Geophysical Journal International

Geophys. J. Int. (2025) **240,** 805–828
Advance Access publication 2024 November 15
GJI Seismology

https://doi.org/10.1093/gji/ggae407

Integrating deep learning and discrete cosine transform for surface waves full-waveform inversion

Felipe Rincón [1], Sean Berti,[1,2] Mattia Aleardi,[1] Andrea Tognarelli[1] and Eusebio Stucchi[1]

[1] *Earth Sciences Department, University of Pisa, via S. Maria* 53, 56126 *Pisa, Italy. E-mail: felipe.rincon@phd.unipi.it*
[2] *Earth Sciences Department, University of Florence, via G. la Pira* 4, 50121 *Firenze, Italy*

Accepted 2024 November 13. Received 2024 October 1; in original form 2024 May 3

SUMMARY

Accurate estimations of near-surface *S*-wave velocity (V_s) models hold particular significance in geological and engineering investigations. On the one hand, the popular multichannel analysis of surface waves (MASWs) is limited to the 1-D and the plane wave assumptions. On the other hand, the more advanced and computationally expensive full-waveform inversion (FWI) approach is often solved within a deterministic framework that hampers an accurate uncertainty assessment and makes the final predictions heavily reliant on the starting model. Here we combine deep learning with discrete cosine transform (DCT) to solve the FWI of surface waves and to efficiently estimate the inversion uncertainties. Our neural network approach effectively learns the inverse non-linear mapping between DCT-compressed seismograms and DCT-compressed *S*-velocity models. The incorporation of DCT into the deep learning framework provides several advantages: it notably reduces parameter space dimensionality and alleviates the ill-conditioning of the problem. Additionally, it decreases the complexity of the network architecture and the computational cost for the training phase compared to training in the full domain. A Monte Carlo simulation is also used to propagate the uncertainties from the data to the model space. We first test the implemented inversion method on synthetic data to showcase the generalization capabilities of the trained network and to explore the implications of incorrect noise assumptions in the recorded seismograms and inaccurate wavelet estimations. Further, we demonstrate the applicability of the implemented method to field data. In this case, available borehole information is used to validate our predictions. In both the synthetic and field applications, the predictions provided by the proposed method are compared with those of a deterministic FWI and the outcomes of a network trained in the full data and model spaces. Our experiments confirm that the implemented deep-learning inversion efficiently and successfully solves the FWI problem and yields more accurate and stable results than a network trained without the DCT compression. This opens the possibility to efficiently train a neural network that provides accurate instantaneous predictions of V_s near-surface models and related uncertainties.

Key words: Monte Carlo methods; Neural networks, fuzzy logic; Surface waves and free oscillations; Waveform inversion.

1 INTRODUCTION

In near-surface applications, *S*-wave velocity models (V_s) are commonly obtained through surface waves analysis. In this context, one of the most popular approaches is the multichannel analysis of surface waves (MASW) which exploits the dispersion characteristics of surface waves (Park *et al.* 1999; Xia *et al.* 1999). However, the extraction of the dispersion curve usually depends on the user's interpretation, thus introducing a degree of subjectivity especially when multimodal dispersion patterns are observed.

Numerous efforts have been made to improve the accuracy of surface-wave dispersion analysis (e.g. Forbriger 2003; Kumar & Naskar 2017) and to address the ill-posedness of this inverse problem, such constraining the inversions or using probabilistic approaches (e.g. Xia *et al.* 2003; Bohlen & Saenger 2006; Ivanov *et al.* 2006; Socco & Boiero 2008; Maraschini & Foti 2010; Dal Moro 2019). In addition, the forward operator involved in the computation of the dispersion curves presents two strong assumptions: a 1-D layered model and plane waves. Therefore, a limitation arises concerning the lateral resolution of the obtained 2-D subsurface

models, as the profile length of the analysed seismic arrays cannot be arbitrarily small.

The full-waveform inversion (FWI) is a more advanced, although computationally expensive, approach that utilizes the complete waveform information recorded by seismic sensors to derive a more detailed reconstruction of the subsurface properties. This makes this approach potentially able to address the previously mentioned limitations of the MASW. FWI is traditionally solved through deterministic approaches which seek to find a single best-fitting model that explains the observed data (Tarantola 1984; Virieux 1986; Virieux & Operto 2009). Although this approach is computationally efficient, it heavily depends on a good starting model and fails to accurately estimate the uncertainties. This dependency on the starting model is related to the 'cycle skipping problem,' which occurs when the observed and simulated seismic waveforms are out of phase by more than half a cycle (Schäfer *et al.* 2014; Alkhalifah 2016). As an alternative, a probabilistic approach to FWI can be applied; this framework operates on the foundation of Bayes theorem, integrating prior model information, data uncertainty, and even modelling errors into the posterior probability density (PPD) function, which fully describes the uncertainties affecting the solution. In case of non-linear forward modellings, the PPD is numerically estimated, for example by using Markov Chain Monte Carlo (MCMC) methods (Mosegaard & Tarantola 2002; Gebraad *et al.* 2020). These sampling methods are less dependent on a good starting model and provide an estimation of the uncertainties; however, they entail significant computational expenses (Sajeva *et al.* 2017; Aleardi *et al.* 2019; Zhao & Sen 2021; Berti *et al.* 2024, b).

The rapid advances in algorithms and computing powers constitute an unprecedented opportunity for significant progress in seismic inversion, enabling the solution of previously infeasible problems through data-driven approaches. A promising avenue of research is to approximate the inverse mapping of FWI by training a neural network on pairs of seismic data and corresponding velocity models (Wu *et al.* 2018; Kazei *et al.* 2021; Zhu *et al.* 2021). These approaches seek to leverage the power of deep learning to learn complex relationships between seismic data and subsurface properties, potentially revolutionizing the traditional FWI methodology. However, the efficacy of learning-based methods stems from their ability to leverage vast amounts of high-quality training data, which poses a challenge for seismic methods due to the high acquisition costs and confidentiality concerns.

In geophysics, and especially in seismic inversion, significant research efforts have been devoted to using deep learning methodologies. For example, Richardson (2018) and Mosser *et al.* (2020) used a generative adversarial network (GAN) as a form of regularization to promote plausible velocity models and reduce model parameters by leveraging information from the network's latent space. They then successfully incorporated this approach into a FWI framework. Ovcharenko *et al.* (2019) and Sun & Demanet (2020) applied neural networks to extrapolate the missing low frequencies and help alleviate the cycle-skipping problem. Wu & McMechan (2019) proposed a convolutional neural network (CNN) domain FWI, which uses a CNN to reparametrize the velocity model or gradient field, updating the neural network weights to minimize the loss function. Additionally, the authors provided uncertainty estimations using the Monte Carlo (MC) dropout technique (Gal & Ghahramani 2016; Osband 2016). In a similar vein, Yang & Ma (2019) formulated a UNet-based supervised training approach to approximate the inverse non-linear operator as an alternative to the classic acoustic FWI formulation. However, their study was limited to synthetic data, and they utilized impractical array settings by aligning the number of geophones with the grid size of the velocity model. To address the limitations of their training, they used transfer learning techniques with new training models akin to the simulated ones, underscoring the importance of a wide and diverse training data set in the supervised approach. To assess the generalization capabilities of data-driven methods, Zhang & Lin (2020) analysed the prediction robustness of a broad and diverse data set. However, their study necessitated fine-tuning via transfer learning techniques and was also limited to synthetic examples. Dhara & Sen (2022) demonstrated that combining CNN with a physics-based forward process in elastic FWI can mitigate issues such as local minima and crosstalk effects (defined as the interference arising from the coupling between different model parameters) thereby enhancing inversion accuracy. A more pragmatic approach was introduced by Zhang & Alkhalifah (2022), who integrated well-log information to regularize the inverse problem. Their method addressed the challenge of low parameter illumination based on observed data and demonstrated efficacy across both synthetic and field data scenarios. Zhu *et al.* (2023) implemented FWI using the Fourier neural operator as the decoder of DeepONet (Lu *et al.* 2021), thus improving the generalization and robustness of data-driven machine-learning approaches. So far, the previously mentioned investigations have focused on acoustic FWI, limited to synthetic data, with few of them incorporating uncertainty estimation. In the case of surface waves FWI for near-surface scenarios, Vantassel *et al.* (2022) made significant contributions. While the authors demonstrated the reliability of data-driven approaches to invert surface waves, their study was limited to synthetic data and subsurface models with simple geometries (i.e. specifically two and three-layer models). In addition, they considered just a single shot as the observed data, and no estimation of the model uncertainties was provided. Yust *et al.* (2023) presented a field data application in which the authors investigate the impact of starting models on FWI results. This includes the use of a V_s model predicted by a frequency-velocity CNN that is based on Rayleigh wave dispersions, as detailed in Abbas *et al.* (2023).

In this paper, we introduce a computationally efficient deep-learning approach to solve the FWI of surface waves. One of the main aims of the work was to train a network with good generalization capabilities, and for this reason, a particular effort was devoted to generate a training data set representing realistic geological scenarios for near surface investigations. These models were generated using the GemPy algorithm (de la Varga *et al.* 2019). For what concerns the acquisition layout we considered a common 2-D seismic array configuration for near-surface applications, using setups with 48 receivers and 3 shots to characterize the subsurface over a length of 54 m with a maximum depth of 30 m. We generated the synthetic data using a finite difference (FD) scheme using the SOFI2D algorithm (Bohlen 2002). In this study, we exclusively focus on the inversion of the V_s, as it is the parameter primarily informed by Rayleigh waves (Miller *et al.* 1955; Richart *et al.* 1970). We assume the *P*-wave velocity (V_p) to be twice that of V_s and a constant density of 1.8 g cm^{-3}. We trained the network in the discrete cosine transform (DCT) domain, where the input corresponds to the DCT compression of each shot (aggregated in a 3-D tensor), whereas the output corresponds to the DCT-transformed V_s model. The use of the DCT reduces computational time during the training stage and mitigates the ill-posedness problem inherent in the approach. The implemented deep-learning inversion is also integrated with a MC

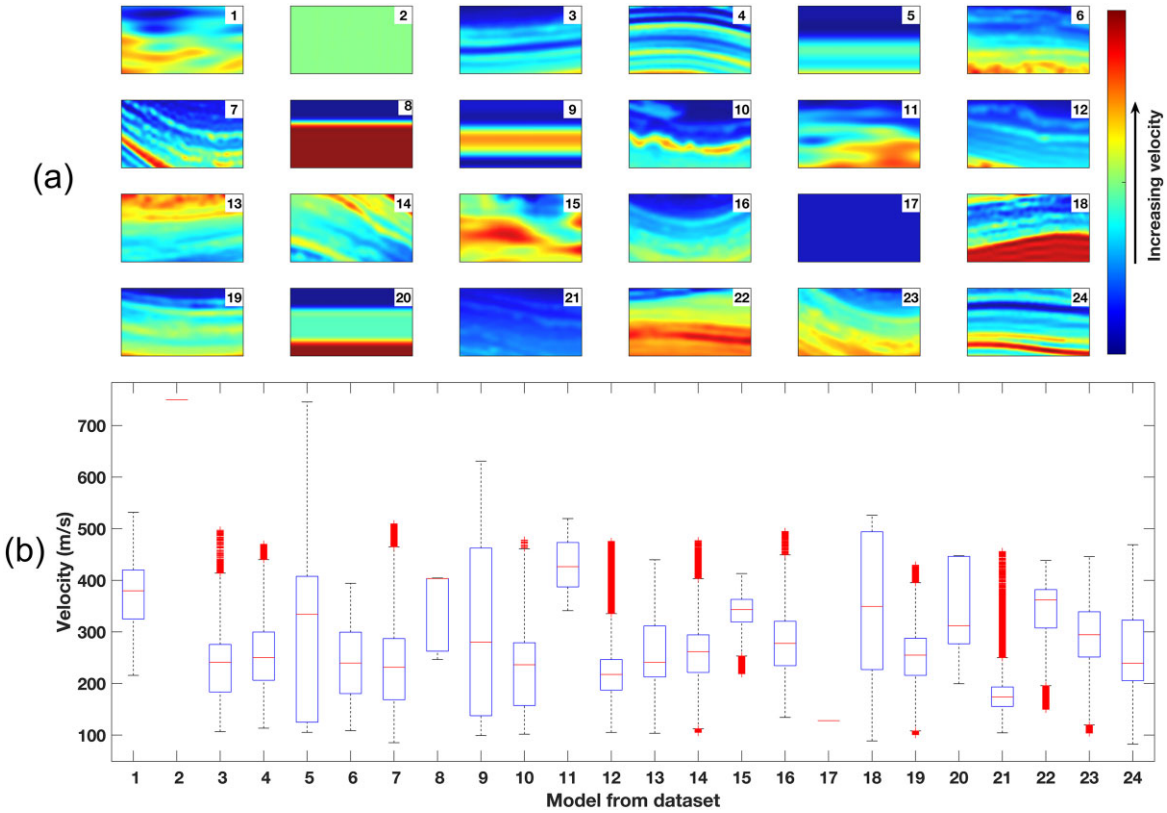

Figure 1. Examples of generated V_s models from the training and validation sets: (a) Velocity models. (b) Box plots representing the velocity range for each model shown in (a). In (b) note the very different velocities ranges spanned by the generated models.

Table 1. Hyperparameters of the forward modelling code.

Forward operator hyperparameters	
Number of gridpoints in (Ny, Nx)	(150, 276)
Distance between gridpoints (DH)	0.25 m
Number of gridpoints in the absorbing frame	30
Peak frequency	25 Hz
Sampling interval (dt)	5e-5 s
Registration length (t)	0.5 s

simulation to propagate the uncertainties from the data to the model space.

