Lähivaara, Timo; Kärkkäinen, Leo; Huttunen, Janne M.J.; Hesthaven, Jan S.

Deep convolutional neural networks for estimating porous material parameters with ultrasound tomography

Published in: Journal of the Acoustical Society of America

DOI: 10.1121/1.5024341

Published: 01/02/2018

Please cite the original version:

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.
Deep convolutional neural networks for estimating porous material parameters with ultrasound tomography

Timo Lähivaara,1,a) Leo Kärkkäinen,2,b) Janne M. J. Huttunen,2,b) and Jan S. Hesthaven3

1Department of Applied Physics, University of Eastern Finland, Kuopio, Finland
2Nokia Technologies, Espoo, Finland
3Computational Mathematics and Simulation Science, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

(Received 27 September 2017; revised 4 January 2018; accepted 29 January 2018; published online 22 February 2018)

The feasibility of data based machine learning applied to ultrasound tomography is studied to estimate water-saturated porous material parameters. In this work, the data to train the neural networks is simulated by solving wave propagation in coupled poroviscoelastic-viscoelastic-acoustic media. As the forward model, a high-order discontinuous Galerkin method is considered, while deep convolutional neural networks are used to solve the parameter estimation problem. In the numerical experiment, the material porosity and tortuosity is estimated, while the remaining parameters which are of less interest are successfully marginalized in the neural networks-based inversion. Computational examples confirm the feasibility and accuracy of this approach.

© 2018 Acoustical Society of America. https://doi.org/10.1121/1.5024341

I. INTRODUCTION

Measuring the porous properties of a medium is a demanding task. Different parameters are often measured by different application-specific methods, e.g., the porosity of rock is measured by weighing water-saturated samples and comparing their weight with dried samples. The flow resistivity of the damping materials can be computed from the pressure drop caused by the gas flowing through the material. In medical ultrasound studies, the porosity of the bone is determined indirectly by measuring the ultrasound attenuation as the wave passes through the bone.

For the porous material characterization, information carried by waves provides a potential way to estimate the corresponding material parameters. Ultrasound tomography (UST) is one technique that can be used for material characterization purposes. In this technique, an array of sensors is placed around the target. Typically, one of the sensors is acting as a source while others are receiving the data. By changing the sensor that acts as a source, a comprehensive set of wave data can be recorded which can be used to infer the material properties. For further details on the UST, we refer to Ref. 1, and references therein.

The theory of wave propagation in porous media was rigorously formulated in the 1950s and 1960s by Biot.2–5 The model was first used to study the porous properties of bedrock in oil exploration. Since then, the model has been applied and further developed in a number of different fields.5–9 The challenge of Biot’s model is its computational complexity. The model produces several different types of waveforms, i.e., the fast and slow pressure waves and the shear wave, the computational simulation of which is a demanding task even for modern supercomputers. Computational challenges further increase when attempting to solve inverse problems, as the forward model has to be evaluated several times.

In this work, we consider a process for the parameter estimation, comprising two sub-tasks: (1) forward model: the simulation of the wave fields for given parameter values and (2) inverse problem: estimation of the parameters from the measurements of wave fields. The inverse problem is solved using deep convolutional neural networks that provide a framework to solve the inverse problems: we can train a neural network as a model from wave fields to the parameters. Over the last decades, neural networks have been applied in various research areas such as image recognition,10,11 cancer diagnosis,12,13 forest inventory,14,15 and groundwater resources.16,17 Deep convolutional neural networks (CNNs) are a special type of deep neural networks11,18,19 that employ convolutions instead of matrix multiplications (e.g., to reduce the number of unknowns). In this study, we employ deep convolutional neural networks to two-dimensional ultrasound data.

The structure of the rest of the paper is as follows. First, in Sec. II, we formulate the poroviscoelastic and viscoelastic models and describe the discontinuous Galerkin method. Then, in Sec. III, we describe the neural networks technique for the prediction of material parameters. Numerical experiments are presented Sec. IV. Finally, conclusions are given in Sec. V.

A. Model justification

The purpose of this paper is to study the feasibility of using a data based machine learning approach in UST to characterize porous material parameters. In the synthetic model setup, we have a cylindrical water tank including an elastic shell layer. Ultrasound sensors are placed inside the water tank. In the model, we place a cylindrically shaped
porous material sample in water and estimate its porosity and tortuosity from the corresponding ultrasound measurements by a convolutional neural network. Figure 1 shows a schematic drawing of a UST setup studied in this work.

