generative adversarial nets" in *Advances in Neural Information Processing Systems*, Z. W. Ghahramani, Ed. *et al*. (Curran Associates Inc., 2014), pp. 2672–2680.
40. X. Wang *et al*., "Esrgan: Enhanced super-resolution generative adversarial networks" in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops* (Springer International Publishing, 2018).
41. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. arXiv [Preprint] (2014). https://arxiv.org/abs/1412.6980 (Accessed 7 February 2020).
42. M. Abadi *et al*., Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv [Preprint] (2016). https://arxiv.org/abs/1603.04467v2 (Accessed 7 February 2020).