We first use synthetic tests to analyse the impact of the number of retained DCT coefficients on the training performance and prediction accuracy, and we also illustrate the results obtained when the network is trained in full-domain (without the DCT compression). Next, we demonstrate the method is fairly robust when the estimated wavelet is not too far from the propagating wavelet. In our final test, we use the predicted model from the neural network as the starting point for a deterministic FWI, and we also compare this result with that obtained when the predictions from a MASW inversion (obtained with the Geopsy algorithms; Wathelet *et al.* 2004; 2020) are used to trigger the deterministic FWI. The deterministic FWI is solved using the IFOS2D algorithm (Bohlen 2002). Finally, we illustrate the applicability of our deep-learning approach to field data. To the best of our knowledge, this study represents the first attempt in which deep learning algorithms, MC simulation, and a compression strategy are combined to efficiently solve the probabilistic FWI of surface waves.

2 METHODS

2.1 Data set generation

For all the experiments in this study, we assume a subsurface area that is 30 m deep and 54 m long, discretized into quadratic cells measuring 0.25 m × 0.25 m. One of the main challenges in supervised deep learning approaches lies in generating a large and geologically plausible data set (Deng *et al.* 2022). To address this challenge, we created a variety of 3-D models with different geometrical relations representing typical geological scenarios for near-surface applications, including stratigraphic sequences, faults, sinkholes, landslides, fills, etc. To generate these models, we utilized the open-source algorithm GemPy, which performs implicit geological modelling (de la Varga *et al.* 2019). For each geological scenario, we systematically varied some conditions, including the number of layers and simulated different orientations of lithology contacts. Subsequently, we randomly extracted 2-D vertical slices from various directions of the generated 3-D models. Finally, we rescale the *S*-wave velocity in a physically plausible range for near-surface investigations (from 100 to 1200 m s^{-1}).

Since we are primarily interested in the estimation of the V_s model, we simply set the V_p as twice V_s, while density is fixed at 1.8 g cm^{-3}. Using the previously described strategy, we generated 1000 3-D models, from which we extracted 7000 2-D subsurface elastic models for the training data set. This process was repeated across 100 3-D models to generate 500 2-D models for the validation data set, and 20 additional 3-D models were used to extract 100 2-D models for the testing data set. This approach enables the generation of an ensemble of almost independent V_s models that realistically

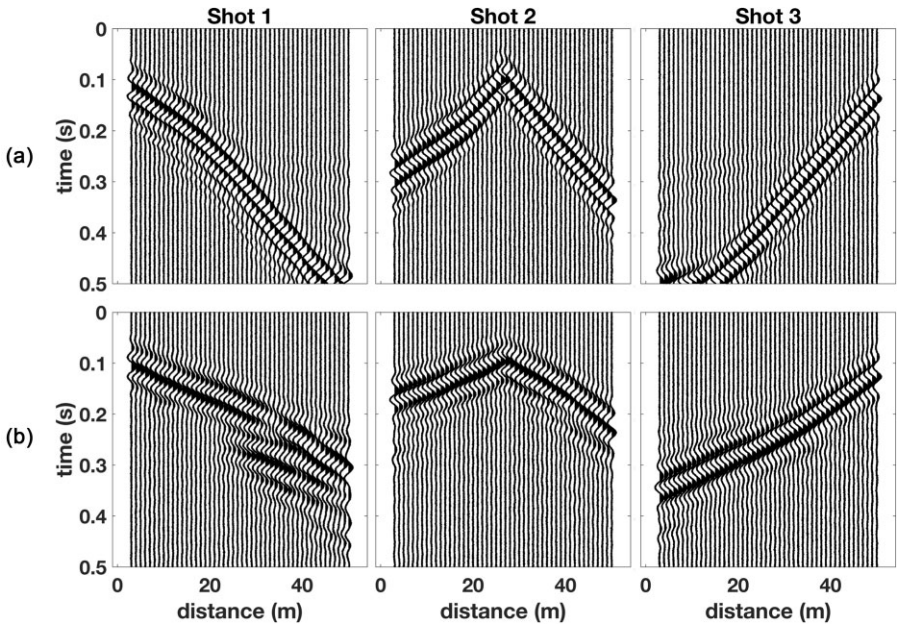

Figure 2. Examples of three shot gathers from Model 10 (a) and Model 18 (b) (previously shown in Fig. 1a) generated using the Morlet wavelet.

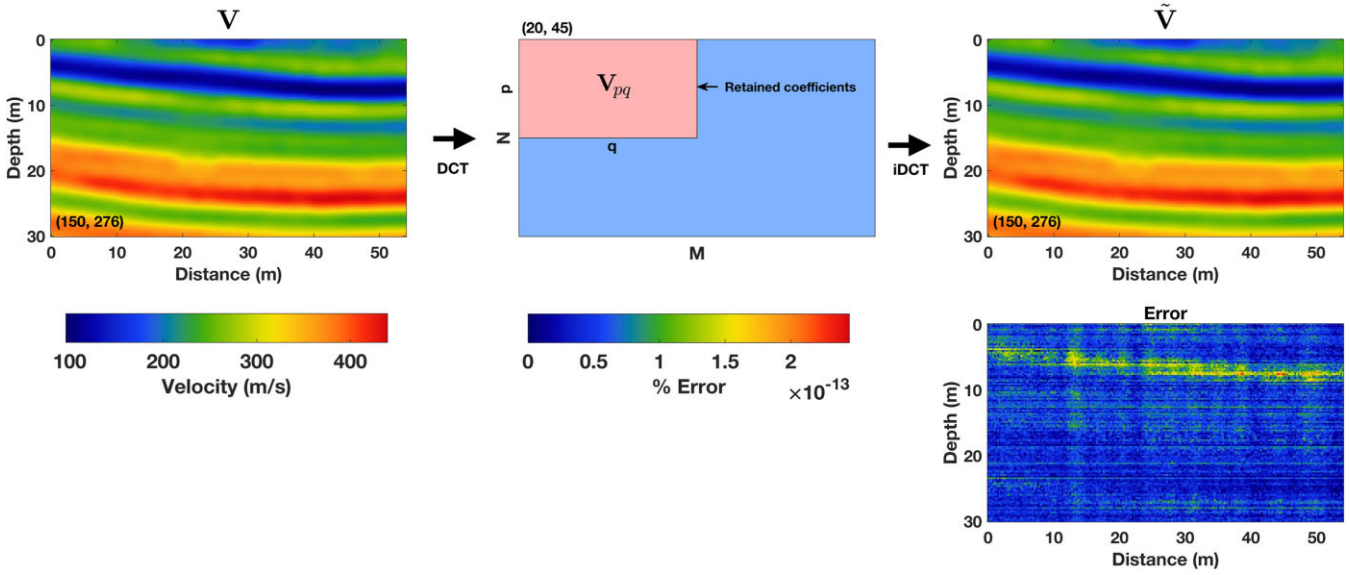

Figure 3. Example of DCT compression and associated error in the approximated velocity model. The velocity model size is (150, 276), and only (20, 45) DCT coefficients are used for compression. iDCT denotes the inverse DCT.

mimic subsurface scenarios. Fig. 1 shows examples of generated 2-D V_s profiles.

To generate the seismic data, we used the SOFI2D algorithm (Bohlen 2002). In this algorithm a FD scheme is used to simulate the seismic wave propagation in 2-D elastic media. The accuracy of the FD-simulated data has been validated, for 1-D models, with the outcomes of a reflectivity method (Schmidt & Jensen 1985). In this case, for a more quantitative comparison, the 3-D–2-D correction (Forbriger *et al.* 2014; Schäfer *et al.* 2014) has been applied to the seismograms computed via the reflectivity method.

The acquisition array consists of 48 receivers separated by 1 m, and 3 sources (two off-end shots and one split-spread). To assess the network's robustness when the source wavelet is unknown, we created three independent data sets using a Ricker, a Morlet and a wavelet estimated from field data. Both the Ricker and Morlet wavelets were computed with a 25 Hz peak frequency to match the frequency content present in the field data. Independent training was performed for each data set.

It is also important to note that during the creation of the seismic data, we simulated a vertical force and considered the vertical component of the receivers. Additionally, we incorporated Gaussian uncorrelated noise into each computed seismogram, with a standard deviation equal to the 5 per cent of that of the noise-free data. This specific value was directly derived from the field data, just by assuming simple Gaussian distributed noise and by computing the standard deviation of repeated shots acquired in the field. Table 1 summarizes the forward hyperparameters used for computing the

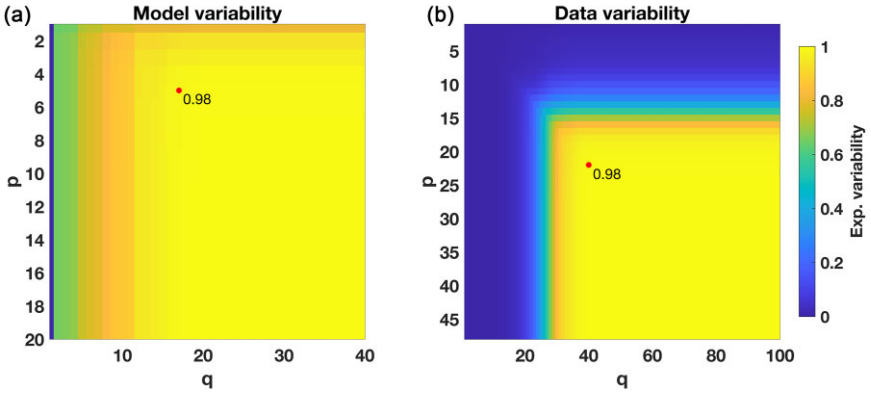

Figure 4. Examples of the explained variability calculated from the V_s model shown in Fig. 3. Panel (a) illustrates the model variability, while panel (b) shows the variability of the associated data. In both cases, the (p, q) coordinates represent the number of retained coefficients along the 2 DCT dimensions while the colormap codes the corresponding preserved variability.

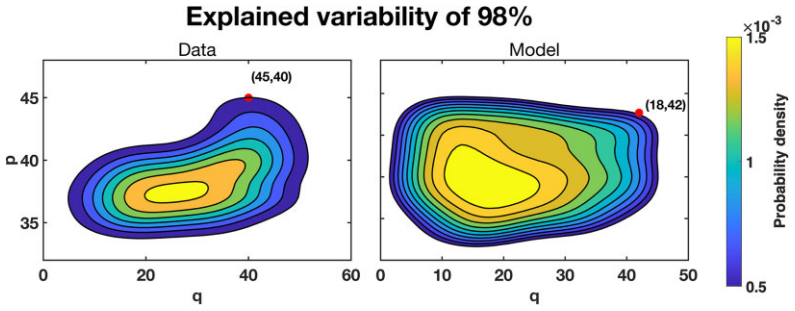

Figure 5. Probability density maps representing, for all generated data and model pairs in the training set, the minimum number of DCT coefficients required to explain 98 per cent of the total variability.

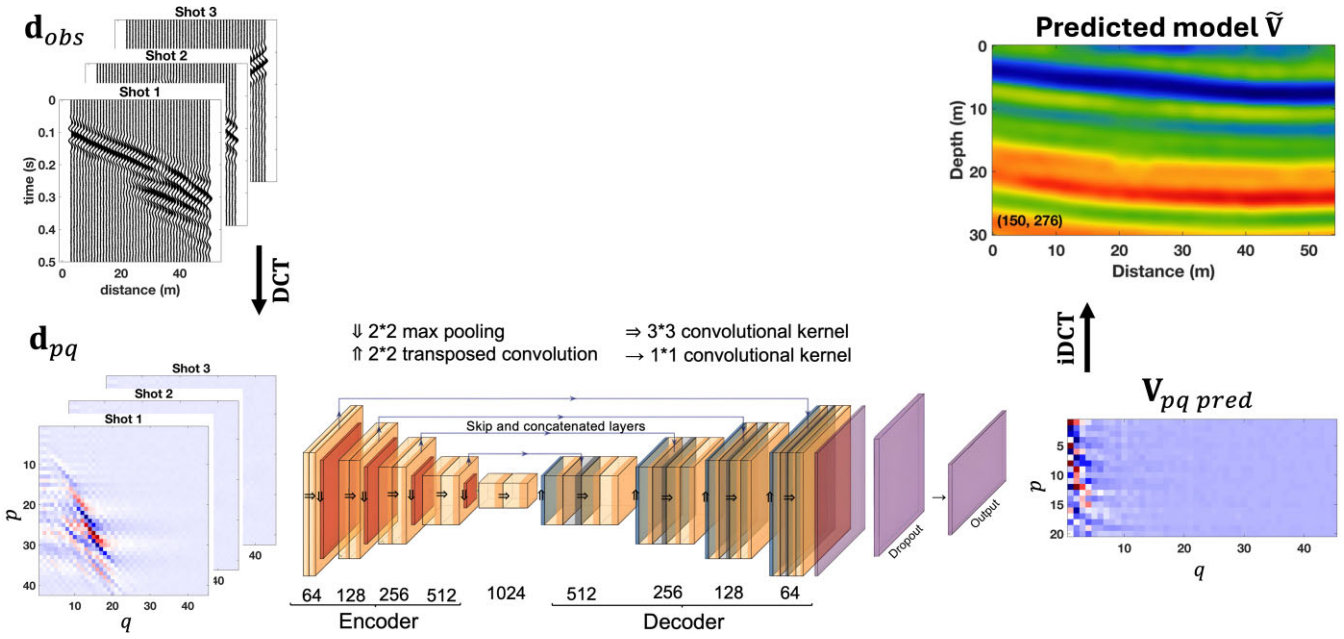

Figure 6. Schematic representation of the deep learning inversion framework. The symbol ⇒ and ⇑ express the 2-D convolution and transposed convolution operations, the term ⇓ represents the max pooling operations, and → denotes the 1-D convolution.

data. In the table, 'PML' refers to the Perfectly Matched Layer technique used as the artificial absorbing layer. It is worth noting that we utilized the same hyperparameters to create three independent data sets, each varying the wavelet. Fig. 2 illustrates two examples of the computed data using models 10 and 18 from Fig. 1(a). Computing the data for all the 7600 V_s realizations took 1.9 hr using a parallel MATLAB code running on a cluster powered by an Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz, utilizing 10 cores.