The proposed model has a wide range of potential applications. Cylindrically shaped core samples can be taken of bedrock or manmade materials, such as ceramics or concrete, to investigate their properties. In addition, from the medical point of view, core samples can be taken, for example, of cartilage or bones for diagnosing purposes. Depending on the application, the model geometry and sample size together with sensor setup needs to be scaled.

II. WAVE PROPAGATION MODEL

In this paper, we consider wave propagation in isotropic coupled poroviscoelastic-viscoelastic-acoustic media. In Secs. II A and II B, we follow Refs. 20–23 and formulate the poroviscoelastic and viscoelastic wave models. The discontinuous Galerkin method is discussed in Sec. II C.

A. Biot’s poroelastic wave equation

We use the theory of wave propagation in poroelastic media developed by Biot in Refs. 2–5. We express Biot equations in terms of solid displacement \( u_s \) and the relative displacement of fluid \( \bar{w} = \phi(u_f - u_s) \), where \( \phi \) is the porosity and \( u_f \) is the fluid displacement. In the following, \( \rho_s \) and \( \rho_f \) denote the solid and fluid densities, respectively. We have

\[ \rho_s \frac{\partial^2 u_s}{\partial t^2} + \rho_f \frac{\partial^2 \bar{w}}{\partial t^2} = \nabla \cdot \mathbf{T}, \quad (1) \]

where \( \rho_s = (1 - \phi) \rho_s + \phi \rho_f \) is the average density, \( \mathbf{T} \) is the total stress tensor, and \( \mathbf{T}_f \) is the fluid stress tensor. In Eq. (2), \( \tau \) is the tortuosity, \( \eta \) is the fluid viscosity, and \( k \) is the permeability.

The third term in Eq. (2) is a valid model at low frequencies, when the flow regime is laminar (Poiseuille flow). At high frequencies, inertial forces may dominate the flow regime. In this case, the attenuation model may be described in terms of viscous relaxation mechanics as discussed, for example, in Refs. 21 and 23. In this paper, we use the model derived in Ref. 23. The level of attenuation is controlled by quality factor \( Q_0 \).

One can express the stress tensors as

\[ \mathbf{T} = 2\mu_f \mathbf{E} + \left( \kappa_f + x^2 M - \frac{2}{3} \mu_f \right) \text{tr} \left( \mathbf{E} \right) \mathbf{I} - x M \zeta, \quad (3) \]

\[ \mathbf{T}_f = M \left( x \text{tr} \mathbf{E} - \zeta \right) \mathbf{I}, \quad (4) \]

where \( \mu_f \) is the frame shear modulus, \( \mathbf{E} = \frac{1}{2} \left( \nabla u_f + (\nabla u_f)^T \right) \) denote the solid strain tensor, \( \kappa_f \) is the frame bulk modulus, \( \text{tr} (\cdot) \) is the trace, \( \mathbf{I} \) is the identity matrix, and \( \zeta = -\nabla \cdot \bar{w} \) is the variation of the fluid content. The effective stress constant \( x \) and modulus \( M \), given in Eqs. (3) and (4), can be written as \( x = 1 - \kappa_f / \kappa_s \), where \( \kappa_s \) is the solid bulk modulus, and \( M = \kappa_s / (x - \phi [1 - \kappa_s / \kappa_f]) \), where \( \kappa \) is the fluid bulk modulus.

B. Viscoelastic wave equation

The following discussion on the elastic wave equation with viscoelastic effects follows Carcione’s book.20 in which a detailed discussion can be found. Expressed as a second order system, the elastic wave equation can be written in the following form:

\[ \rho_e \frac{\partial^2 \mathbf{u}_e}{\partial t^2} = \nabla \cdot \mathbf{S} + \mathbf{s}, \quad (5) \]

where \( \rho_e \) is the density, \( \mathbf{u}_e \) the elastic displacement, \( \mathbf{S} \) is a stress tensor, and \( \mathbf{s} \) is a volume source. In the two-dimensional viscoelastic (isotropic) case considered here, components of the solid stress tensor \( \mathbf{S} \) may be written as

\[ \sigma_{11} = (\lambda_e + 2\mu_e) \varepsilon_{11} + \lambda_e \varepsilon_{22} + (\lambda_e + \mu_e) \sum_{\ell=1}^{L_1} \nu^{(\ell)}_{11}, \quad (6) \]

