

Christopher Kleman

Aerospace Engineering and Mechanics,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: ckleman@crimson.ua.edu

Shoaib Anwar

Aerospace Engineering and Mechanics,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: sanwar1@crimson.ua.edu

Zhengchun Liu

Data Science and Learning,
Argonne National Laboratory,
Lemont, IL 60439;
Consortium for Advanced Science and
Engineering,
University of Chicago,
Chicago, IL 60637
e-mail: zhengchun.liu@anl.gov

Jiaqi Gong

Computer Science,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: jiaqi.gong@ua.edu

Xishi Zhu

Computer Science,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: xzhu39@crimson.ua.edu

Austin Yunker

Data Science and Learning,
Argonne National Laboratory,
Lemont, IL 60439
e-mail: ayunker@anl.gov

Rajkumar Kettimuthu

Data Science and Learning,
Argonne National Laboratory,
Lemont, IL 60439;
Consortium for Advanced Science and
Engineering,
University of Chicago,
Chicago, IL 60637
e-mail: kettimut@mcs.anl.gov

Jiaze He¹

Aerospace Engineering and Mechanics,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: jhe26@eng.ua.edu

Full Waveform Inversion-Based Ultrasound Computed Tomography Acceleration Using Two-Dimensional Convolutional Neural Networks

Ultrasound computed tomography (USCT) shows great promise in nondestructive evaluation and medical imaging due to its ability to quickly scan and collect data from a region of interest. However, existing approaches are a tradeoff between the accuracy of the prediction and the speed at which the data can be analyzed, and processing the collected data into a meaningful image requires both time and computational resources. We propose to develop convolutional neural networks (CNNs) to accelerate and enhance the inversion results to reveal underlying structures or abnormalities that may be located within the region of interest. For training, the ultrasonic signals were first processed using the full waveform inversion (FWI) technique for only a single iteration; the resulting image and the corresponding true model were used as the input and output, respectively. The proposed machine learning approach is based on implementing two-dimensional CNNs to find an approximate solution to the inverse problem of a partial differential equation-based model reconstruction. To alleviate the time-consuming and computationally intensive data generation process, a high-performance computing-based framework has been developed to generate the training data in parallel. At the inference stage, the acquired signals will be first processed by FWI for a single iteration; then the resulting image will be processed by a pre-trained CNN to instantaneously generate the final output image. The results showed that once trained, the CNNs can quickly generate the predicted wave speed distributions with significantly enhanced speed and accuracy.

[DOI: 10.1115/1.4062092]

Keywords: ultrasonic imaging, full waveform inversion, convolutional neural networks, ultrasound computed tomography, deep learning

1 Introduction

Nondestructive testing (NDT) is a critical part of quality control and ensures the safety of products and infrastructure. The challenge of NDT is often in finding a way to examine and analyze internal

¹Corresponding author.

Manuscript received October 8, 2022; final manuscript received February 17, 2023; published online March 31, 2023. Assoc. Editor: Yuris Dzenis.

This work is in part a work of the U.S. Government. ASME disclaims all interest in the U.S. Government's contributions.

properties or defects that we are unable to physically see or measure. Medical fields face similar challenges in visualizing the properties of internal bodily tissue. In both cases, ultrasound computed tomography (USCT) can be used to help us visualize and analyze these internal properties. Full waveform inversion (FWI) is a USCT technique based on numerical wave propagation [1] that seeks to find a high-resolution subsurface model [2] of a region of interest. The ability to visualize internal defects or varying material properties without causing damage to the underlying structure has applications in many areas, including NDT [3], structural health monitoring [4], aquifer identification [5], early medical diagnosis [6], and industrial process monitoring [7].

The FWI [8,9] process starts with an initial model predicting the region of interest [10] which estimates the underlying distribution of material properties or potential defects. This model is used in combination with the wave equation to generate a predicted wavefield with ultrasonic or seismic excitation [11]. The wavefield signals are recorded at a series of receiver locations as the measurements or data [12]. These predicted measurements are compared to the observed measurements to calculate the data misfit between them. Based on adjoint theory, the misfit data are used to update the model of the region and this process is repeated until the misfit between the predicted and observed data is minimized to an acceptable level [8,13].

While FWI-based USCT [8,9,14–19] has been shown to have applications for numerous industries and is becoming a more commonly used method to reconstruct material properties or damage distribution, this technique is not without its own challenges. Waveform inversion is a typical non-linear and ill-posed inverse problem and existing physics-driven computational methods for solving waveform inversions suffer from cycle-skipping [12] and local-minima issues [20]. Due to the complexity of solving the related inverse problem in FWI, this is often an intensive task, both in terms of mathematical complexity [21] and computational time [22]. In an effort to reduce the required computational time, the number of iterations of the calculations are often limited, or larger errors are allowed leading to more of an approximation rather than an exact solution, which often compromises the quality of the reconstructed model [23]. Because of these complexities in solving traditional FWI, various methods have been researched to solve the inverse problem of FWI and provide a model of the underlying structure directly from the observed full waveform dataset. Classical solutions to the FWI inverse problem are numerically calculated, and early attempts [24] at solving the inverse problem were iterative and data-driven [25].

Modern advancements in machine learning and image generation have shown great success in solving the inverse problem in various image reconstruction-related areas (e.g., X-ray tomography) [26–28]. Related deep-learning approaches also show promise in alleviating the aforementioned challenges in FWI. In recent years, efforts have been made to incorporate a generative adversarial network (GAN) with seismic FWI [4] and similar data-driven inversion methods such as seismic impedance inversion problems [29]. Zhang and Lin [30] proposed a GAN-based model to estimate a velocity map from raw seismic waveform data. Studies where FWI was applied to elastic and transversely isotropic media using recurrent neural networks (RNNs) also showed significant potential in deep-learning-based FWI research [31,32]. In addition to GAN and RNN, some attempts are being made at incorporating physics information into machine learning loss functions to be used in neural networks. These are termed physics-informed neural networks (PINNs) in ultrasound inversion studies [33,34]. Raissi et al. [33] demonstrated that PINN approximates the solution of a partial differential equation (PDE) by training on control points and constraining the solution's time and space derivatives to obey the PDE. In RNN-based FWI, on the other hand, the partial derivatives are computed using finite differencing, and automated differentiation is only used on the model parameters. However, the RNN method requires storage for the entire multi-component wavefield [31].

Most of the above deep-learning-based improvements for FWI are still in geophysics. For FWI-based USCT, deep-learning-based enhancement is still in its infancy. Robins et al., in their deep-learning-based USCT study [35], implemented convolutional neural network (CNN) as a preprocessing step to the input data of FWI and were able to recover high-resolution velocity models. However, to achieve such effective results they had to go through 208 iterations of FWI, making the process still computationally expensive. Feigin et al. [36] analyzed deep-learning-based FWI with a single iteration using high frequencies (5–40 MHz). Nevertheless, high frequencies lead to non-negligible attenuation [37], which creates another model parameter to deal with. Several studies have thereby been performed on neural network architecture in FWI problems, and there is ample room for research on theoretical analysis of network architecture on FWI, the effects of different hyperparameters on final outputs, etc. [31]. Another possible method is to treat the problem as an image-to-image translation and implement a CNN to solve the inverse problems in different fields [27,35,36,38,39].