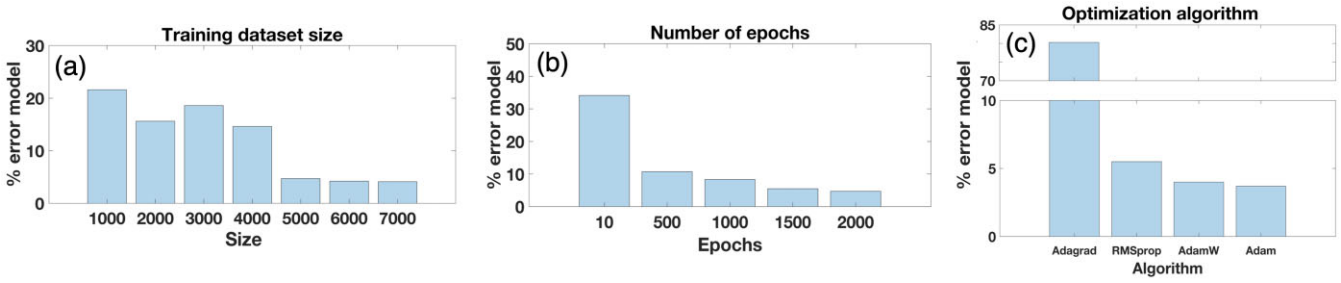

Figure 7. Model percentage error computed when systematically varying the network hyperparameters: (a) size of the training data set; (b) number of epochs and (c) optimization algorithm.

Table 2. Selected hyperparameters for the network training.

Neural network hyperparameters	
Training data set size	5000
Number of epochs	1500
Optimization algorithm	Adam
Number of filters	[64, 128, 256, 512, 1028]
Number of convolutional layers	24
Learning rate	0.001
Batchsize	10

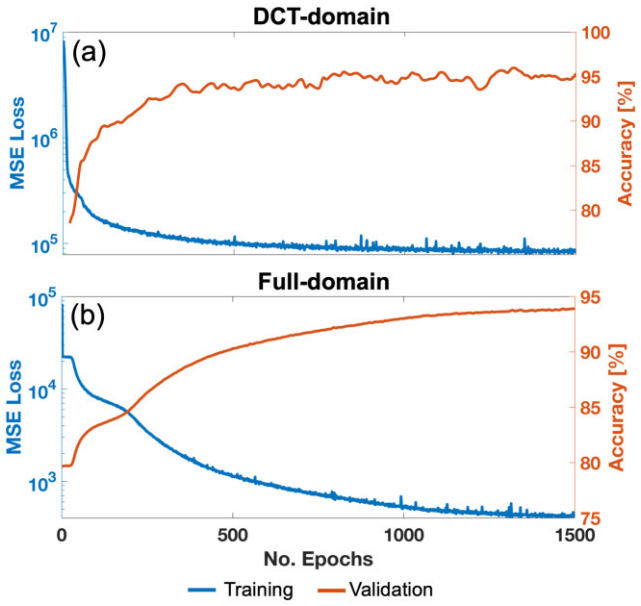

Figure 8. Evolution of Loss function and accuracy for the training and validation sets, respectively. In (a) DCT domain and (b) in the full-domain.

2.2 Discrete cosine transform

In this work, the DCT (in its 2-D formulation) is used for both data and model compression. This technique is effectively used in digital signal processing and data compression, and it allows the transformation of a multidimensional signal from the spatial domain to the frequency domain. Like other Fourier-related transforms, the DCT expresses a signal as a sum of cosine functions with varying frequencies and amplitudes (Britanak *et al.* 2010). As an example, consider a 2-D velocity model, $\mathbf{V}(x, y)$, where $x = [0, 1, \ldots M-1]$ and $y = [0, 1, \ldots N-1]$ denote the horizontal and vertical coordinates, respectively. The corresponding 2-D DCT projection ($\mathbf{V}_{dct}$) is analytically defined as follows:

$$\begin{cases} \mathbf{V}_{dct}\left(\boldsymbol{K}_x, \boldsymbol{K}_y\right) = \sqrt{\frac{1}{M}}\cdot\sqrt{\frac{1}{N}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\mathbf{V}(x,\ y)\ , \text{ if } \boldsymbol{K}_x = \boldsymbol{K}_y = 0 \\ \mathbf{V}_{dct}\left(\boldsymbol{K}_x, \boldsymbol{K}_y\right) = \sqrt{\frac{2}{M}}\sqrt{\frac{2}{N}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\mathbf{V}(x,\ y)\cos\left(\frac{(2\,x+1)\,\pi\ \boldsymbol{K}_x}{2\,M}\right) \\ \cos\left(\frac{(2\,y+1)\,\pi\ \boldsymbol{K}_y}{2\,N}\right),\ \text{if } \boldsymbol{K}_x,\ \boldsymbol{K}_y \neq 0 \end{cases}, \quad (1)$$

where the matrix $\mathbf{V}_{dct}(\boldsymbol{K}_x, \boldsymbol{K}_y)$ represents the k_xth and k_yth DCT coefficients. By defining the DCT basis functions as two matrices $\boldsymbol{B}_x$ and $\boldsymbol{B}_y$ of dimensions $\boldsymbol{M}$ x $\boldsymbol{M}$ and $\boldsymbol{N}$ x $\boldsymbol{N}$, respectively, we can compactly rearrange the eq. (1) in matrix form as:

$$\mathbf{V}_{dct} = \boldsymbol{B}_y\,\mathbf{V}\,\boldsymbol{B}_x^T, \quad (2)$$

DCT efficiently concentrates most of the energy distribution of the signal in the low-order coefficients, thereby enabling its representation with a reduced number of coefficients, typically with dimensions ($p < N, q < M$). The scalar pair (p, q) represents the retained number of basis-functions along the y and x direction used to compress the velocity model, while all the other DCT coefficients are set to zero for the compression. In this way, after retaining only the coefficients (p, q), we can obtain an approximated version of $\mathbf{V}$, by performing the inverse of the DCT (iDCT) as follows:

$$\tilde{\mathrm{V}} = \left(\boldsymbol{B}_y^p\right)^T\mathbf{V}_{pq}\,\boldsymbol{B}_x^q, \quad (3)$$

From now on, a variable followed by the subscripts 'pq' indicates the retained coefficients in the compressed space. Instead, the symbol~denotes the approximated variable after performing the inverse DCT (iDCT) with the considered coefficients. For instance, in eq. (3), $\mathbf{V}_{pq}$ corresponds to the matrix with p rows and q columns expressing the retained DCT coefficients, whereas $\tilde{\mathrm{V}}$ denotes the approximated velocity after performing the iDCT. In this way, the DCT transformation effectively reduces the number of parameters from an $(M \times N)$-D full velocity domain to the $(p \times q)$-D DCT-compressed space. Fig. 3 illustrates an example of the DCT compression effect on a velocity model $\mathbf{V}$ when recovering its approximated version $\tilde{\mathrm{V}}$, and the associated error. Notably, only $(p = 20,\ q = 45)$ coefficients guarantee a satisfactory reconstruction of the original $\mathbf{V}$ model in which all the major velocity contrasts are preserved, with negligible approximation error. This procedure is also applied to compress the observed shot gathers and in what follows, we refer to the observed data in the full domain as $\mathbf{d}_{obs}$, whereas $\tilde{\mathbf{d}}_{\mathbf{obs}}$ represents the associated approximation after performing the DCT. In particular, the 2-D DCT is independently applied to each seismic gather under consideration, and subsequently, all compressed gathers are merged in a 3-D tensor. Therefore, in our study, the network receives as input a tensor representing the DCT-compressed data (with the third dimension denoting the number of

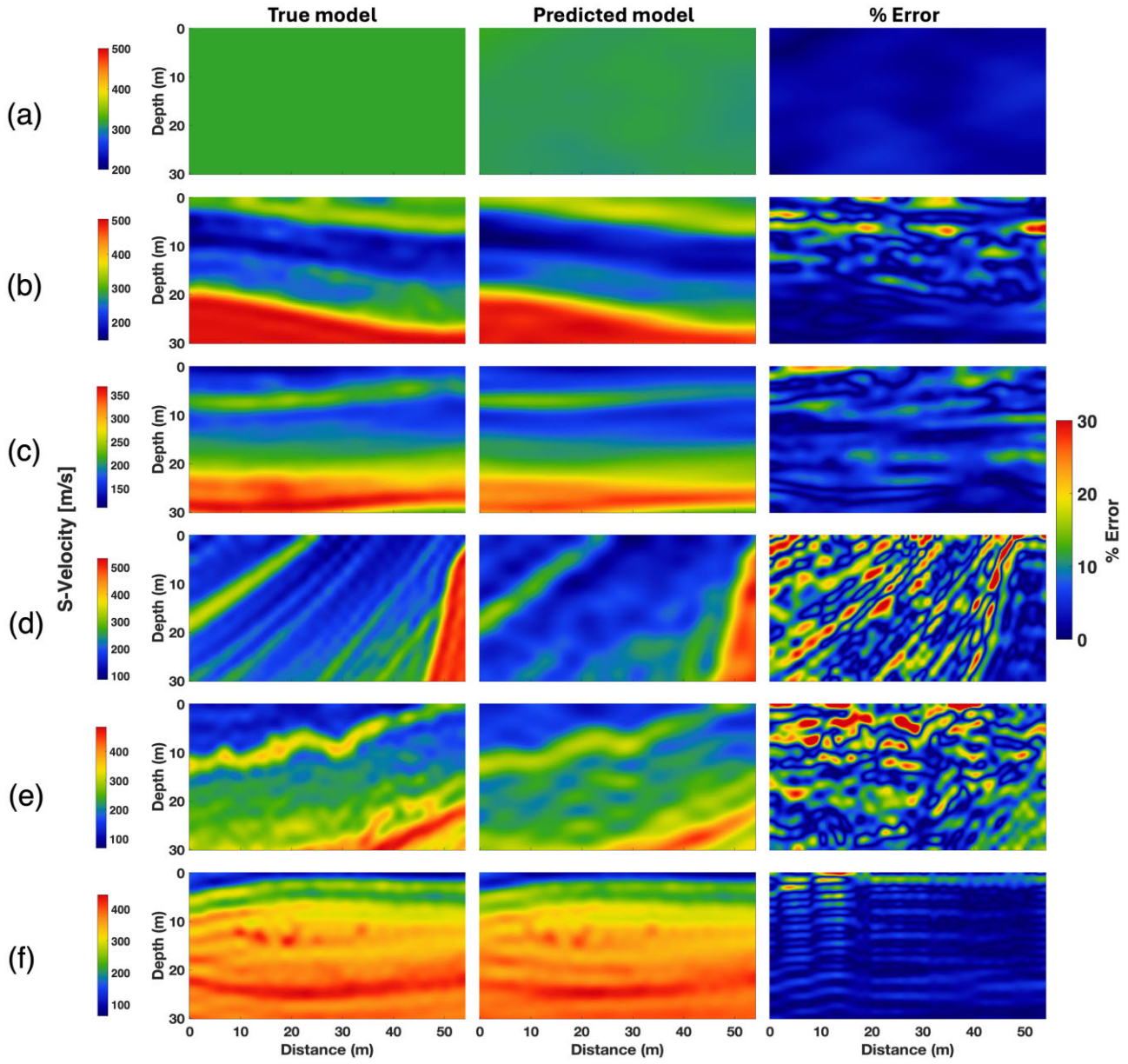

Figure 9. Comparison of predicted and true models during the testing phase, including the resulting percentage error.

shots), while the output consists of a matrix representing the DCT coefficients of the estimated model.

While other compression techniques are viable (i.e. wavelet transforms and/or machine learning techniques; Pang *et al.* 2020; Aleardi *et al.* 2022), in our application, we opted for the DCT because it is a simple, fast, analytical, and stable method with both unique forward and inverse transformations. This compression algorithm was also successfully used by other authors (i.e. Lochbühler *et al.* 2014; Aleardi *et al.* 2020; Kotsi *et al.* 2020). Utilizing overly complex parametrizations can unnecessarily increase computational costs and complicate the inverse mapping from the data to the model space.

As the data set contains velocity models significantly different from one another, we require a strategy to estimate the optimal (p, q) retained coefficients that not only efficiently compress the data set but also mitigate the loss of information. To achieve this, we analysed the explained variability (EV) for all the models and data from the training data set. EV is calculated as the ratio between the variance of the compressed model and data to that of the original model and data, respectively (Aleardi *et al.* 2020). Fig. 4 illustrates an example of the model and data variability for the velocity model presented in Fig. 3. The results suggest that in this example, the combinations (23, 40) and (5, 16) are the minimum pairs of (p, q) coefficients that account for 98 per cent of the total data and model variability, respectively, which is the value that we considered as optimal in our applications (see discussion below and in the chapter dedicated to the synthetic inversion).