\[ \sigma_{22} = (\lambda_e + 2\mu_e) \varepsilon_{22} + \lambda_e \varepsilon_{11} + (\lambda_e + \mu_e) \sum_{\ell=1}^{L_1} \nu^{(\ell)}_{11}, \quad (7) \]

\[ \sigma_{12} = 2\mu_e \varepsilon_{12} + 2\mu_e \sum_{\ell=1}^{L_1} \nu^{(\ell)}_{12}, \quad (8) \]
where $\lambda_0$ and $\mu_0$ are the unrelaxed Lamé coefficients, $\epsilon_{11}$, $\epsilon_{22}$, and $\epsilon_{12}$ are the strain components, and $L_\ell$ is the number of relaxation terms. The memory variables $\nu_1^{(\ell)}$, $\nu_2^{(\ell)}$, and $\nu_1^{(\ell)}$ satisfy

$$\frac{\partial \nu_1^{(\ell)}}{\partial t} = -\frac{\nu_1^{(\ell)}}{\tau_{\sigma_1^{(\ell)}}} + \phi_1^{(0)}(\epsilon_{11} + \epsilon_{22}),$$

$$\frac{\partial \nu_2^{(\ell)}}{\partial t} = -\frac{\nu_2^{(\ell)}}{\tau_{\sigma_2^{(\ell)}}} + \phi_2^{(0)}(\epsilon_{11} - \epsilon_{22})/2, \quad \ell = 1, \ldots, L_\ell,$$

$$\frac{\partial \nu_1^{(\ell)}}{\partial t} = -\frac{\nu_1^{(\ell)}}{\tau_{\sigma_1^{(\ell)}}} + \phi_2^{(0)}(\epsilon_{12}),$$

where

$$\phi_\ell^{(0)}(t) = \frac{\tau_{\sigma_1^{(\ell)}}}{\tau_{\sigma_1^{(\ell)}} - \tau_{\sigma_2^{(\ell)}}} \left(1 + \frac{\tau_{\sigma_1^{(\ell)}}}{\tau_{\sigma_2^{(\ell)}}} \left(\sum_{\ell=1}^{n} \frac{\tau_{\sigma_2^{(\ell)}}}{\tau_{\sigma_1^{(\ell)}}}\right)^{-1}\right) \times \exp\left(-t/\tau_{\sigma_2^{(\ell)}}\right), \quad k = 1, 2.$$

In Eq. (12), $\tau_{\sigma_1^{(k)}}$ and $\tau_{\sigma_2^{(k)}}$ are relaxation times corresponding to dilatational ($k=1$) and shear ($k=2$) attenuation mechanisms.

The acoustic wave equation can be obtained from the system by setting the Lamé coefficient $\mu_0$ to zero.

### C. Discontinuous Galerkin method

Wave propagation in coupled poroviscoelastic-viscoelastic-acoustic media can be solved using the discontinuous Galerkin (DG) method (see, e.g., Refs. 22–26), which is a well-known numerical approach to numerically solve differential equations. The DG method has properties that makes it well-suited for wave simulations, e.g., the method can be effectively parallelized and it can handle complex geometries and, due to its discontinuous nature, large discontinuities in the material parameters. These are all properties that are essential features for the method to be used in complex wave problems. Our formulation follows Ref. 27, where a detailed account of the DG method can be found.

### III. ESTIMATING MATERIAL PARAMETERS BY NEURAL NETWORKS

The aim of this paper is to estimate porous material parameters by applying artificial neural networks trained to simulated data. Compared to traditional inverse methods, neural networks have an advantage that they allow computationally efficient inferences. In other words, after the network has been learned, inferences can be carried out using the network without evaluating the forward model. Furthermore, the neural networks provide a straightforward approach to marginalize uninteresting parameters in the inference.

First we will give a brief summary to deep neural networks. For a wider representation of the topic, the reader is directed to the review articles by LeCun et al.\textsuperscript{11} and Bengio,\textsuperscript{19} or to the book by Buduma.\textsuperscript{28}

We consider the following supervised learning task. We have a set of input data $\{X_i\}$ with a labels $\{Y_i\}$ ($\ell = 1, \ldots, N$). In our study, the input data comprises of the measured ultrasound fields considered as “images” such that rows correspond to temporal data and columns to receiver and the outputs are corresponding material parameters. The “image” should be interpreted as two dimensional data, not a traditional picture; there is no color mapping used in our algorithm. The aim is to find a function $\Theta$ between the inputs and outputs,

$$Y = \Theta(X).$$

The task is to find a suitable form for the function and learn it from the given data. Contrary to traditional machine learning in which the features are pre-specified, deep learning has an advantage that features are learned from the data.