The present work aims to develop a CNN that will solve the inverse problem of FWI in USCT with low computational processing requirements of the trained machine learning model. Due to the complexity of solving the FWI, some limitations are set in the initial testing and training of the CNN for this application. However, applying adjoint tomography-based deep-learning to the FWI problem takes into account the full non-linearity of the wave propagation [40], allowing for more usable information from each iteration. This can improve the reconstruction quality while also requiring less computational cost than classical FWI methods [41]. If CNNs can be trained to accurately predict the inverse problem for FWI, large reductions in computation requirements can be realized. This serves to reduce the entry cost of implementing FWI-based USCT systems and expand its application, use, and adoption in the industry. We propose to leverage modern advancements in machine learning and image generation to solve the inverse problem as an image translation issue. Our previous work [38] shows promise in applying 1D CNNs for single wave speed regions with the same wave speed. In this paper, we propose to build and train a two-dimensional (2D) CNN to identify and learn patterns to solve the inverse problem from the first iteration FWI input cost-effectively without compensating much on the performance, for multiple regions with various wave speeds. The effects of various loss functions in this CNN-based approach for FWI-based USCT problems have also been studied.

The remainder of the paper is organized as follows. First, the dataset generation will be introduced. This includes the creation of ground truth and FWI datasets which will be used to train, verify, and test the capabilities of the machine learning model, followed by the preparation of this data for use in machine learning. Next, the architecture of the machine learning model will be discussed along with the selection of various hyperparameters used for the training series. Finally, the results of the machine learning model are presented and discussed with an analysis of how each of the chosen hyperparameters affected the outcomes of each of the machine learning iterations.

2 Methods and Data

Our proposed approach is to develop a two-dimensional CNN to accelerate and enhance the inversion results. To build and test this, we need to implement a two-step approach. The first step is to generate datasets to work with; the second step is to train, validate, and test the machine learning models. In the first step of data creation, a set of randomly generated ground truth models has been created, and then a single iteration FWI is performed on the corresponding ultrasonic measurements. Because the combination of indefinite boundaries and varying shapes and sizes can quickly result in very complex challenges to solve, the dataset is limited to only a few regions of simple rectangular

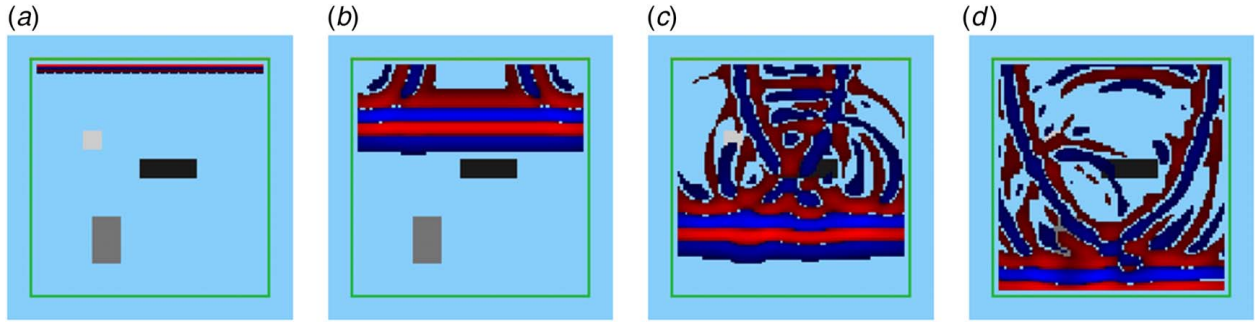


Fig. 1 Illustration of the wave propagation at various times in the domain of 12 mm × 12 mm with 100 transducers near the top wall and 400 sensors (square boundaries) surrounding the domain. There are three unknown materials inside this domain: (a) 0.75 μs, (b) 3.0 μs, (c) 6.0 μs, and (d) 7.5 μs

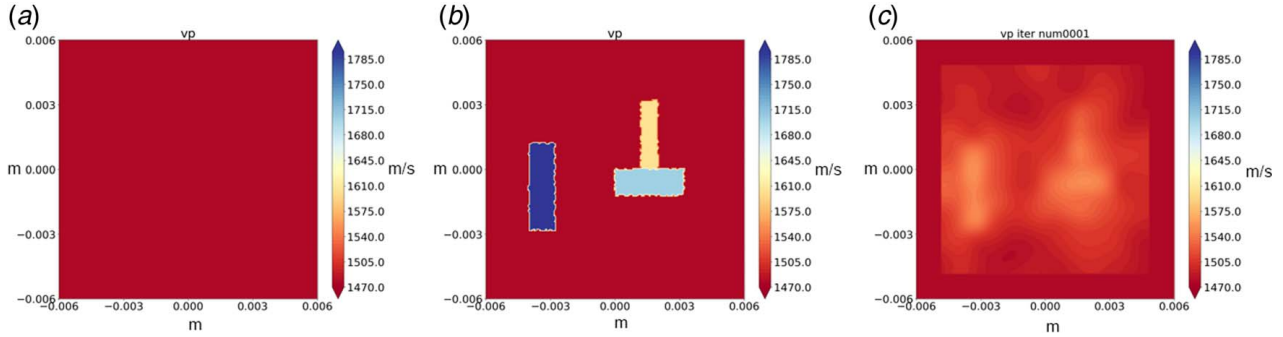


Fig. 2 Longitudinal wavespeed (v_p , m/s) mapping of (a) initial model, (b) true model, and (c) reconstructed FWI model after a single iteration

shapes with definite boundaries, but with different wave propagation speeds in rectangular regions. The inversion images generated by the FWI process have been used as inputs to the machine learning model to train said model to predict what the ground truth solution should be.

2.1 Implementation of Full Waveform Inversion. FWI is a non-linear inversion method that aims to identify model parameters by minimizing the gradient of the misfit function between the observed and synthetic data in an iterative manner [8,13]. The process begins with selecting sources and an assumed model, which is referred to as an initial model. A finite element analysis is performed with this initial model to create the synthetic data. This process is called forward simulation [8] and models the ultrasound wave propagation in the domain. For the acoustic case, sensors measure pressure signals. Hence, the wavefield $\mathbf{s}(\mathbf{x}, t)$ can be expressed as

$$\rho \partial_t^2 s = -\Delta p + \mathbf{f} \quad (1)$$

where p denotes the pressure, ρ is the mass density, and \mathbf{f} is the source excitation force. For waveform tomography, Tarantola [42] introduced the least-squares waveform misfit function. For this study, we also define the misfit function as

$$\chi(\mathbf{m}) = \frac{1}{2} \sum_r \int [p_{\text{syn}}(\mathbf{x}_r, t, \mathbf{m}) - p_{\text{obs}}(\mathbf{x}_r, t)]^2 dt \quad (2)$$