Fig. 5 represents the probability density function associated to the number of DCT coefficients needed to represent the 98 per cent variability for models and seismic data in the training set. This diagram helped us in the selection of the number of coefficients we retained in the model and data spaces. We can see that the number of retained coefficients (i.e. (18, 42) for the model and (45, 40) for the data) allows to preserve 98 per cent of variability for

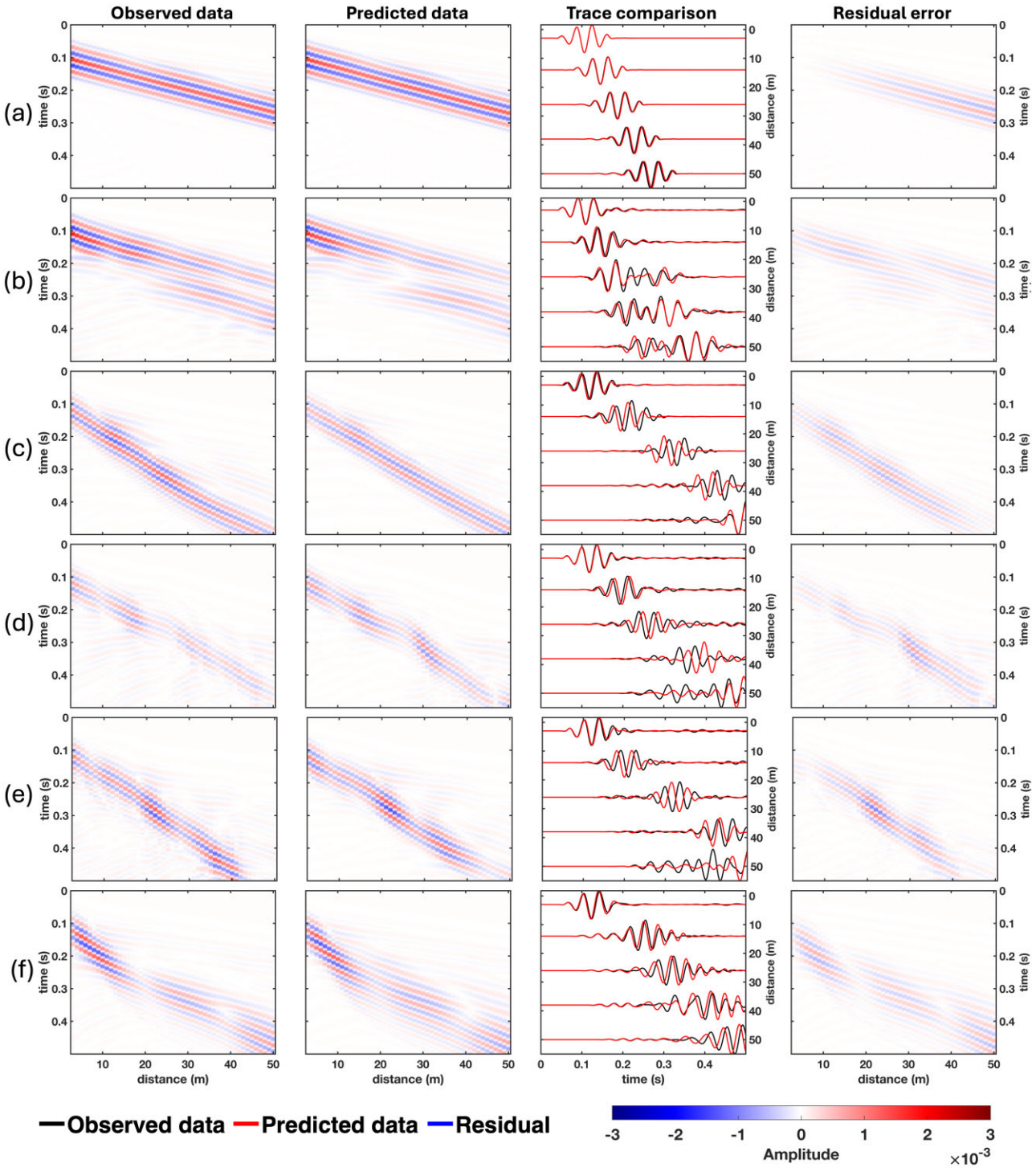

Figure 10. Comparison of the observed and predicted data (generated on the predictions of Fig. 9) along with their sample-by-sample difference. The traces are represented with raw amplitudes without any normalization applied.

most models and data in the training sets, and it is chosen such that the spatial resolution in the estimated model is reasonably preserved without excessively increasing the size of the input and output of the network and hence the computational cost of the training phase (see the discussion in the section dedicated to the synthetic inversion).

We now bring the attention of the reader to another practical benefit of training a neural network in the DCT domain. Usually, the input and output of a CNN are in the same domain to establish a pixel-to-pixel relationship (Long et al 2015). In our application, the input and output of the network are the seismic gathers and V_s model, respectively, each one with very different dimensions: (2000, 48, 3) for the data and (150, 276) for the velocity model. Training in the DCT domain conveniently allows us to match the input and output shapes to facilitate the network operations without the need to adjust the architecture (Ronneberger *et al*. 2015; Dumoulin & Visin 2016). In this way, we can modify the number of

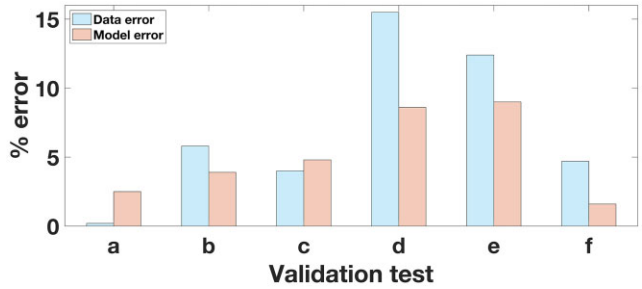

Figure 11. Model and data percentage errors for the different considered models when the wavelet is known.

coefficients to both preserve 98 per cent of the total variability while matching the input and output dimensions. Specifically, we utilized 42 instead of 40 coefficients along the second dimension to match the input shape (45, 42, 3) and the output shape (18, 42). By using the selected retained coefficients, the DCT effectively reduces the data space from 288000-D to 5670-D, whereas the 41400-D model space is compressed to a 756-D domain. Therefore, incorporating the DCT in the neural network, decreases the computational costs for the training phase and the complexity of the neural network architecture needed to get a satisfactory prediction of the inverse operator (see discussion below and Aleardi 2020). Finally, the DCT significantly reduces the parameter space dimensionality, thus helping to mitigate the ill-conditioning of the problem (Aleardi 2020; Berti *et al.* 2024c).

2.3 The implemented DCT-deep learning inversion

We can represent the observed seismic data as $\mathbf{d}_{obs} = G(\mathbf{V}, \mathbf{s})$, where $\mathbf{V}$ denotes the subsurface model properties, $\mathbf{s}$ represents the seismic sources and G is the forward operator, which maps the model and source tensors to the seismic wavefield tensor. In the deterministic approach to FWI, we seek to find a single best-fitting model that minimizes a particular objective function (e.g. L2-norm misfit between observed and predicted seismic gathers). In this context, the model solution can be expressed as $\mathbf{V} = G^{-1}(\mathbf{s},\ \mathbf{d}_{obs})$. However, the mathematical properties of G, noisy observed data, specific model parametrizations, and other contributing factors, render FWI a highly ill-posed inverse problem. In our work, we propose an alternative formulation of the FWI in which a network is trained to approximate the non-linear mapping from the DCT-compressed seismic data to the DCT-compressed V_s model. In this formulation, the inverse solution can be expressed as:

$$\mathbf{V}_{pq\ pred} = Net\left(\mathbf{d}_{pq}; \varphi\right), \tag{4}$$

where $Net(.)$ represents the neural network and φ denotes the set of trainable weights (W) and biases (b) forming the network model. W and b are updated according to the solution of the optimization problem:

$$\hat{\varphi} = arg\ \min_{\varphi}\ \frac{1}{\mathbf{N} * \mathbf{C}} \sum_{n=1}^{\mathbf{N}} \mathbf{L}\left(\mathbf{V}_{pq}, Net\left(\mathbf{d}_{pq,n}; \varphi\right)\right), \tag{5}$$

where $\hat{\varphi}$ denotes the updated set of (W, b); C represents the number of coefficients of the model in the DCT domain ($p \times q$); N denotes the data set size, while n corresponds to each data and model pair in the data set and $\mathbf{L}(.)$ is the loss function that we compute as the L2-norm difference between the true and the predicted model in DCT domain ($\mathbf{V}_{pq}$ and $\mathbf{V}_{pq\ pred}$, respectively). Commonly, CNNs consist of a series of network layers, each performing a differentiable mathematical operation. We used a symmetric encoder-decoder block architecture with 24 convolutional layers and skip connections, as described by Ronneberger *et al.* (2015). The last two layers correspond to a dropout layer followed by a 1-D convolutional layer. Incorporating dropout layers has been proven to improve the accuracy in tasks like visual relocalization and semantic segmentation (Kendall *et al.* 2017). The encoder block gradually reduces the spatial dimensions of the input while increasing the number of feature maps using max-pooling operations. The decoder relies on transposed convolution operations to upsample the feature maps back to the original spatial dimensions while reducing the number of channels. These operations are accompanied by skip and concatenated layers, which are established between corresponding encoder and decoder layers to help preserve spatial information during upsampling. The last layer consists of a 1×1 convolutional layer that maps the features onto the output space. Fig. 6 describes the implemented inversion scheme, together with the adopted CNN architecture.

2.4 Uncertainty estimation

To assess the uncertainty affecting the estimated solution, we projected the noise affecting the seismic data onto the model space using a MC simulation (Hansen & Cordua 2017; Aleardi *et al.* 2021). The aim is to get an estimate of the so-called posterior uncertainties that fully characterize the ambiguities in the recovered solution. The flowchart of the MC strategy is the following:

(i) Use the trained network to derive the DCT-compressed predicted velocity model from the compressed observed data. $\mathbf{V}_{pq\ pred} = Net(\mathbf{d}_{pq}; \varphi)$, then perform the iDCT to obtain $\mathbf{V}_{pred}$ in full-domain.

(ii) Compute the noise-free predicted data $\mathbf{d}_{pred}$, from the predicted model.

(iii) Estimate the statistical properties of the noise $\mathbf{n}$, contaminating the data.

(iv) Sample S models such as:

For $k\ =\ 1$ to S

(i) Perturb $\mathbf{d}_{pred}$ as: $\mathbf{d}_{pert,\ k} = \mathbf{d}_{pred} + \mathbf{n}_k$, where $\mathbf{n}_k$ is a noise realization randomly generated from the assumed noise distribution. In this work, we assume Gaussian uncorrelated noise contaminating the data, hence $p(d) = \mathcal{N}(0, \mathbf{C}_{\boldsymbol{n}})$; where $\mathcal{N}$ represents the Gaussian probability density function, and $\mathbf{C}_n$ is the covariance matrix of the noise.

(ii) Predict a model using the perturbed data $\mathbf{V}_{pq\ pert,k} = Net(\mathbf{d}_{pq\ pert,k}; \varphi)$, and perform the inverse DCT to obtain $\mathbf{V}_{pred,k}$ in full-domain.

(iii) Store the ensemble of models $\mathbf{V}_e = \{\mathbf{V}_{pred,1}, \mathbf{V}_{pred,2} \ldots\ \mathbf{V}_{pred,k}\}$.

(iv) Numerically estimate the statistical properties of the posterior distribution (e.g. mean, standard deviation, percentiles, etc.) from the ensemble of generated models.

Each model in the ensemble $\mathbf{V}_e$ can be considered as a random realization drawn from the posterior and in agreement with the trained network, the assumed noise, and the observed data. For simplicity, we assume uncorrelated Gaussian noise, but the MC approach can handle whatever parametric or non-parametric error distribution. If needed, the same approach can be used to also propagate the modelling error introduced by the network approximation. Even

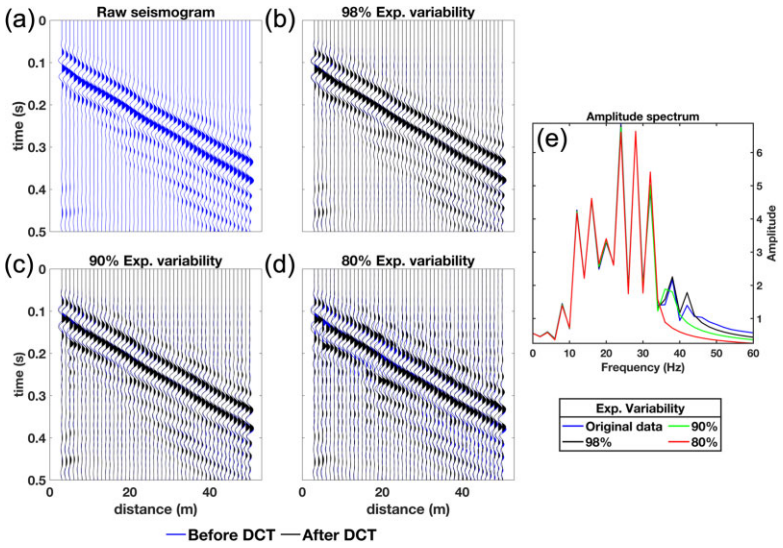

Figure 12. Effect of the DCT compression when explaining the 98, 90 and 80 per cent of the data variability. In (a) the raw seismogram; in (b), (c) and (d) the observed seismogram superimposed to the gathers after compression and in (e) a comparison of the amplitude spectra.