Neural networks are a widely used class of functions in machine learning. Figure 2 (left) shows an example of a neural network with one hidden layer. Each circular node represents an artificial neuron and an arrow represents a connection between the output of one neuron to the input of the other. This network has four inputs $x_1, \ldots, x_4$ (input layer), five hidden units with output activations $a_1, \ldots, a_5$ (hidden layer), and two outputs $y_1$ and $y_2$ (output layer). Neurons in the hidden layer are computational units that, for a given input $(x_1, \ldots, x_4)$, calculate the activations as

$$a_j = \sigma\left(\sum_{i=1}^{4} w_{ji}^{(1)} x_i + b_j^{(1)}\right), \quad j = 1, \ldots, 5,$$

where $w_{ji}^{(1)}$ is a weight assigned to the connection between the $i$th input, $b_j^{(1)}$ is a bias term and $j$th activation, and $\sigma$ is a non-linear function (nonlinearity). Similarly, the neurons at output layer are units for which the output is calculated as

![Diagram](image-url)
\[ y_j = \sigma \left( \sum_{i=1}^{5} w_j^{(2)} a_i + b_j^{(2)} \right), \quad j = 1, \ldots, 2, \]

where \( w_j^{(2)} \) and \( b_j^{(2)} \) are again weights and biases.

At present, the most typically used nonlinearity is the rectified linear unit (ReLU), \( \sigma(x) = \max(0, x) \).\(^9\) Over the past decades, smoother nonlinearities such as the sigmoid function \( \sigma(x) = (1 + \exp(-x))^{-1} \) or tanh have been used as nonlinearity, but the ReLU typically allows better performance in the training of deep neural architectures on large and complex datasets.\(^11\) Some layers can also have different nonlinearities than others, but here we use the same \( \sigma \) for notational convenience.

It is easy to see that the neural network can be presented in matrix form as

\[
A = \sigma(w^{(1)} \cdot X + b^{(1)}), \quad Y = \sigma(w^{(2)} \cdot A + b^{(2)}),
\]

where \( X = (x_1, \ldots, x_d)^T \), \( Y = (y_1, y_2)^T \), and \( A = (a_1, \ldots, a_s)^T \). \( w^{(1)} \) and \( w^{(2)} \) are matrices/tensors comprising of the weights \( w_i^{(1)} \) and \( w_i^{(2)} \), respectively, \( b^{(1)} \) and \( b^{(2)} \) are the vectors comprising of the bias variables \( b_i^{(1)} \) and \( b_i^{(2)} \), respectively, and the function \( \sigma \) operates on the vectors element-wise. The network can be also written as a nested function \( \Theta_\theta(X) = \sigma(w^{(2)}(\sigma(w^{(1)} \cdot X + b^{(1)})) + b^{(2)}) \), where \( \theta \) represents the unknowns of the network \( \{\theta = (w^{(1)}, b^{(1)}, w^{(2)}, b^{(2)})\} \).

Deeper networks can be constructed similarly. For example, a network with two hidden layers can be formed as \( A^{(1)} = \sigma(w^{(1)} \cdot X + b^{(1)} \cdot A^{(1)}), A^{(2)} = \sigma(w^{(2)} \cdot A^{(1)} + b^{(2)}) \), and \( Y = \sigma(w^{(3)} \cdot A^{(2)} + b^{(3)}) \), or \( \Theta_\theta(X) = \sigma(w^{(3)} \cdot \sigma(w^{(2)} \cdot \sigma(w^{(1)} \cdot X + b^{(1)}) \cdot b^{(2)})) + b^{(3)} \), where \( \theta = (w^{(1)}, b^{(1)}, w^{(2)}, b^{(2)}, w^{(3)}, b^{(3)}) \).

The neural networks described above are called fully connected: there are connections between all nodes of the network. A disadvantage of such fully connected networks is that total number of free parameters can be enormous especially when the dimensions of input data is high (e.g., high resolution pictures) and/or the network has several hidden layers. CNNs were introduced\(^6,10,30\) to overcome this problem. Figure 2 (right) shows an example of a 1D-convolutional layer. In a convolutional layer, the matrix multiplication is replaced with a convolution: for example, in 1D, the activations of the layer are calculated as

\[
A = \sigma(w^{(1)} \ast X + B^{(1)}),
\]

where the asterisk denotes the discrete 1D-convolution and \( w^{(1)} \) is a set of filter weights (a feature to be learned). In 2D, the input \( X \) can be considered as an image and the convolution \( w^{(1)} \ast X \) is two dimensional with a convolution matrix or mask \( w^{(1)} \) (a filtering kernel).