where \mathbf{m} is the model space in ultrasound tomography and depends on material parameters such as density, wavespeed, attenuation, etc. At a lower frequency, attenuation is much lower [37]. For our study, therefore, we neglected attenuation and assumed the density to be constant throughout the process. The p_{obs} and p_{syn} are the pressure signals of the true data and synthetic data, respectively. The gradient

of the misfit function is

$$\delta\chi(\mathbf{m}) = \sum_r \int [p_{\text{syn}}(\mathbf{x}_r, t, \mathbf{m}) - p_{\text{obs}}(\mathbf{x}_r, t)] \delta p_{\text{syn}}(\mathbf{x}_r, t, \mathbf{m}) dt \quad (3)$$

where δp_{syn} is the perturbation in pressure field p_{syn} due to the model perturbation $\delta\mathbf{m}$ [13,43]. The aim of FWI is to reduce the misfit gradient between these two data points so that model parameters can be obtained and a high-resolution, numerical model of the material distribution can be reconstructed as the final model [8]. However, in this study, we stopped the process after a single iteration of inversion. The results of this “single iteration FWI” process and the corresponding true models are then used to train–validate–test the deep-learning model.

2.2 Training and Test Dataset Generation. A spectral finite element solver—called SPEC-FEM2D [44]—was used to simulate forward and adjoint wave propagation in two-dimensional media. A 12 mm × 12 mm water-immersed domain was designed as the 2D media. The domain consists of 900 spectral elements (30 along the x -axis and 30 along the y -axis) in the mesh. Each spectral element contained 25 control points. Excitation was generated from 100 source transducers located at one side of the domain (just below top wall in Fig. 1). All transducers were excited simultaneously to generate a plane wave with a center frequency of 1 MHz, and 400 sensor-elements were setup around the domain to receive the signals. Figure 1 illustrates how the plane wave propagated at different time-steps inside the domain. We built the initial model with a water-immersed environment at room temperature with a 1479.7 m/s wave propagation speed.

The ground truth models were created by introducing regions with unknown material properties (i.e., wave speed in this research). To generate a diverse dataset, the number, length, width, location, and wave speed for these regions were randomly selected. A total of 1000 ground truth models were created. Figure 2 shows the

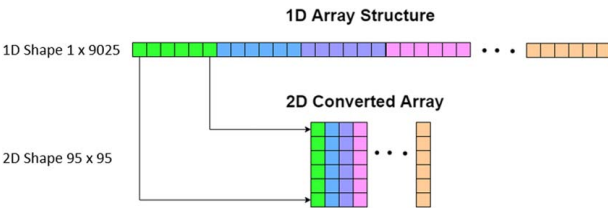


Fig. 3 Representative process of rearranging one-dimensional array into the correct two-dimensional arrangement

longitudinal wavespeed mapping of the initial model, a ground truth model, and an inverted model after a single iteration of FWI of that ground truth model.

We used a python-based framework called SeisFlows for FWI [8,45]. The output of both forward and inverse simulations was the wavespeed information of every control point in the domain saved as separate NumPy arrays.

FWI is a state-of-the-art tool for producing high-accuracy velocity models [46]. However, this method can be very computationally expensive [47]. Initially, the data generation was performed sequentially with one dataset sample being calculated at a time and then later stacked to create a single NumPy output array on a lab desktop. This approach proved to be time-consuming taking more than 15 h to generate 1000 samples. To remove this bottleneck, high-performance computing was used to allow data generation to be performed in parallel. The parallelization process was managed using Simple Linux Utility for a Resource Management (SLURM) software package. A batch of 100 models was simulated as a single SLURM job using 128 processor cores simultaneously. Each SLURM job was completed in approximately 13 min. To generate 1000 models, we needed 10 SLURM jobs. Thus, after parallelization, the time to generate a dataset of 1000 models was reduced to approximately 13 min.

2.3 Data Preprocessing. The output of SPEC2D and SeisFlows is a one-dimensional array which contains wavespeed information of every control point in the spectral elements. However, since the problem is from a two-dimensional region, we needed to convert the data back into the original width and height dimensions. Additionally, there are many duplicate wave speed data points from overlapping control points of adjacent spectral elements that can be removed, and since the allowable gradient update region was a masked subset region, the domain can be reduced from the original 12 mm × 12 mm region to a 9.4 mm ×

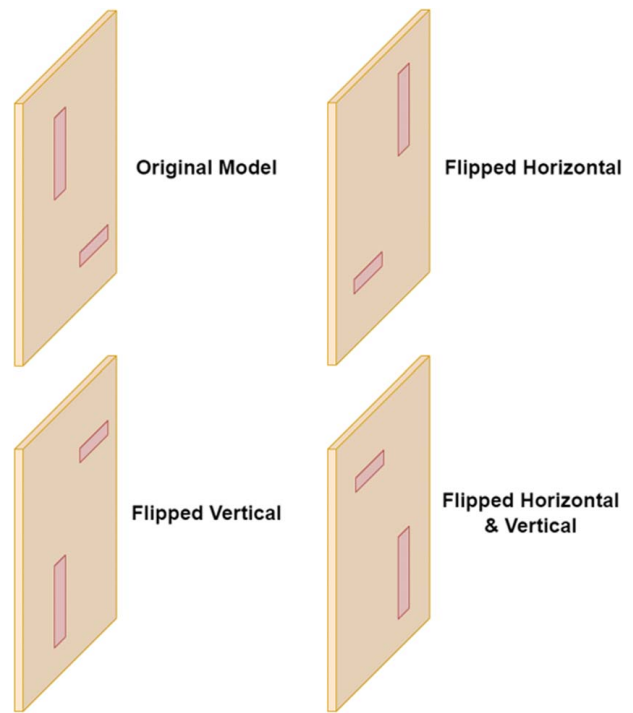


Fig. 4 Augmentation of original dataset to three new samples

9.4 mm region. As in common practice, the wavespeeds were normalized between 0 and 1 using a linear regression normalization technique. Therefore, after normalization, the least wavespeed (the background wavespeed) became 0, and the highest wavespeed in the entire dataset was rescaled to 1. At this point, each individual sample exists as a one-dimensional array with a shape of 1 × 9025. However, each array can be rearranged back into a two-dimensional shape of size 95 × 95. Figure 3 illustrates the process of rearranging the 1D array into the 2D array.

Once converted into the two-dimensional array, we can expand the size of our dataset by augmenting the two-dimensional array by flipping the datasets along its axes (Fig. 4). Due to the symmetry of this surrounding array setup, this augmentation strategy allows us to enhance our original dataset with 3000 additional samples giving us a dataset of 4000 samples, without introducing non-physical artifacts.