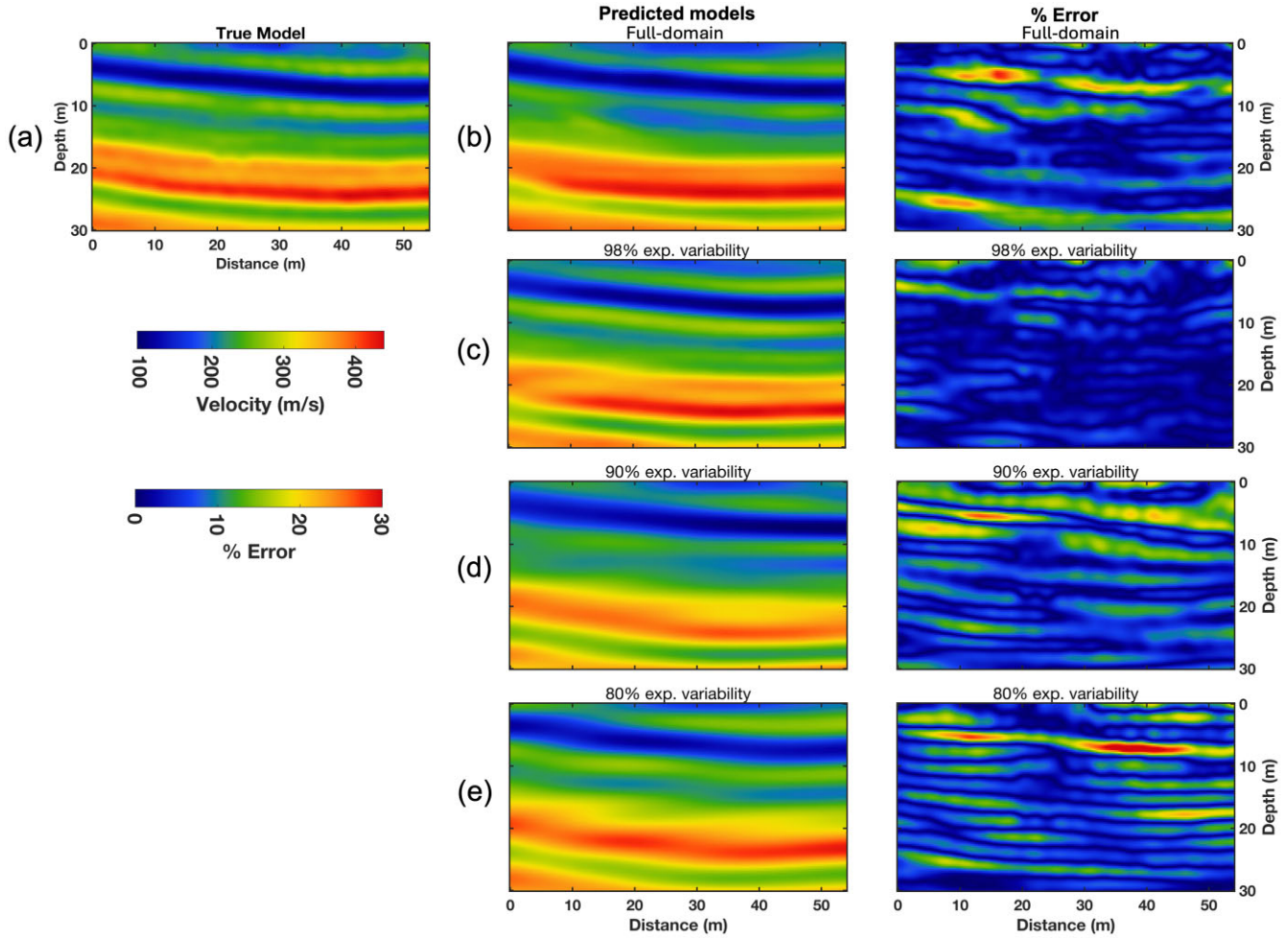

Figure 13. Comparison of predicted models when training in full-domain and in DCT domain with different EV values. In (a), the true model; in (b) the predicted model and its associated percentage error when the network is trained in full-domain; in (c), (d) and (e), the predicted model and error when the network is trained in DCT domain with EV equal to 98, 90 and 80 per cent, respectively.

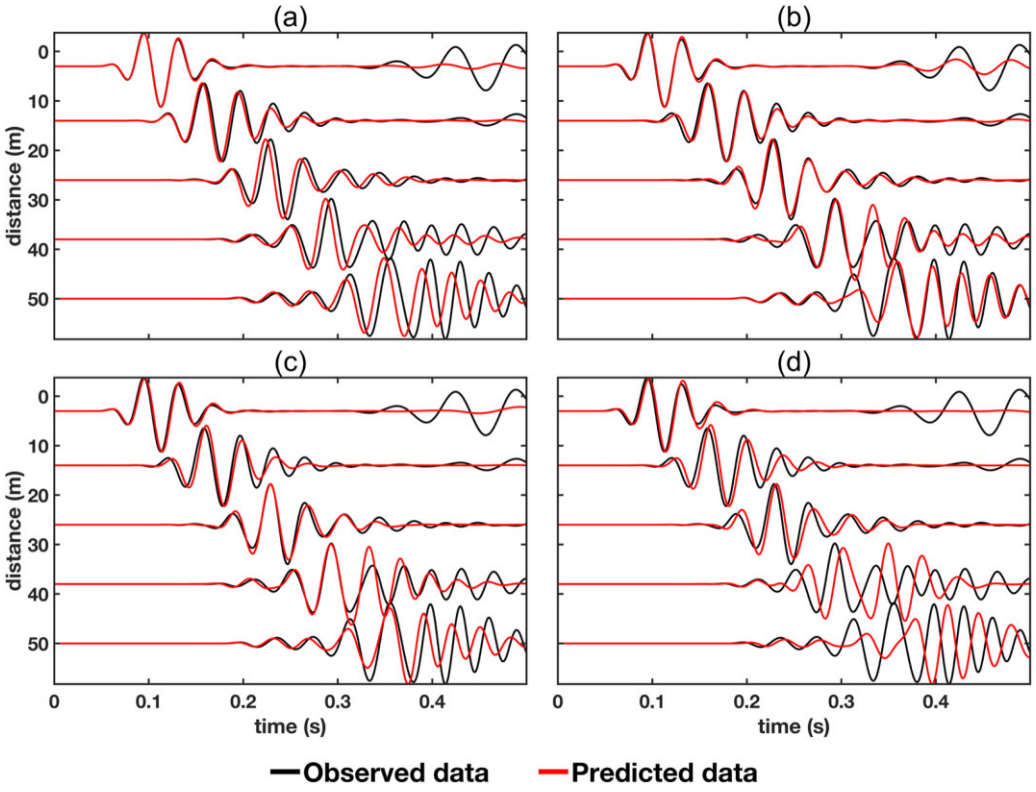

Figure 14. Comparison of observed and predicted data across different levels of explained variability used for training the network. The comparison is shown for (a) Training in full-domain, (b) EV = 98 per cent, (c) EV = 90 per cent and (d) EV = 80 per cent.

Table 3. Summary of the predicted model and data errors when varying the EVs and for the full-domain training.

	Model (p,q)	Data (p,q)	Model percentage error	Data percentage error	Training time
Full-domain	-	-	6.7	10.6	45.8 hr
EV = 98 per cent	(18, 42)	(45, 42)	3.1	6.8	3.3 hr
EV = 90 per cent	(16, 38)	(36, 38)	5.8	11.7	2.8 hr
EV = 80 per cent	(14, 35)	(35, 35)	6.2	18.7	2.6 hr

in this case, both non-parametric and parametric probability densities for the modelling error can be properly handled (Aleardi *et al.* 2021). Note that the MC approach is extremely fast because the network almost instantaneously predicts a model from the input data.

2.5 Setting the network hyperparameters

After creating the data set and identifying the appropriate combination of DCT coefficients, we proceeded to select the optimal hyperparameters for the network. To achieve this, we conducted a series of experiments, systematically adjusting one parameter at a time. For the sake of brevity here we limit to show the effect of the size of the training data set, the number of epochs used to run the training, and the optimization algorithm used to adjust the networks internal parameters. In all cases we evaluate the network performance by computing the percentage error of the desired and actual output on a validation set of 500 V_s models. We anticipate that, although the learning phase operates in the DCT domain, we quantitatively assess our results by comparing true and predicted models and observed and predicted data in full domain. In this work, the approximated percentage error for both model and data is computed as:

$$\text{Error} = \frac{\sum_k |\text{predicted}\,(k) - \text{true}\,(k)|}{\sum_k |\text{true}\,(k)|} \times 100, \tag{6}$$

where k represents each sample value of the evaluated variables. Fig. 7(a) illustrates the network performance as the training data set size varies from 1000 to 7000 samples. We observed a notable decrease in the model percentage error when utilizing 5000 samples. Since using a higher number of samples does not significantly improve the network performances, we conclude that 5000 samples constitute the optimal size for the training set. Fig. 7(b) illustrates a systematic decrease in the model percentage error with an increase in the number of epochs. However, beyond 1500 epochs, the model percentage error exhibits only a slight decrease, thus confirming that this number of epochs constitutes the optimal compromise between computational cost and generalization capability of the network. Finally, Fig. 7(c) illustrates that the Adagrad optimizer displays poor generalization capabilities, with a mean model percentage error of 80.3 per cent. In contrast, RMSprop, AdamW and Adam showed excellent performance metrics in generalizing model predictions. The Adam optimizer is the one we finally adopted.

The selected network hyperparameters are summarized in Table 2. The training phase took 3.3 hr on a server with an Debian

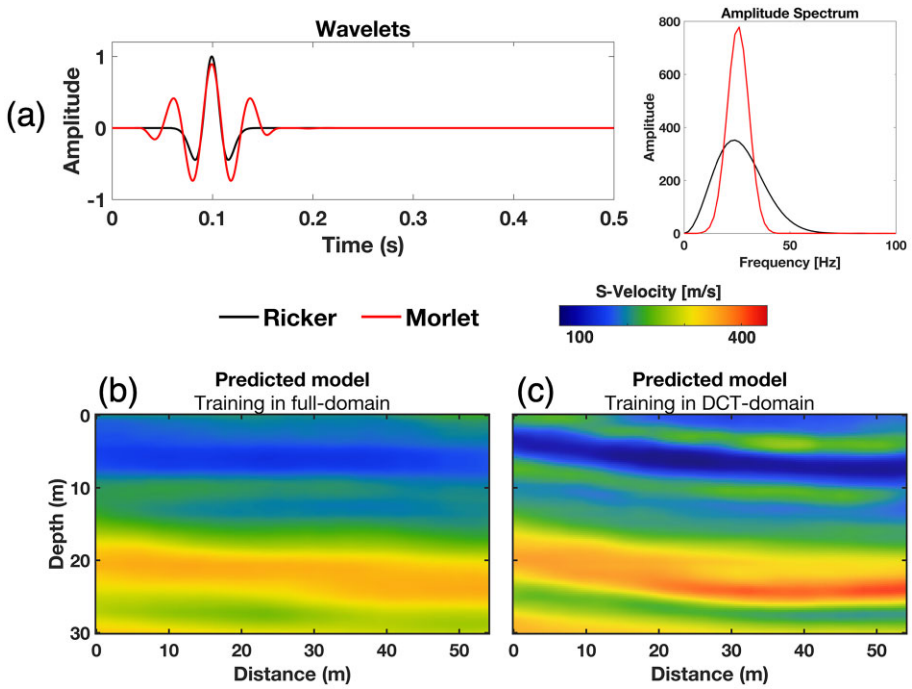

Figure 15. Network prediction with incorrect source wavelet and noise estimation: (a) comparison of Ricker and Morlet wavelets in the time and frequency domains; In this test the observed seismograms were generated with the Ricker wavelet, but the network was trained assuming the Morlet as the source wavelet. Predicted models using full-domain (b) and DCT domain (c) training.