In a practical solution, however, only one convolution mask (per layer) is typically not enough for good performance. Therefore, each layer includes multiple masks that are applied simultaneously and the output of the convolutional layer is a bank (channels) of images. This three dimensional object forms the input of the next layer such that each input channel has its own bank of filtering kernels and the convolution is effectively three dimensional.

Usually, convolutional neural networks also employ pooling\(^10,31\) for down-sampling (to reduce dimension of activations). See, for example, LeCun et al.\(^11\) or Buduma\(^28\) for details.

The training of the neural networks is based on a set of training data \( \{X_t, Y_t\} \). The purpose is to find a set if weights and biases that minimize the discrepancy between the outputs \( \{Y_t\} \) and the corresponding predicted values given by the neural networks \( \{\Theta_\theta(X_t)\} \). Typically, in regression problems, this is achieved by minimizing the quadratic loss function \( f(\theta) \) over the simulation dataset

\[
f(\theta) = f(\theta; \{X_t\}, \{Y_t\}) = \frac{1}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} (\Theta_\theta(X_t) - Y_t)^2
\]

IV. NUMERICAL EXPERIMENT

In this section, we present the results obtained from testing the data driven approach to estimate porous material parameters in ultrasound tomography. In this paper, initial conditions are always assumed to be zero and we apply the free condition as a boundary condition.

In the following results, time integration is carried out using an explicit low-storage Runge-Kutta scheme.\(^33\) For each simulation, the length of the time step \( \Delta t \) is computed from

\[
\Delta t = \left( \frac{h_{\text{min}}}{2c_{\text{max}}^\ell(N_{\ell})^z} \right)_{\text{min}}, \quad \ell = 1, \ldots, K,
\]

where \( c_{\text{max}}^\ell \) is the maximum wave speed, \( N_{\ell} \) is the order of the polynomial basis, \( h_{\text{min}} \) is the smallest distance between
two vertices in the element \( \ell \), and \( K \) is the number of elements.

A. Model setup

Let us first introduce the model problem. Figure 3 shows the studied two dimensional problem geometry. The propagation medium contains three subdomains: a cylindrical shaped poroelastic inclusion (black), a fluid (light gray), and a solid shell (dark gray). The computational domain is a circle with radius of 10 cm with a 1 cm thick shell layer. A circular shaped poroelastic inclusion with a radius of 4 cm is vertically shifted to 2 cm from the center of the circular domain. Shifting the target from the center avoids symmetrically positioned sensors (with respect to x axis) to receive equal signal and therefore have more information from the target.

A total of 26 uniformly distributed ultrasound sensors are located at a distance of 8 cm from the center of the circle. The source is introduced on the strain components \( u_{11} \) and \( u_{22} \) by the first derivative of a Gaussian function with frequency \( f_0 = 40 \) kHz, a time delay \( t_0 = 1.2 f_0 \). The sensor that is used as a source does not collect data in our simulations. Receivers collect solid velocity components \( u_s \) and \( v_s \). In the following, the simulation time is 0.4 ms. Note that recorded data is downsampled to a sampling frequency of 800 kHz on each receiver.

The inclusion is fully saturated with water. The fluid parameters are: the density \( \rho_f = 1020 \) kg/m\(^3\), the fluid bulk modulus \( K_f = 2.295 \) GPa, and the viscosity \( \eta = 1.0 \times 10^{-3} \) Pa s. All other material parameters of the inclusion are assumed to be unknown. In this paper, we assume a relatively wide range of possible parameter combinations, see Table I. Furthermore, the unknown parameters are assumed to be uncorrelated. The physical parameter space gives \( \sim 1.5 \) kHz as an upper bound for Biot’s characteristic frequency

\[
f_c = \frac{\eta \phi}{2 \pi \mu k}
\]

and hence we operate in Biot’s high-frequency regime \((f_c < f_0)\) in all possible parameter combinations.