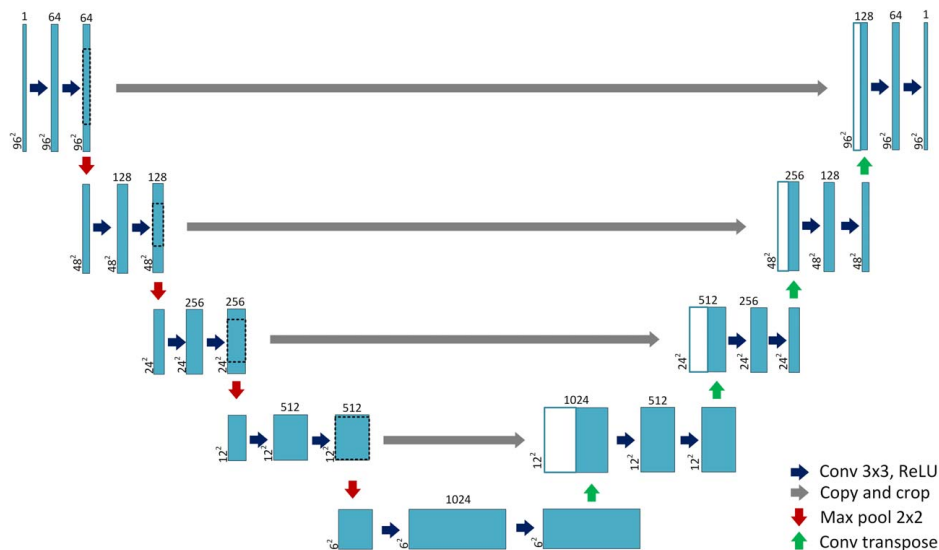


Fig. 5 The implemented 2D U-NET architecture

Table 1 U-NET parameters

Error function	MSE		MAE	
	ReLU	LeakyReLU	ReLU	LeakyReLU
Activation function				
Optimization algorithm	Adam			
Kernel size	3 × 3			
Batch size	32			
Training epochs	25			

Next, all samples are stacked into a single three-dimensional NumPy array of size of $4000 \times 95 \times 95$ pertaining to the now 4000 samples with the 95×95 pixel values making up each sample. Once stacked, the samples are zero-padded along the bottom and right side increasing the dimension to 96×96 to allow easy pooling and scaling of the datasets in the machine learning processes. The deep-learning model processes each sample as a grey-scale image giving us a channel of one (instead of three for red-green-blue images). Hence, the final shape of our dataset is $4000 \times 96 \times 96 \times 1$. The final step in the data preparation process is to randomize and split the dataset into respective training, validation, and testing sets. The training and validation datasets are used to train the CNN and the testing dataset is reserved to test the performance of the trained model. The distribution of the datasets was 80% of the samples for training, 15% for validation, and 5% reserved for testing purposes.

2.4 Architectures. With the original wave speed region and our dataset being a two-dimensional domain, a 2D CNN is well suited for this problem adaptation [48]. We use a CNN because

of its ability to detect and classify features in an image [49], to deblur images [50], and for its ability to increase details through super-resolution [51]. Once trained, the CNN model facilitates the solving of the associated inverse problem and produces a high-resolution image of the predicted model in near real time. Operating in a Linux environment and utilizing the Anaconda 4.10 PYTHON package along with TensorFlow and Keras, a two-dimensional U-shaped CNN (U-NET) model was developed and trained using the single iteration FWI data as its input. While training, the CNN progresses the learning via mapping the prediction and the given ground truth to improve its predictions. During testing of the CNN, only FWIs from the test dataset are used.

The U-NET CNN Fig. 5 consists of an encoder and decoder block configured in a U shape with skip connections connecting encoder/decoder layers. The encoder block works to extract features from the input image and through the layers learns a representation of the image. The encoder block consists of two sets of convolutional layers utilizing batch normalization and an activation function. The skip connection allows the skipping of deeper U-NET layers as a shortcut to the decoder block. After each convolutional layer, a max pooling layer is implemented to reduce the physical dimensions of the input image by half while doubling the number of feature channels. The encoder and decoder blocks are connected with a bridge consisting of another pair of convolutional layers, batch normalization, and activation function combinations. The decoder block then works in reverse of the encoder block by doubling the input image size through a two-dimensional convolutional transpose and concatenating the feature map from the skip connection, and then another pair of convolutional, batch normalization, and activation blocks. The final convolutional layer then reduces the number of channels back to 1 to match the input image channel depth.

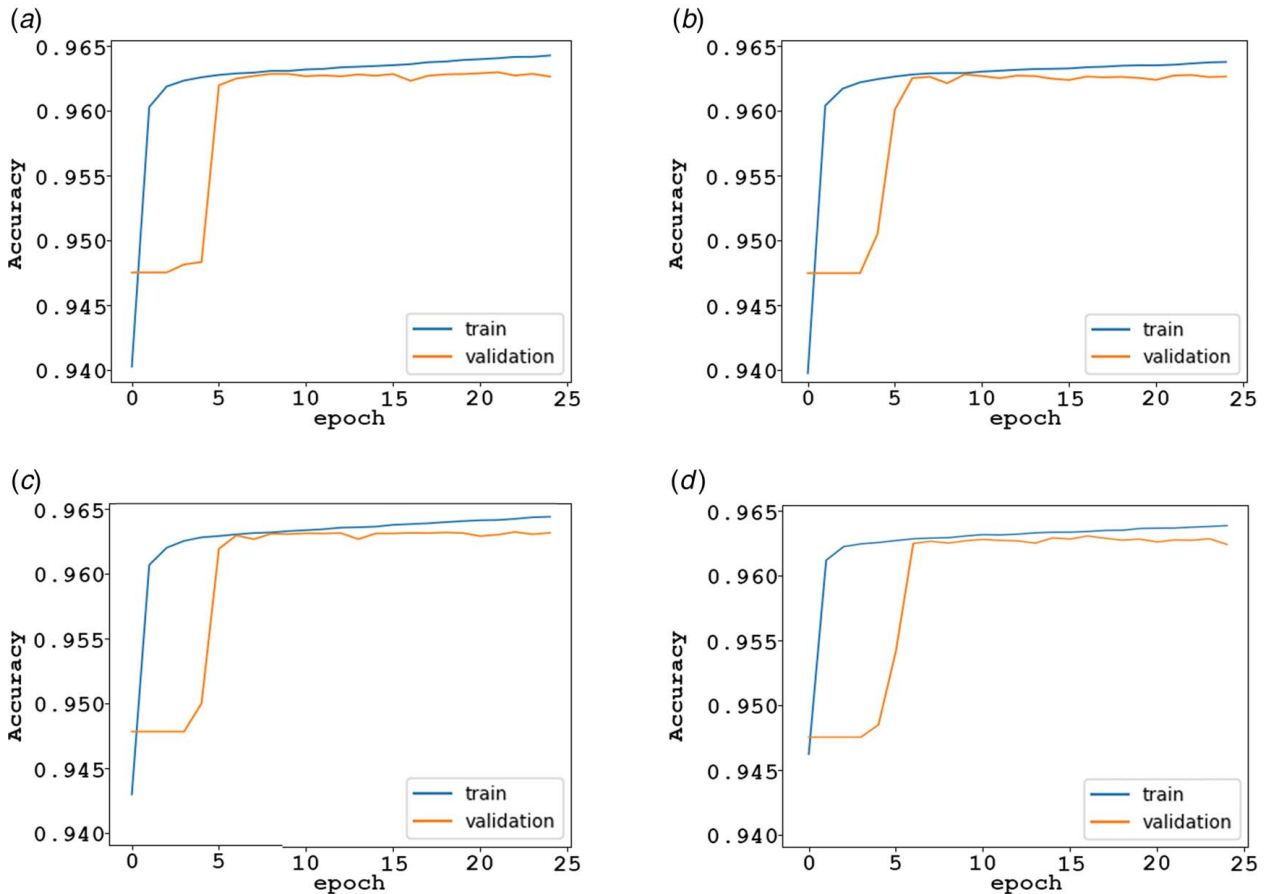


Fig. 6 2D U-NET accuracy comparison: (a) MSE ReLU accuracy, (b) MSE LeakyReLU accuracy, (c) MAE ReLU accuracy, and (d) MAE LeakyReLU accuracy