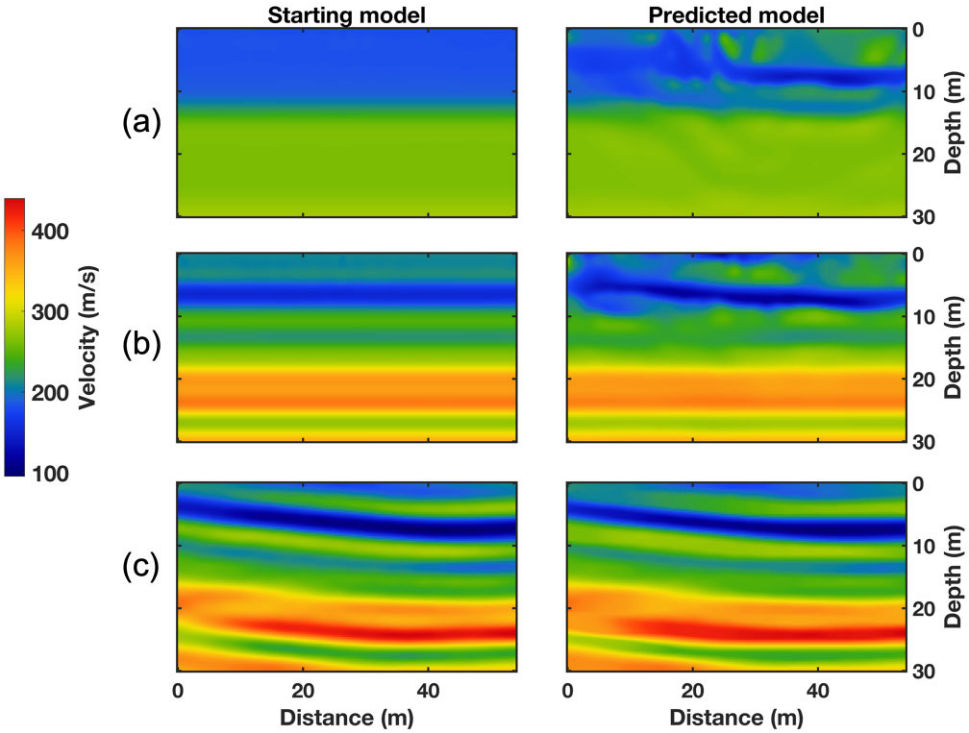

Figure 16. Comparison of the starting and predicted models for the different tests. Initial models correspond to: (a) a lateral extension of the MASW predictions; (b) a lateral extension of a column extracted from the true model and (c) the neural network solution.

operating system, powered by a 12th Gen Intel(R) Core(TM) i9-12900KF processor, and equipped with an NVIDIA GeForce RTX 3080 Ti graphics card. The algorithms were implemented using PyTorch (http://pytorch.org). Fig. 8 illustrates the evolution of the loss function during the training phase and the model percentage error as the validation metric when training the network in DCT and full-domain. The curves indicate that the learning process successfully converges in 1500 epochs using the hyperparameters previously discussed. The total time required to generate the data set, including

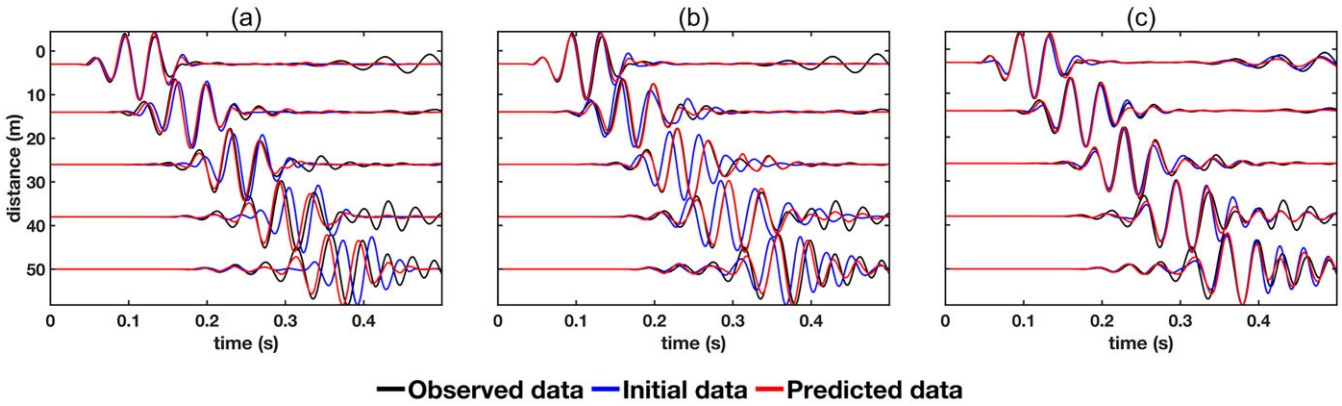

Figure 17. Comparison of the observed, starting, and predicted traces for a single shot gather for the different tests. The initial models correspond to (a) a model from MASW inversion; (b) a well-log assumption and (c) the neural network prediction.

Table 4. Percentage error for both data and models resulting from the different deterministic FWI experiments.

	Percentage of error with respect to the true model and observed data			
Test	Initial model	Predicted model	Initial data	Predicted data
(a) MASW	21.8	20.7	13.0	11.1
(b) Well-log	9.8	9.2	10.8	8.9
(c) Neural network	3.1	3.1	6.8	6.8

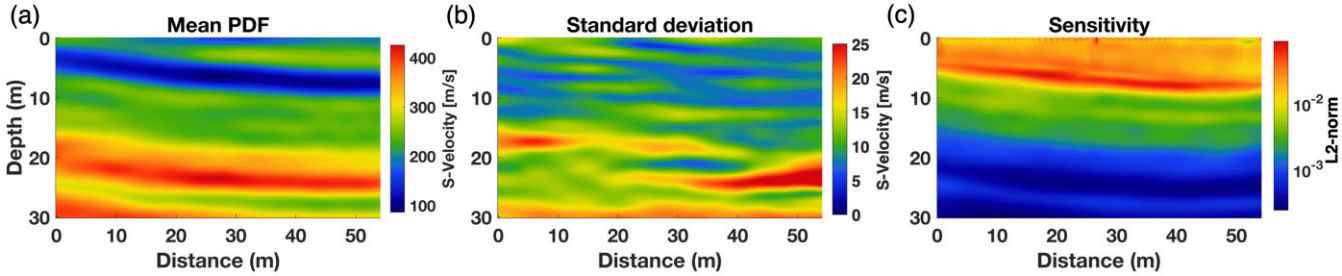

Figure 18. Uncertainty analysis using synthetic data. (a) Mean and (b) standard deviation estimated with the MC approach. (c) Sensitivity kernel computed around the network prediction.

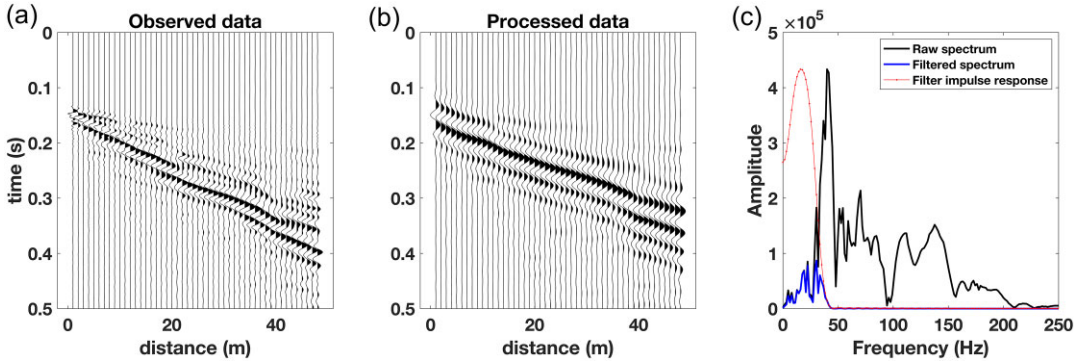

Figure 19. In (a) left shot of the raw seismic data, in (b) same shot of the processed data, whereas (c) displays the mean amplitude spectra of the raw and filtered data along with the spectrum of the applied bandpass filter.

5000 samples for training and 500 samples for validation, and to perform the training with the selected hyperparameters, amounts to 4.6 hr for the DCT domain, compared to 45.8 hr for the full-domain. This means a 92 per cent reduction of the training computational cost. In addition, DCT compression guarantees a reduction of 98 per cent of memory requirements if compared with the training in the full data and model domains. In more detail, the storage requirement decreases from 13.5 to 0.26 GB. Finally, we highlight that, although the Mean Squared Error (MSE) losses in the DCT and full-domain trainings are not directly comparable, both decrease by two orders of magnitude when increasing the number of epochs. More interestingly, the accuracy computed on the validation set is higher in the DCT domain, thus indicating better generalization capabilities compared with the training in full-domain.

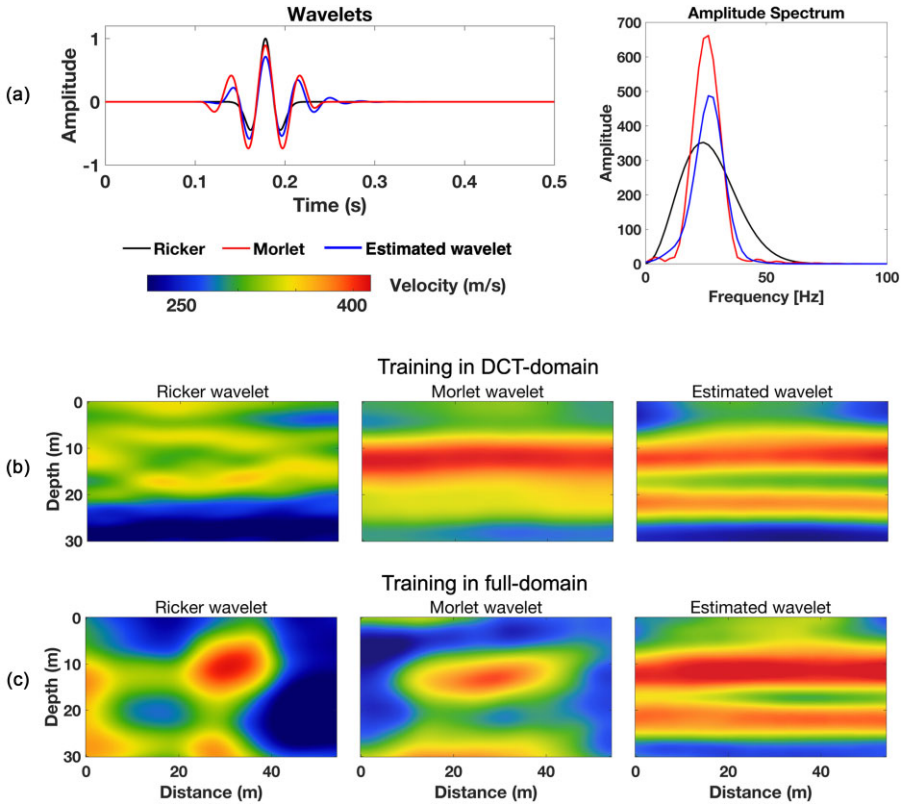

Figure 20. (a) The three different wavelets in the time and frequency domains. Predicted models are shown in (b) for DCT domain training and in (c) for full-domain training, using the different wavelets.

Table 5. Data percentage errors for the DCT and full domain trainings when using the different wavelets.

	Data percentage error	
	DCT domain	Full-domain
Ricker	22.3	30.1
Morlet	9.2	27.2
Estimated	8.3	8.7

3 RESULTS

3.1 Application to synthetic data

In this section, we test the network performance on some models extracted from the test set. We selected these models to illustrate the generalization capability of the trained network, as well as cases in which the results are less accurate. Fig. 9 shows true and predicted models, and the associated percentage error. These tests evaluate the network's performance when the wavelet is perfectly known (i.e. the same wavelet is used to create both the training and testing seismic data). In general, the results show accurate predictions for both simple subsurface geometries and more complex stratigraphic sequences. Notably, the network is also able to predict a homogeneous subsurface model. All predictions resulted in a mean percentage error lower than 5 per cent, except for tests (d) and (e), where the network is not capable to accurately predict the V_s in correspondence of very fine layers and in the presence of sharp lateral and vertical velocity contrasts. This poorer result is also probably related to the smoothing effect introduced by the DCT compression. However, even in these two more challenging tests, the main characteristics of the models are preserved, and the predictions do not exceed a 9 per cent mean model error.

Fig. 10 presents a comparison of the observed and predicted data related to the model predictions from Fig. 9, along with the associated residuals. For brevity, only one of the three shots is presented here for comparison. Apart from tests (d) and (e) where some cycle-skipping and phase mismatch are evident particularly at mid and far offsets, for the other cases a good match between observed and predicted data can be noted, which could represent a good starting point for a gradient-based FWI inversion.

The percentage errors for both the model and data are summarized in Fig. 11. As anticipated, the mean percentage error for the predicted data remains below 6 per cent, except for Tests (d) and (e), due to discrepancies in the velocity predictions. Overall, these results are promising and indicate that the network outcomes could also be used to trigger a local FWI for a further refinement of the V_s predictions.

3.1.1 Impact of the DCT compression on the deep-learning inversion

In this section, we evaluate the impact of the retained number of DCT coefficients in both model and data spaces, on the network performance. A comparison with the predictions offered when the training runs in full-domain (without the DCT compression) is also illustrated. For what concerns the DCT, we analyse the results for fixed explained variabilities of 98, 90 and 80 per cent. In these cases, the minimum number of retained coefficients for the model and data is estimated following the procedure described in Section 2.2. The retained (p, q) coefficients are as follows: for 98 per cent

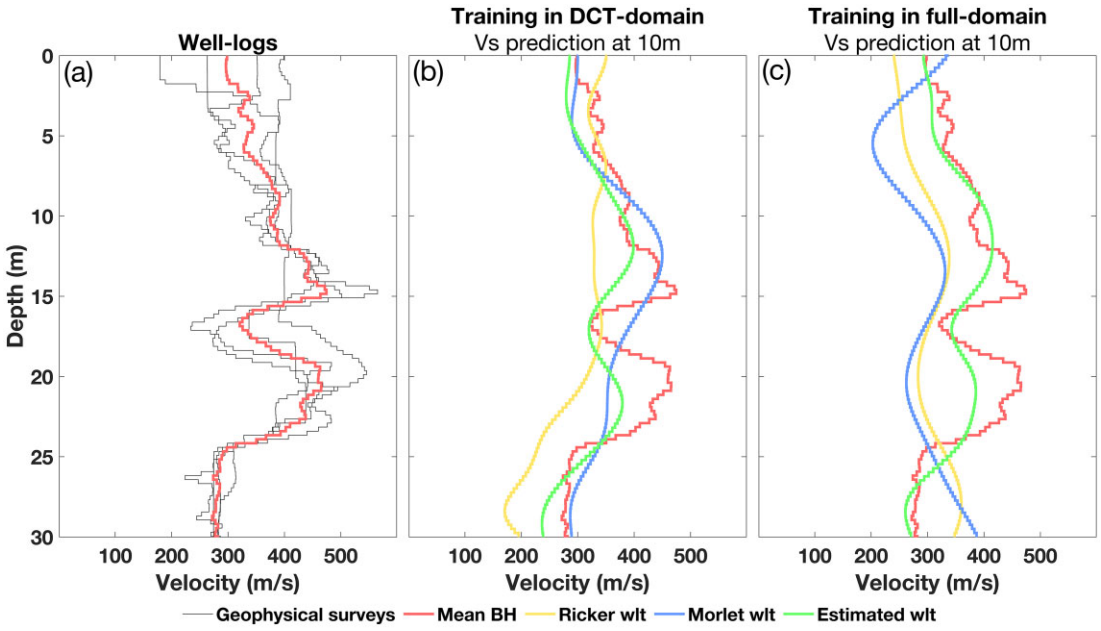

Figure 21. (a) Different V_s models obtained from several borehole surveys (black curves), with the computed average velocity shown in red. Comparison between the average borehole velocity and the network predictions when different wavelets are used to train the network in (b) DCT domain and (c) full domain.