For the fluid subdomain (water), we set the density \( \rho_f = 1020 \) kg/m\(^3\), the first Lamé parameter \( \lambda_f = 2.295 \) GPa, and the second Lamé parameter \( \mu_f = 0 \). The elastic shell layer has the following parameters: \( \rho_s = 2000 \) kg/m\(^3\), \( \lambda_s = 12.940 \) GPa, and \( \mu_s = 5.78 \) GPa. The relaxation times for the viscoelastic attenuation in the shell layer are given in Table II.

The derived wave speeds for each subdomain are given in Table III. For the poroelastic inclusion, both the minimum and maximum wave speeds are reported. It should be noted that the reported values for the wave speeds in the inclusion correspond to the values generated by sampling the material parameters and hence they may not correspond to the global maximum and minimum values. A detailed approach for calculating wave speeds is given in Ref. 23.

It should be noted that, depending on the application, only some of the material parameters are unknowns of interest. In this work, we focus on estimating the porosity and tortuosity of the inclusion while the solid density and bulk modulus, frame bulk and shear modulus, permeability, and quality factor are marginalized in the neural networks-based inversion algorithm, discussed in detail below.

B. Training, validation, and test data

For the convolutional neural networks algorithm used in this work, the recorded wave data is as expressed as images

![Figure 3](image-url)

FIG. 3. Figure shows the problem geometry. In the graph, the circles denote the receivers and the cross denotes the source.
$X \in \mathbb{R}^d$, $d = N_t \times N_r$, where $N_t$ denotes the number of time steps and $N_r$ the number of receivers. The input dimension is $d = 320 \times 25$, as the data $X$ can be seen as a 2D-image, comprising 25 pixels, corresponding to the receiver positions, times 320 pixels corresponding to the time evolution of the signal.

As the data on each column of the image, we use the following:

$$X(:, \ell) = \frac{x_\ell \left( u_\ell - u_{s, noi} \right) + y_\ell \left( v_\ell - v_{s, noi} \right)}{\sqrt{(x_\ell)^2 + (y_\ell)^2}},$$

where $(x_\ell, y_\ell)$ are the coordinates of the $\ell$th receiver and $(u_\ell, v_\ell)$ are the horizontal and vertical velocity components on the $\ell$th receiver. In Eq. (16) subscript $\text{noi}$ denotes the data that is simulated without the porous inclusion. Two example images are shown in Fig. 4.

We have generated a training data set comprising 15,000 samples using computational grids that have $\sim 3$ elements per wavelength. The physical parameters for each sample are drawn from the uniform distribution (bounds given in Table I). The order of the basis functions is selected separately for each element of the grid. The order $N_\ell$ of the basis function in element $\ell$ is defined by

$$N_\ell = \left\lceil \frac{2\pi h_{\ell, \max} c_{\ell}}{\lambda_{\ell}^d} + b \right\rceil,$$

where $\lambda_{\ell}^d = c_{\ell, \min} / f_0$ is the wavelength, $c_{\ell, \min}$ is the minimum wave speed, and $\lceil \cdot \rceil$ is the ceiling function. The parameters $a$ and $b$ control the local accuracy on each element. Following Refs. 35 and 36, we set $(a, b) = (1.0294, 0.7857)$.

Figure 5 shows two examples of computational grids and the corresponding basis order selection on each triangle. Example grids consist of 466 elements and 256 vertices ($h_{\min} = 0.81 \text{ cm}$ and $h_{\max} = 1.95 \text{ cm}$) (sample 1) and 1156 elements and 601 vertices ($h_{\min} = 0.32 \text{ cm}$ and $h_{\max} = 1.75 \text{ cm}$) (sample 2).

Figure 6 shows two snapshots of the scattered solid velocity field $(u - u_{s, noi})^2 + (v - v_{s, noi})^2$ for two time instants. At the first time instant, the transmitted fast pressure wave and also the first reflected wave front is clearly visible. At the second time instant, all wave components have reflected back from the left surface of the phantom. Furthermore, more
complicated wave scattering patterns can be seen inside the porous inclusion. These wave fields demonstrate how the received signals are obtained as combinations of multiple wave fronts.

To include observation noise in the training, each image in the simulation set is copied five times and each of the copies is corrupted with Gaussian noise of the form

\[ X_{\text{noised}} = X_t + A \epsilon^A + B|X_t|\epsilon^B, \tag{18} \]

where \( \epsilon^A \) and \( \epsilon^B \) are independent zero-mean identically distributed Gaussian random variables. The second term represents additive white noise and the last term represents noise relative to the signal strength. To represent a wide range of different noise levels, for each sample image, the coefficients \( A \) and \( B \) are randomly chosen such that the standard deviations of the white noise component is between 0.03%–5% (varying logarithmically), and the standard deviations of the relative component is between 0%–5%. The total number of samples in the training set is \( N_{\text{nn}} = 5 \times 15000 = 75000 \).