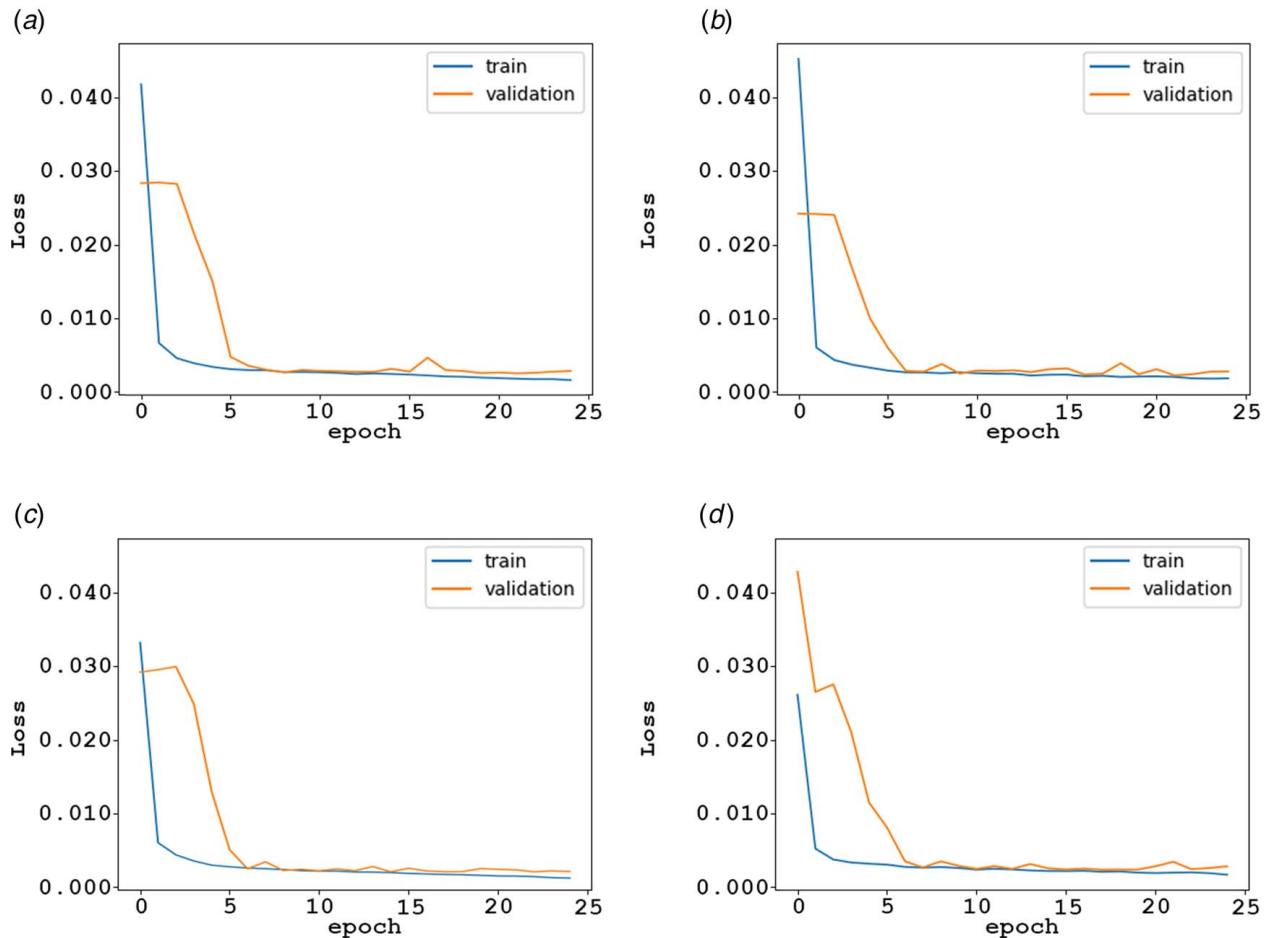


Fig. 7 2D U-NET loss comparison: (a) MSE ReLU loss, (b) MSE LeakyReLU loss, (c) MAE ReLU loss, and (d) MAE LeakyReLU loss

While the architecture of the U-NET is in place, there are still several hyperparameters that can be adjusted to tune the machine learning model. As each of these hyperparameters has its own strengths and weaknesses, it is necessary to test the performance of each hyperparameter and their combination with other hyperparameters. The main parameters considered in our U-NET implementation was the mean squared error (MSE) and mean absolute error (MAE) loss functions as well as the rectified linear unit (ReLU) and leaky rectified linear unit (LeakyReLU) activation functions. While the activation functions are both very similar, they exhibit some key differences in their performance. The ReLU function has the benefit of being fast but can also lead to some neurons being deactivated during training due to a vanishing gradient for negative values. The LeakyRelu activation function does not exhibit the vanishing gradient issue, but its value tends to be less stable [52].

The MSE loss function works by taking the difference between the model's predicted value and the ground truth value, squaring it, and then averaging it across the dataset. As the error is squared in this approach, large errors carry a heavier weight than smaller errors and this approach excels at finding outliers in the dataset.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (4)$$

The MAE loss function works similar to the MSE function but differs in applying an absolute value to the difference of the prediction and ground truth models instead of squaring the difference. The benefit in this approach is that all errors are weighted on a linear scale. Using this approach, the impact of outliers is reduced

during training.

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (5)$$

The next parameter being compared in the U-NET implementation is the activation function which is utilized for the convolutional blocks. The purpose of the activation function is to learn complex non-linear functions. All of our U-NET configurations are based around the ReLU activation function [53]. The ReLU function will return its input value if it is a positive value, and if the input is negative it will return a 0 value. The LeakyReLU activation function works similarly to the ReLU activation function but will remap negative values by multiplying them with a small positive value.

With a total of two loss functions and two activation functions to test, we have a total of four U-NET models to compare. The complete list of parameters and a breakdown of their configurations can be seen in Table 1. All models were trained with 25 epochs and a batch size of 32. Model parameters were updated using the Adam optimizer. All convolutional layers utilized a 3×3 kernel size. Each U-NET model configuration utilized a single activation and error function for the entire model [54]. The four models were then trained on a high-performance cluster using a single GPU node with four NVIDIA Tesla V100-SXM2-32GB graphics cards with a total training time of approximately 4 min per model.