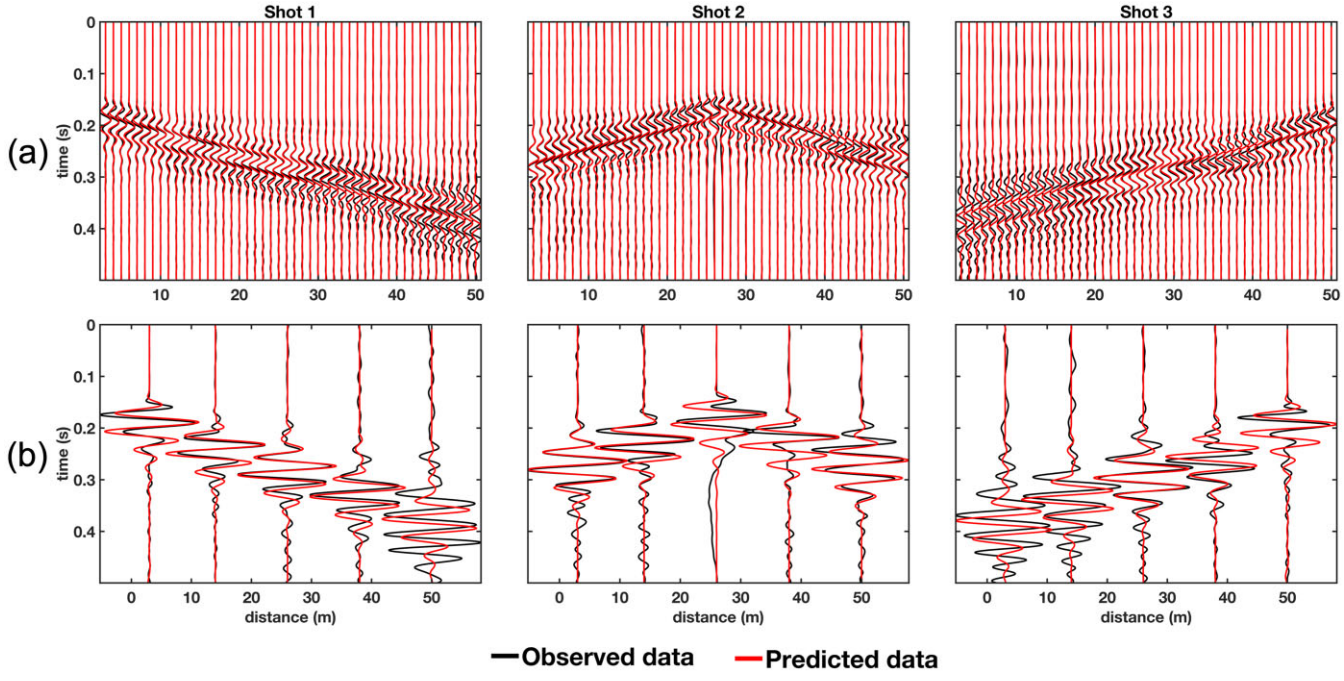

Figure 22. (a) Comparison between the observed and predicted data when using the neural network trained with the estimated wavelet in DCT domain. (b) Close-ups of five equally spaced traces extracted from each shot.

EV, (18, 42) and (45, 42); for 90 per cent EV, (16, 38) and (36, 38) and for 80 per cent EV, (14, 35) and (35, 35), for model and data, respectively.

We first analyse the impact of the compression on the input seismograms. Fig. 12(a) shows the raw seismogram computed from the benchmark test model presented in Fig. 13(a). As expected, the DCT acts as a low-pass filter, attenuating frequencies higher than 36 Hz. The approximation errors in time domain after compression are equal to 4.7, 6.8 and 11 per cent for EV = 98 per cent, EV = 90 per cent and EV = 80 per cent, respectively. However, in all cases, the selected EV levels reasonably preserve the kinematics and amplitude information of the original seismogram.

We now quantify the impact of the selected number of coefficients on the model predictions. Fig. 13(b) shows the results when training in full-domain and we can observe that the network accurately predicts the velocity magnitudes and most of the layers present in the true model, with a mean model error of 6.7 per cent. When training in the DCT domain, as expected, the error increases as the EV decreases (i.e. when fewer coefficients are retained). In the best scenario, using EV = 98 per cent, the mean predicted model error

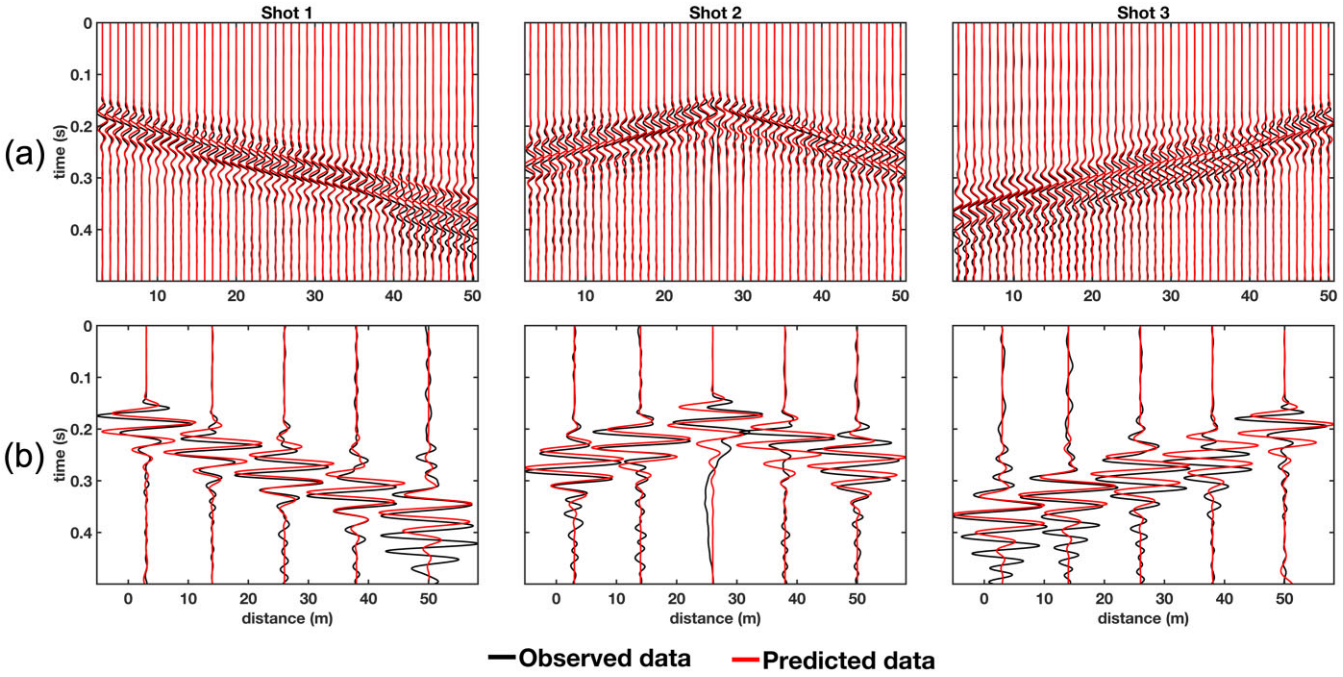

Figure 23. (a) Comparison between the observed and predicted data when using the neural network trained with the estimated wavelet in full-domain. (b) Close-ups of five equally spaced traces extracted from each shot.

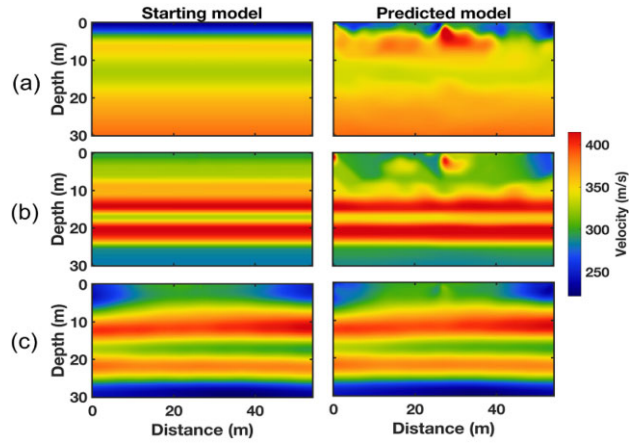

Figure 24. Comparison of the starting and predicted models for the different tests. The initial models correspond to: (a) a lateral extension of the MASW prediction; (b) a V_s model obtained from the average borehole information and (c) the neural network prediction when the estimated wavelet is used.

is 3.1 per cent, and all layers are accurately predicted (see Fig. 13c). With EV = 90 per cent the mean error rises from 3.1 per cent to 5.8 per cent (see Fig. 13d), while for an EV of 80 per cent, the final mean model error is 6.2 per cent (see Fig. 13e). Although some high-frequency features are missing when the EV is decreased, all tests produce reasonable V_s predictions, demonstrating the stability of the proposed inversion procedure with respect to the number of selected DCT basis functions. For what concerns the mean model errors, all DCT experiments outperform the results achieved with the full-domain training.

We now analyse the results in terms of predicted data. Fig. 14(a) shows the observed and predicted seismograms when training the network in full-domain. The predictions match the amplitudes and kinematics of the observed data, with some differences at far offsets. This results in an average error of 10.6 per cent. In contrast, Fig. 14(b) shows the results when training the network with EV = 98 per cent. Notably, the predicted kinematics and amplitudes closely align with the observations, with final errors of 6.8 per cent. When the number of retained coefficients is reduced to EV = 90 per cent, the data error increases from 6.8 to 11.7 per cent (see Fig. 14c). As expected, a further reduction of the retained coefficients (EV = 80 per cent) leads to a poorer alignment between predicted and observed data, particularly at far offsets. In this case, the error increases reaching up to 18.7 per cent (see Fig. 14d).

Table 3 summarizes the errors for the different tests along with the computational cost of the training. Notably, reducing the number of retained coefficients partially reduces the accuracy of the predictions without substantially lowering the computational cost. The errors obtained with the full-domain training are comparable to those achieved by the training with EV = 90 per cent. It is also clear that the best results are obtained with EV = 98 per cent. Therefore, these tests demonstrate that the selected EV value constitutes the best compromise between the accuracy of the predictions and the computational cost.

3.1.2 Impact of wrong assumptions about the source wavelet and data noise

Finally, we aim to assess the stability of the network predictions in a more realistic scenario where the source wavelet and the noise contaminating the data are not perfectly estimated. Specifically, we compute the observed data (the input for the network) using a Ricker wavelet with a peak frequency of 25 Hz. Additionally, we contaminate the noise-free observations with 10 per cent of Gaussian and uncorrelated noise (remind that the training data set was generated with a 25 Hz Morlet wavelet and assuming the 5 per cent of uncorrelated Gaussian noise in the seismograms; see Fig. 15a). As in the

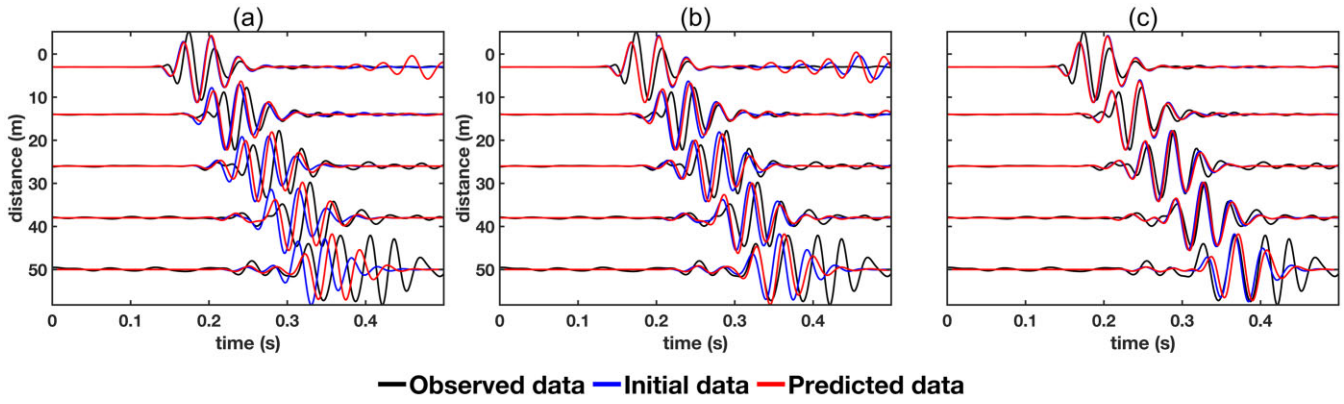

Figure 25. Comparison of the observed, starting, and predicted traces for the different tests. The initial models correspond to (a) a lateral extension of the MASW prediction; (b) a V_s model derived from the borehole information and (c) the neural network prediction when the estimated wavelet is used.