Furthermore, two additional data sets were generated: a validation data set and a test set, that both comprise 3000 samples. In machine learning, the validation data set is traditionally used to evaluate performance during the training process and the test set is used for the final evaluation of the network.

The neural network architecture is shown in Table IV and Fig. 7. The CNN architecture is similar to ones used for image classification, for example, ALEXNET,10 with some exceptions. In our problem, the input data can be considered as one channel images instead of color images with three channels (i.e., we do not apply any color mapping). Our network also lacks the SOFTMAX layer, which is used as an outmost layer to provide the classification. Instead, the outmost two layers are simply fully connected layers. In addition, our network also has a smaller number of convolutional and fully connected layers with smaller dimensions in filter banks, which leads to significantly smaller number of unknowns. We wish, however, to note that the purpose was not to find the most optimal network architecture. For example, there can be other architectures that...
can provide similar performance with an even smaller number of unknowns. Furthermore, similar performance can also be achieved with fully connected networks with ~3 layers, but at the expense of significantly larger number of unknowns.

The loss function is chosen to be quadratic [Eq. (13)]. The implementation is carried out using TENSORFLOW, which is a PYTHON toolbox for machine learning. The optimization was carried out with the ADAM optimizer.\(^{38}\) The batch size for the stochastic optimization is chosen to be 50 samples.

### D. Results

Figure 8 shows the loss of the training and validation data. The loss is shown for the two unknown parameters of interest, e.g., porosity and tortuosity. In both cases, we observe that the network has practically reached its generalization capability at least after 2000 full training cycles. The accuracy of the network is affected by the marginalization over all other parameters. In principle, the effect of the other parameters could compensate the changes that, for example, a different porosity would cause, leaving the waveform of the measurement intact.

We have applied the trained network to predict porosity and tortuosity from images of the test that are corrupted with the white noise component with low noise level (Fig. 9), moderate noise level (Fig. 10), and high noise level (Fig. 11). The figures also include error histograms. Table V shows statistics of the prediction error for different noise levels. Figure 12 shows the maximum absolute error and the root-mean-square error [the square root of Eq. (13)] as a function of the noise level in white noise component. The predictions are slightly positively biased with smaller noise levels, but positive bias is diminished with higher noise levels. Such a behavior might be due to the discretization error in forward models (the simulation of training and testing data were carried out using different levels of discretizations) which may dominate with lower observation noise levels but becomes negligible with higher noise levels. On the contrary, with high levels of noise, the predictions are negatively biased especially for larger values of porosity and tortuosity.

We have also studied the effect of the relative noise [the third term in Eq. (18)] to the results. Figure 13 shows the maximum absolute error and the root-mean-square error as a function of the noise level in the relative component. As we can see, the predictions are almost unaffected by the relative error even with significantly high noise levels (~5%–10%).

![FIG. 8. Training and validation loss for porosity (left) and tortuosity (right) as function of number of epochs (full training cycles in stochastic optimization).](image)

![FIG. 9. Predicted porosities (left) and tortuosities (middle) for the test data with white noise of 0.8% (relative to the maximum absolute value of the signal in the training set). Bottom row (left and middle) shows histograms of the prediction error (difference between the predicted and true values). The right column shows two examples of the sample images in this noise level.](image)
FIG. 10. Predicted porosities (left) and tortuosities (middle) for the test data with white noise of 3.2% (relative to the maximum absolute value of the signal in the training set). Bottom row (left and middle) shows histograms of the prediction error (difference between the predicted and true values). The right column shows two examples of the sample images in this noise level.

FIG. 11. Predicted porosities (left) and tortuosities (middle) for the test data with white noise of 8.1% (relative to the maximum absolute value of the signal in the training set). Bottom row (left and middle) shows histograms of the prediction error (difference between the predicted and true values). The right column shows two examples of the sample images in this noise level.

TABLE V. Bias, standard deviation, kurtosis, and percentiles (25% and 75%) for the errors of the predicted porosities and tortuosities with different noise levels.