3 Results and Discussion

Once trained, all four of the 2D CNN models were able to quickly predict the ground truth model from a single iteration FWI image.

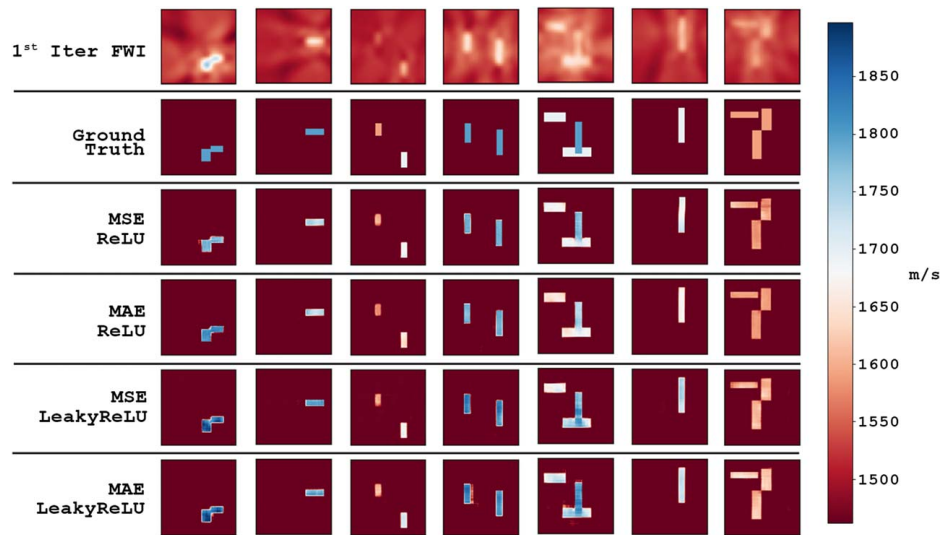


Fig. 8 2D U-NET prediction comparison. First Iter FWI is the FWI input to the CNN. Ground truth is the true model the CNN is attempting to predict. MSE ReLU is the predicted outputs of the U-NET configured with MSE loss and ReLU activation functions. MAE ReLU is the predicted outputs of the U-NET configured with MAE loss and ReLU functions. MSE LeakyReLU is the predicted outputs of the U-NET configured with MSE loss and LeakyReLU functions. MAE LeakyReLU is the predicted outputs of the U-NET configured with MAE loss and LeakyReLU functions. Please view the color image online for accurate velocity identification if viewing in black and white

All configurations of the U-NET were able to converge to greater than 96% accuracy after approximately seven epochs of training (Fig. 6). This means the CNN models were able to correctly predict most of the values and the total required training time is reduced to just over 1 min per model. Additionally, all of the models exhibited similar convergence on the loss values (Fig. 7), which is a measurement of how far from the true value the prediction was. A quantitative review of the model predictions (Fig. 8) shows that some of the more complex regions would cause over or under-prediction of values, particularly in locations where regions overlapped. Despite these challenges, the overall results of the predicted values proved quite favorable.

To quantify the performance of the U-NET models, a total of 200 of the test dataset inversions were predicted by each of the trained models (the results summarized in Table 2). All four of the trained U-NET models proved to perform exceptionally well at predicting the background speed distributions with none of the models exhibiting under-predicting or over-predicting of the background speed. Since the background speed is also the lowest speed in the models, all of the performance measurements of the U-NET configurations in Table 2 are calculated by the percent overshoot or percent undershoot of the maximum wave speed region located in the individual predicted sample.

As can be seen by this summary (and by Fig. 8), both U-NET configurations, when using the LeakyReLU activation function, have a greater tendency to over-predict the maximum wave speed

as compared to their counterpart U-NET models using the ReLU activation function. However, the LeakyReLU activation function also tends to be less likely to under-predict the true wave speed. By comparing the different U-NET configurations in Fig. 8, we can also see that the LeakyReLU activation function tends to perform better at displaying the edges of complex region interactions, particularly in overlapping regions, while the standard ReLU activation function tends to perform better at predicting a uniform internal region wave speed.

Similarly, by comparing the MSE versus MAE loss functions, we can see that the MSE function tends to produce more consistent internal region speed predictions, while the MAE function tends to produce cleaner region edges. We also saw that the MAE loss function was slightly less likely to over-predict wave speeds; however, it was also slightly more prone to under-predicting the speeds when compared to the MSE loss function. We also saw that the MAE loss function in combination with the LeakyReLU activation function was more likely to predict false regions or fuzzy edges of regions.

Each of the U-NET models compared was configured to only use one error function (MSE or MAE) and one activation function (ReLU or LeakyReLU). However, as we can see from the results of the independent models, each of the error functions and activation functions do have certain benefits. A proposed path forward is to leverage a custom error function that combines the MSE and the MAE in a shared manner so that the strengths of each can be utilized. Additionally, since each encoder and decoder block in the U-NET uses a pair of convolutional layers, each with an activation layer, the activation functions could also be split between ReLU and LeakyReLU activations in an attempt to further leverage the strengths of each of the individual activation functions; this includes further tuning the LeakyReLU parameters to tailor its performance to the region identification task it excels at while attempting to reduce its tendency to over-predict wave speed values.

Table 2 Prediction comparison

	ReLU		LeakyReLU	
	MSE	MAE	MSE	MAE
Max over-prediction	6.77%	6.74%	9.71%	10.67%
Max under-prediction	-2.13%	-4.16%	-1.29%	-1.23%
Min over-prediction	0.05%	0.10%	0.09%	0.03%
Min under-prediction	-0.10%	-0.02%	-0.08%	-0.06%
Average over-prediction	2.03%	1.36%	3.58%	3.53%
Average under-prediction	-0.87%	-1.11%	-0.59%	-0.58%

4 Conclusion

Two-dimensional CNN architectures were explored to instantaneously solve the associated FWI problem in USCT. 2D CNNs were found to be able to quickly provide quantitative results after

data acquisition with only a few training epochs required to achieve a high level of accuracy. Various configurations of U-NETs were explored to analyze the strengths and weaknesses of various error and activation functions, whose performance was also qualitatively and quantitatively analyzed. Our studies show that 2D CNNs can be leveraged for solving the inverse problem in FWI-based USCT according to our results, and we suspect that the accuracy and performance of the U-NET can be improved through combinations of error and activation function implementations and will be tested in continued research.

Acknowledgment

We deeply appreciate the financial support provided by NSF projects #2152764 and #2152765, the computational resources provided by ACCESS (formerly XSEDE) grants #MSS190025/#MDE220005, and the high-performance computing technical support for data generation by Dr. Paul Rodriguez through the XSEDE Extended Collaborative Support Service (ECSS) grant. The authors also appreciate the discussions with Mr. Benoît Guillard, Dr. Youzuo Lin, Dr. Mark Anastasio, and Dr. Umberto Villa.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

References

- Fichtner, A., and Trampert, J., 2011, "Resolution Analysis in Full Waveform Inversion," *Geophys. J. Int.*, **187**(3), pp. 1604–1624.
- Warner, M., Ratcliffe, A., Nangoo, T., Morgan, J., Umpleby, A., Shah, N., Vinje, V., Štekl, I., Guasch, L., Win, C., and Conroy, G., 2013, "Anisotropic 3d Full-Waveform Inversion," *Geophysics*, **78**(2), pp. R59–R80.
- Loshelder, A., He, J., Aktharuzzaman, M., Harb, M. S., and Rao, J., 2023, "Apex-shifted Radon Transform for Baseline-subtraction-free (BSF) Damage Scattered Wave Extraction," *Struct. Health Monit.*, **13**.
- Song, G., Li, H., Gajic, Bosko, Zhou, W., Chen, P., and Gu, H., 2013, "Wind turbine blade health monitoring with piezoceramic-based wireless sensor network," *Int. J. Smart Nano Mater*, **4**(3), pp. 150–166.
- Jardani, A., Vu, T., and Fischer, P., 2022, "Use of Convolutional Neural Networks With Encoder–Decoder Structure for Predicting the Inverse Operator in Hydraulic Tomography," *J. Hydrol.*, **604**, p. 127233.
- Li, R., Chen, H., Peng, Y., Li, J., and Li, Y., 2020, "Ultrasound Computed Tomography of Knee Joint," *Chin. J. Electron.*, **29**(4), pp. 705–716.
- Koulountzios, P., Aghajanian, S., Rymarczyk, T., Koironen, T., and Soleimani, M., 2021, "An Ultrasound Tomography Method for Monitoring CO₂ Capture Process Involving Stirring and CaCO₃ Precipitation," *Sensors*, **21**(21), p. 6995.
- He, J., Rao, J., Fleming, J. D., Gharti, H. N., Nguyen, L. T., and Morrison, G., 2021, "Numerical Ultrasonic Full Waveform Inversion (FWI) for Complex Structures in Coupled 2D Solid/Fluid Media," *Smart Mater. Struct.*, **30**(8), p. 085044.
- He, J., Borisov, D., Fleming, J. D., and Kasemer, M., 2022, "Subsurface Polycrystalline Reconstruction Based on Full Waveform Inversion—A 2D Numerical Study," *Materialia*, **24**, p. 101482.
- Prieux, V., Lambaré, G., Operto, S., and Virieux, J., 2013, "Building Starting Models for Full Waveform Inversion From Wide-Aperture Data by Stereotomography," *Geophys. Prospecting*, **61**, pp. 109–137.
- Louboutin, M., Witte, P., Lange, M., Kukreja, N., Luporini, F., Gorman, G., and Herrmann, F. J., 2017, "Full-Waveform Inversion, Part 1: Forward Modeling," *Leading Edge*, **36**(12), pp. 1033–1036.
- Zhang, Z., Wu, Z., Wei, Z., Mei, J., Huang, R., and Wang, P., 2020, "FWI Imaging: Full-Wavefield Imaging Through Full-Waveform Inversion," SEG Technical Program Expanded Abstracts, Online, Oct. 11–16.
- Tromp, J., Tape, C., and Liu, Q., 2005, "Seismic Tomography, Adjoint Methods, Time Reversal and Banana-Doughnut Kernels," *Geophys. J. Int.*, **160**(1), pp. 195–216.
- Lin, Y., Huang, L., and Zhang, Z., 2012, "Ultrasound Waveform Tomography With the Total-Variation Regularization for Detection of Small Breast Tumors," *Medical Imaging 2012: Ultrasonic Imaging, Tomography, and Therapy*, San Diego, CA, Feb. 4–9, Vol. 8320, SPIE, pp. 13–21.
- Robins, T., 2022, "Towards Ultrasound Full-Waveform Inversion in Medical Imaging," PhD thesis, Imperial College London, London.
- Perez-Liva, M., Herraiz, J. L., Urdás, J. M., Cox, B. T., and Treeby, B. E., 2016, "Full-Wave Attenuation Reconstruction in the Time Domain for Ultrasound Computed Tomography," *IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, Prague, Czech Republic, Apr. 13–16, IEEE, pp. 710–713.
- Rao, J., Ratasapp, M., and Fan, Z., 2016, "Guided Wave Tomography Based on Full Waveform Inversion," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, **63**(5), pp. 737–745.
- He, J., Rocha, D. C., and Sava, P., 2020, "Guided Wave Tomography Based on Least-Squares Reverse-Time Migration," *Struct. Health Monit.*, **19**(4), pp. 1237–1249.
- He, J., Rocha, D. C., Leser, P. E., Sava, P., and Leser, W. P., 2019, "Least-Squares Reverse Time Migration (LSRTM) for Damage Imaging Using Lamb Waves," *Smart Mater. Struct.*, **28**(6), p. 065010.
- Van Leeuwen, T., and Herrmann, F. J., 2013, "Mitigating Local Minima in Full-Waveform Inversion by Expanding the Search Space," *Geophys. J. Int.*, **195**(1), pp. 661–667.
- Witte, P., Louboutin, M., Lensink, K., Lange, M., Kukreja, N., Luporini, F., Gorman, G., and Herrmann, F. J., 2018, "Full-Waveform Inversion, Part 3: Optimization," *Leading Edge*, **37**(2), pp. 142–145.
- Yao, G., Wu, D., and Wang, S.-X., 2020, "A Review on Reflection-Waveform Inversion," *Petroleum Sci.*, **17**(2), pp. 334–351.
- Virieux, J., Asnaashari, A., Brossier, R., Métivier, L., Ribodetti, A., and Zhou, W., 2017, "An Introduction to Full Waveform Inversion," *Encyclopedia of Exploration Geophysics*, V. Grechka, and K. Wapenaar, eds., Society of Exploration Geophysicists, Houston, TX, p. R1-1.
- Commer, M., and Newman, G. A., 2008, "New Advances in Three-Dimensional Controlled-Source Electromagnetic Inversion," *Geophys. J. Int.*, **172**(2), pp. 513–535.
- Feng, Y., Chen, Y., Feng, S., Jin, P., Liu, Z., and Lin, Y., 2022, "An Intriguing Property of Geophysics Inversion," [arXiv:2204.13731v2](https://arxiv.org/abs/2204.13731v2).
- Jin, K. H., McCann, M. T., Froustey, E., and Unser, M., 2017, "Deep Convolutional Neural Network for Inverse Problems in Imaging," *IEEE Trans. Image Process.*, **26**(9), pp. 4509–4522.
- Liu, Z., Bicer, T., Kettimuthu, R., Gursoy, D., De Carlo, F., and Foster, I., 2020, "Tomogan: Low-Dose Synchrotron X-Ray Tomography With Generative Adversarial Networks: Discussion," *JOSA A*, **37**(3), pp. 422–434.
- Liu, Z., Sharma, H., Park, J.-S., Kenesei, P., Miceli, A., Almer, J., Kettimuthu, R., and Foster, I., 2022, "Braggnet: Fast X-Ray Bragg Peak Analysis Using Deep Learning," *IUCrJ*, **9**(1), pp. 104–113.
- Wang, Y.-Q., Wang, Q., Lu, W.-K., Ge, Q., and Yan, X.-F., 2022, "Seismic Impedance Inversion Based on Cycle-Consistent Generative Adversarial Network," *Petroleum Sci.*, **19**(1), pp. 147–161.
- Zhang, Z., and Lin, Y., 2020, "Data-Driven Seismic Waveform Inversion: A Study on the Robustness and Generalization," *IEEE Trans. Geosci. Remote Sens.*, **58**(10), p. 10.
- Wang, W., McMechan, G. A., and Ma, J., 2021, "Elastic Isotropic and Anisotropic Full-Waveform Inversions Using Automatic Differentiation for Gradient Calculations in a Framework of Recurrent Neural Networks," *Geophysics*, **86**(6), pp. R795–R810.
- Zhang, T., Sun, J., Innanen, K. A., and Trad, D. O., 2021, "A Recurrent Neural Network for ϵ 1 Anisotropic Viscoelastic Full-Waveform Inversion With High-Order Total Variation Regularization," *First International Meeting for Applied Geoscience & Energy Expanded Abstracts*, Denver, CO, Sept. 26–Oct. 1, pp. 1374–1378.
- Raissi, M., Perdikaris, P., and Karniadakis, G. E., 2017, "Physics Informed Deep Learning (Part I): Data-Driven Solutions of Nonlinear Partial Differential Equations," [arXiv:1711.10561v1](https://arxiv.org/abs/1711.10561v1).
- Song, C., and Alkhalifah, T. A., 2021, "Wavefield Reconstruction Inversion Via Physics-Informed Neural Networks," *IEEE Trans. Geosci. Remote Sens.*, **60**, pp. 1–12.
- Robins, T., Camacho, J., Agudo, O. C., Herraiz, J. L., and Guasch, L., 2021, "Deep-Learning-Driven Full-Waveform Inversion for Ultrasound Breast Imaging," *Sensors*, **21**(13), p. 4570.
- Feigin, M., Makovsky, Y., Freedman, D., and Anthony, B. W., 2020, "High-Frequency Full-Waveform Inversion With Deep Learning for Seismic and Medical Ultrasound Imaging," *SEG Technical Program Expanded Abstracts 2020*, Online, Oct. 11–16, pp. 3492–3496.
- Bader, K. B., Crowe, M. J., Raymond, J. L., and Holland, C. K., 2016, "Effect of Frequency-Dependent Attenuation on Predicted Histotripsy Waveforms in Tissue-Mimicking Phantoms," *Ultrasound Med. Biol.*, **42**(7), pp. 1701–1705.
- Donaldson, R. W., and He, J., 2021, "Instantaneous Ultrasound Computed Tomography Using Deep Convolutional Neural Networks," *SPIE Smart Structures + Nondestructive Evaluation*, Online, Mar. 22–27, Vol. 11593, SPIE, pp. 396–405.
- Prasad, S., and Almekkawy, M., 2022, "DL-UCT: A Deep Learning Framework for Ultrasound Computed Tomography," *IEEE International Symposium on Biomedical Imaging*, Kolkata, India, Mar. 28–31, IEEE, pp. 1–5.
- Bozdağ, E., Peter, D., Lefebvre, M., Komatitsch, D., Tromp, J., Hill, J., Podhorszki, N., and Pugmire, D., 2016, "Global Adjoint Tomography: First-Generation Model," *Geophys. J. Int.*, **207**(3), pp. 1739–1766.
- Zhang, W., Gao, J., Gao, Z., and Chen, H., 2020, "Adjoint-Driven Deep-Learning Seismic Full-Waveform Inversion," *IEEE Trans. Geosci. Remote Sens.*, **59**(10), pp. 8913–8932.