<table>
<thead>
<tr>
<th>Noise level</th>
<th>Porosity</th>
<th></th>
<th></th>
<th></th>
<th>Tortuosity</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Bias</td>
<td>S.D.</td>
<td>Kurtosis</td>
<td>25%</td>
<td>75%</td>
<td>Bias</td>
<td>S.D.</td>
<td>Kurtosis</td>
<td>25%</td>
</tr>
<tr>
<td>0%</td>
<td>0.012</td>
<td>0.016</td>
<td>3.2</td>
<td>0.0011</td>
<td>0.023</td>
<td>0.041</td>
<td>0.087</td>
<td>14</td>
</tr>
<tr>
<td>0.16%</td>
<td>0.012</td>
<td>0.016</td>
<td>3.2</td>
<td>0.0012</td>
<td>0.023</td>
<td>0.042</td>
<td>0.087</td>
<td>14</td>
</tr>
<tr>
<td>0.49%</td>
<td>0.013</td>
<td>0.017</td>
<td>3.3</td>
<td>0.0013</td>
<td>0.023</td>
<td>0.044</td>
<td>0.091</td>
<td>15</td>
</tr>
<tr>
<td>0.81%</td>
<td>0.013</td>
<td>0.018</td>
<td>3.5</td>
<td>0.00078</td>
<td>0.024</td>
<td>0.046</td>
<td>0.096</td>
<td>15</td>
</tr>
<tr>
<td>1.6%</td>
<td>0.013</td>
<td>0.023</td>
<td>4.1</td>
<td>-0.002</td>
<td>0.026</td>
<td>0.047</td>
<td>0.11</td>
<td>14</td>
</tr>
<tr>
<td>3.2%</td>
<td>0.0087</td>
<td>0.035</td>
<td>4.4</td>
<td>-0.014</td>
<td>0.028</td>
<td>0.035</td>
<td>0.16</td>
<td>11</td>
</tr>
<tr>
<td>4.9%</td>
<td>-0.0029</td>
<td>0.049</td>
<td>4.2</td>
<td>-0.034</td>
<td>0.027</td>
<td>-0.0012</td>
<td>0.22</td>
<td>8.1</td>
</tr>
<tr>
<td>8.1%</td>
<td>-0.043</td>
<td>0.081</td>
<td>3.6</td>
<td>-0.093</td>
<td>0.0099</td>
<td>-0.12</td>
<td>0.34</td>
<td>5.1</td>
</tr>
</tbody>
</table>

Lähivaara et al.
V. CONCLUSIONS

In this paper, we proposed the use of CNNs for estimation of porous material parameters from synthetic ultrasound tomography data. In the studied model, ultrasound data were generated in a water tank into which the poroelastic material sample was placed. A total of 26 ultrasound sensors were positioned in the water. One of the sensors generated the source pulse while others were used in the receiving mode. The recorded velocity data were represented as images which were further used as an input to the CNNs.

In the experiment, the parameter space for the porous inclusion was assumed to be large. For example, the porosity of the inclusion was allowed to span the interval from 1% to 99%. The selected parameter space models a different type of materials. We estimated the porosity and tortuosity of the porous material sample while all other material parameters were considered as nuisance parameters (see Table I). Based on the results, it seems that these parameters can be estimated with acceptable accuracy with a wide variety of noise levels, while the nuisance parameters are successfully marginalized. The error histograms for both porosity and tortuosity show excellent accuracy in terms of root-mean-square error and bias.

We have marginalized our inference of porosity and tortuosity over six other material parameters (listed in Table I), which makes it possible to detect the primary porous material parameters from the waveforms without the knowledge of the values of the nuisance parameters, even if they can have a significant impact on the waveforms themselves. The success in the marginalization significantly increases the potential of the neural networks for material characterization.

Future studies should include more comprehensive investigation of model uncertainties, including geometrical inaccuracies (positioning and size of the material sample),

\[ E = \text{error function} \]

\[ B \text{ in Eq. (18)} \]

\[ J. \text{ Acoust. Soc. Am. 143 (2), February 2018} \]

\[ \text{Lähivaara et al.} \]
fluid parameter changes, viscoelastic parameters of the shell layer, material inhomogeneities, and the sensor setup. In addition, the extension to the three spatial dimensions together with actual measurements are essential steps to guarantee the effectiveness of the proposed method.

ACKNOWLEDGMENTS

This work has been supported by the strategic funding of the University of Eastern Finland and by the Academy of Finland (Project No. 250215, Finnish Centre of Excellence in Inverse Problems Research). This article is based upon work from COST Action DENORMS CA-15125, supported by COST (European Cooperation in Science and Technology).