- [42] Tarantola, A., 1984, "Inversion of Seismic Reflection Data in the Acoustic Approximation: Geophysics," *Geophysics*, **49**(8), pp. 1259–1266.
- [43] Bozdağ, E., Trampert, J., and Tromp, J., 2011, "Misfit Functions for Full Waveform Inversion Based on Instantaneous Phase and Envelope Measurements," *Geophys. J. Int.*, **185**(2), pp. 845–870.
- [44] Tromp, J., Komatitsch, D., and Liu, Q., 2008, "Spectral-Element and Adjoint Methods in Seismology," *Commun. Comput. Phys.*, **3**(1), pp. 1–32.
- [45] Modrak, R. T., Borisov, D., Lefebvre, M., and Tromp, J., 2018, "Seisflows-Flexible Waveform Inversion Software," *Comput. Geosci.*, **115**, pp. 88–95.
- [46] Virieux, J., and Operto, S., 2009, "An Overview of Full-Waveform Inversion in Exploration Geophysics," *Geophysics*, **74**(6), pp. WCC1–WCC26.
- [47] Sun, J., Niu, Z., Innanen, K. A., Li, J., and Trad, D. O., 2020, "A Theory-Guided Deep-Learning Formulation and Optimization of Seismic Waveform Inversion," *Geophysics*, **85**(2), pp. R87–R99.
- [48] Arridge, S., and Hauptmann, A., 2020, "Networks for Nonlinear Diffusion Problems in Imaging," *J. Math. Imag. Vis.*, **62**(3), pp. 471–487.
- [49] Eckels, J. D., Fernandez, I. F., Ho, K., Dervilis, N., Jacobson, E. M., and Wachtor, A. J., 2022, "Application of a U-Net Convolutional Neural Network to Ultrasonic Wavefield Measurements for Defect Characterization," *Rotating Machinery, Optical Methods & Scanning LDV Methods*, Vol. 6, D. Di Maio, and J. Bagersad, eds., Springer, Cham, Switzerland, pp. 167–181.
- [50] Tao, X., Gao, H., Shen, X., Wang, J., and Jia, J., 2018, "Scale-Recurrent Network for Deep Image Deblurring," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, June 18–23, pp. 8174–8182.
- [51] Tian, C., Xu, Y., Zuo, W., Zhang, B., Fei, L., and Lin, C.-W., 2020, "Coarse-to-Fine CNN for Image Super-Resolution," *IEEE Trans. Multimedia*, **23**, pp. 1489–1502.
- [52] Xu, B., Wang, N., Chen, T., and Li, M., 2015, "Empirical Evaluation of Rectified Activations in Convolutional Network," *arXiv:1505.00853v2*.
- [53] Dubey, A. K., and Jain, V., 2019, "Comparative Study of Convolution Neural Network's Relu and Leaky-Relu Activation Functions," *Applications of Computing, Automation and Wireless Systems in Electrical Engineering*, S. Mishra, Y. R. Sood, and A. Tomar, eds., Springer, Singapore, pp. 873–880.
- [54] Ronneberger, O., Fischer, P., and Brox, T., 2015, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention – MICCAI*, Munich, Germany, Oct. 5–9, Springer, pp. 234–241.