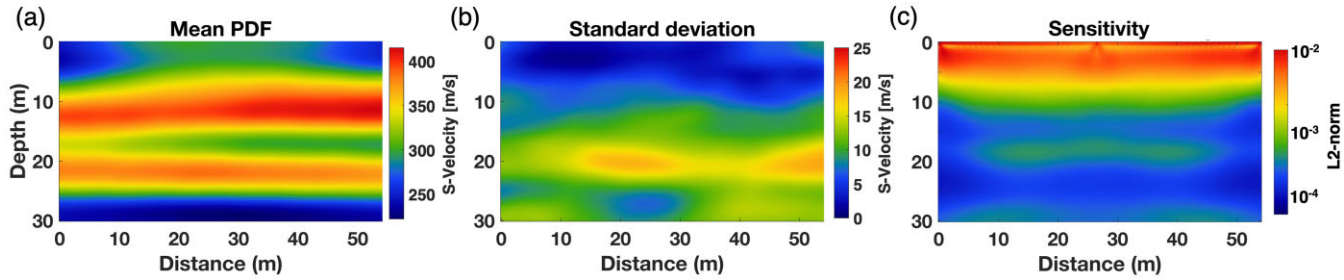

Figure 26. Uncertainty analysis using the field data. Estimated (a) Mean and (b) standard deviation values. (c) Sensitivity kernel computed around the network predictions of Fig. 20(b).

Table 6. Data errors for the deterministic FWI tests.

Test	Percentage of data error with respect to the observed data: Initial data	Predicted data
(a) MASW	13.8	10.2
(b) Well-log	12.1	9.4
(c) NN estimated wavelet	8.3	7.9

previous test, we utilize the V_s model from Fig. 13(a) as the benchmark. We again compare the results obtained from full-domain training with those from DCT domain training with EV = 98 per cent. As expected, the mean percentage model error increases when compared to the previous tests. For the full-domain training, the error rises from 6.7 to 12.2 per cent (see Fig. 15b). In contrast, the DCT domain training yields a lower prediction error equal to 5.8 per cent (see Fig. 15c). Notwithstanding the erroneous assumptions, the proposed approach still effectively predicts the subsurface velocity model and outperforms the network trained in full-domain. This suggests that working in the compressed space ensures more stable predictions in case of erroneous assumptions about the noise level in the data and reasonable errors in the estimated source wavelet.

3.1.3 Deterministic FWI examples

In this section, we show the performance of a traditional deterministic FWI starting from different initial models. We performed the FWI utilizing IFOS2D, an open-source 2-D elastic FWI code (Bohlen 2002). This algorithm uses the pre-conditioned conjugate gradient as the optimization strategy and the adjoint method to compute the gradient (Tarantola 1984). We implemented a multiscale FWI using the frequency marching strategy. The frequency range was from 3 to 40 Hz with frequency steps of 3 Hz. The error function to be minimized is the L2 norm between modelled and observed data, both conveniently scaled with respect to the energy of the observed seismograms (as suggested by Choi & Alkhalifah 2012).

We present three examples in which the true model corresponds to the one presented in Fig. 13(a). For each example, we start the inversion from a different initial model: Test (a) triggers the FWI from a velocity model derived from a lateral extension and a slight vertical smoothing of the V_s predicted by an MASW inversion; Test (b) simulates the availability of borehole information and uses a V_s starting model derived by a lateral extension and a vertical smoothing of a single column extracted from the true V_s. Finally, Test (c) utilizes the neural network prediction as the starting point for the FWI. We performed the MASW inversion utilizing the open-source software Geopsy (Wathelet *et al.* 2004), which uses the neighborhood algorithm to minimize the L2 norm difference between the picked and the modelled dispersion curves. The results of the MASW inversion are briefly illustrated in the Appendix.

Fig. 16 shows the final predictions. Test (a) demonstrates that the MASW method adequately approximates the velocity in the first 10 m. It is important to note that the predicted model from the FWI provides some insights into the location and dipping of the layers, and effectively updates the velocity magnitudes in the shallower part of the subsurface (see Fig. 16a). However, the inversion fails to accurately depict the continuity of the layers, and below 12 m, the final model barely updates from the starting model. In this test, the model percentage error decreases from 21.8 to 20.7 per cent, when considering the starting model and the final FWI predictions, respectively.

For Test (b), the FWI solution still successfully reproduces the velocities and the dip of the layers in the shallow part of the model (i.e. above 12 m depth) but fails to update the velocities in the deeper part of the model. In this case, the mean model error improves from 9.8 to 9.2 per cent (see Fig. 16b). Finally, the results of Test (c) show that the starting model already provides a reasonable approximation of the true one, as evidenced by the lowest percentage error associated with the initial model, equal to 3.1 per cent. In this case, the prediction after the deterministic FWI maintains the same percentage of error with no significant changes, thereby indicating that the neural network already offered a satisfactory approximation of the true velocity (see Fig. 16c).

We now analyse the results concerning the observed, initial, and predicted data (Fig. 17), whereas Table 4 summarizes the model and data percentage errors. In Test (a), the data error improves from 13 to 11 per cent. Although the observed data at near offsets closely aligns with the predicted data, the predicted data does not accurately reproduce the kinematics beyond 0.3 s from middle to far offsets (see Fig. 17a). Test (b) shows improved performance for middle offsets, but with still some misalignments at far offsets. For this test, the data error is reduced from 10.8 to 8.9 per cent (see Fig. 17b). Finally, in Test (c), the initial data already aligns well with most of the kinematics and amplitude of the observed data; the predicted data now closely resembles the observed seismograms even at far offsets. In this third test, no significant changes were observed in both data and model predictions.

3.1.4 Uncertainty estimation

We now present the MC uncertainty estimation obtained for the subsurface model shown in Fig. 13(c) when 10 000 simulations are drawn. We assume 5 per cent of Gaussian uncorrelated noise contaminating the data (as the one assumed to generate the training set). The MC estimation using the network is computationally efficient, indeed less than 5 min is enough to get a numerical approximation of the posterior distribution. Fig. 18 presents the estimated uncertainty, in the form of the posterior mean and posterior standard deviation. The posterior mean retrieves a final solution that closely approximates the true model and generates seismic data very similar to the observed seismograms, with errors of 6.7 and 9.1 per cent for model and data, respectively. Note that the mean of the posterior does not present a low-velocity layer around 10 m depth, which is consistent with a high standard deviation in correspondence of the same depth interval (Fig. 18b). As expected, the uncertainty increases towards the bottom of the model. Additionally, some of the large uncertainties present in the top portion of the model are consistent with slightly misleading predictions of the velocity magnitudes in those portions with respect to the true model. The reliability of the predicted uncertainty is assessed through a comparison with the sensitivity kernel computed around the network solution (see Fig. 18c). For a non-linear inversion, it is indeed known that such a sensitivity kernel gives an approximate local estimation of the uncertainty valid around a specific model.

According to the sensitivity, and as highly expected, the most illuminated parameters are those located in the shallowest part of the model (down to 8 m). Then we have a significant decrease in the parameter's illumination between 8 and 16 m depth, and a very limited illumination below 16 m. The coherence between the MC and the sensitivity outputs is a further confirmation of the reliability of the implemented uncertainty estimation method.

3.2 Application to field data

In this section, we apply the deep learning inversion framework to the field data acquired within the InterPACIFIC project in Grenoble, France (Garofalo *et al.* 2016). We chose this data set due to the availability of borehole information located 10 m along the acquisition profile, which allows us to validate the V_s predictions at that location. The receivers have a natural frequency of 4.5 Hz, and the seismic source consists of an 8 kg sledgehammer. To align the field data with the synthetic data used to train the network, we conducted a preprocessing sequence involving trace-by-trace amplitude normalization, a zero-phase bandpass filter (3–30 Hz), and a 3-D to 2-D correction. The 3-D to 2-D correction accounts for the different geometrical spreading between the real-case point source and the line sources used in the 2-D forward modelling (Forbriger *et al.* 2014). Fig. 19 shows the raw and processed field data with the corresponding amplitude spectra.

We estimated the source wavelet from the near-offset traces of the observed data using the singular value decomposition (SVD) approach (Biondi *et al.* 2014). As discussed earlier, the acquisition geometry matches that described in the synthetic case, and the network architecture, as well as the models forming the training set, are the same used in the synthetic tests. The associated data have initially been generated with the same forward algorithm but considering the wavelet extracted from the recorded seismograms. In addition, to assess the impact of varying wavelets on the inversion results, we also estimated the V_s from networks trained with two different source wavelets: a 25 Hz Ricker wavelet and a 25 Hz Morlet wavelet.

Fig. 20(a) compares the three wavelets in both time and frequency domains. We observe that the estimated and the Morlet wavelets are similar in both time and frequency domains, while the Ricker shows a wider frequency content resulting in a more defined peak in time domain. Figs 20(b) and (c) show the model predictions when training the network with the different wavelets in DCT and full domain, respectively. Both networks give poor predictions when the Ricker wavelet is considered. However, note that the DCT domain training produces more stable results compared with the full-domain training. Indeed, for the DCT training, we get similar velocity predictions for the first 18 m depth when both the Morlet and estimated wavelet are used for training, while the network working in full-domain seems to achieve physically reasonable predictions only when the estimated wavelet is considered. Table 5 summarizes the data errors for all the predicted models in both DCT and full domain using the different wavelets. Again, the DCT training yields similar data predictions when both the Morlet and the estimated wavelets are used, instead the full-domain training produces a good fitting only with the estimated wavelet.

Fig. 21(a) shows the available borehole information from the downhole, cross-hole and *P*-*S* suspension logging measurements (visible as black curves in the background), whereas the red curve depicts the average velocity computed from the different borehole data. The results for the network trained in the compressed domain are shown in Fig. 21(b). The network trained with the Ricker wavelet provides a poor approximation of the velocity magnitude when compared with the borehole information. The network trained using the Morlet wavelet correctly predicts a layered model with a

high-velocity layer located between 10 and 15 m depth. This prediction aligns with the borehole information from 0 to 15 m depth. However, the networks struggle to accurately recover the actual velocity in the deeper parts of the model (below 15 m), where parameter illumination is poor. When the estimated wavelet is used, we observe a good agreement between the velocity from the borehole and the predicted V_s. In particular, the network is capable of predicting the velocity reversal occurring at 15 m depth. Fig. 21(c) shows the predicted velocity when training is performed in the full domain. In this case, reasonable predictions are achieved only when the estimated wavelet is used to train the network. This experiment demonstrates that the use of the DCT guarantees more accurate and stable predictions than those obtained when working in the full domain, especially when the source wavelet is not accurately estimated.

Fig. 22 illustrates a comparison of the observed and predicted data when using the network trained with the estimated wavelet in the DCT domain. Notably, the predicted seismograms show no evidence of cycle skipping, suggesting that, if needed, the network predictions constitute a promising initial model for a deterministic FWI. Finally, Fig. 23 shows the data comparison for the full-domain training when considering the estimated wavelet. Even in this case, we achieve good data matching with no clear evidence of cycle skipping. In both experiments we achieved similar data matching equal to 8.3 per cent for the DCT-domain and 8.7 per cent for the full-domain.

3.2.1 *Deterministic FWI examples*

Similar to the synthetic application, we now perform three different tests to assess the performance of the deterministic FWI when the starting models are:

(i) Test (a): a lateral extension of the MASW prediction (see the Appendix).
(ii) Test (b): a V_s model obtained from a lateral extension of the average borehole information.
(iii) Test (c): the neural network prediction (trained in DCT domain) when the estimated wavelet is considered.

In all the following tests we adopt the same FWI hyperparameters (i.e. number of frequency marching steps, error function, etc.) previously used in the synthetic experiments. Fig. 24 shows the results of the FWI tests. In all cases, the predicted model shows negligible updates below 15–18 m depth, with some artifacts appearing at shallow depth, probably introduced by the gradient operator [especially in tests (a) and (b)]. All these experiments further confirm the importance of the initial model in standard FWI; even when using a multiscale approach, the severe non-linearity of the forward operator causes the inversion to be prone to getting stuck in local minima of the error function. Specifically, the velocity inversion occurring around 15 m depth can be correctly recovered only if already present in the starting model.

Fig. 25 presents a comparison of traces extracted from the observed, initial, and final predicted data for each test. To better appreciate the amplitude and kinematics, we show only five equally spaced traces, extracted from the first shot. In Fig. 25(a), both the initial and predicted data do not align with the observed data, although the percentage error decreases from 12.1 to 9.4 per cent when moving from the starting model to the final FWI predictions. Figs 25(b) and (c) show better data fitting, with test (c) achieving the lowest data misfit values for both the starting model and the final FWI prediction (8.3 and 7.9 per cent, respectively). Table 6 summarizes the data errors for all these tests.

3.2.2 *Uncertainty estimation*

We now use the MC method to evaluate the uncertainty affecting the neural network solution. Similar to the previous synthetic test, we conducted 10 000 simulations, under the assumption of 5 per cent of Gaussian and uncorrelated noise affecting the data. As previously mentioned, this quantification is obtained from the standard deviation computed on repeated shots acquired in the field. Fig. 26 illustrates the results of the estimated uncertainty, summarized by the posterior mean and the posterior standard deviation. It's noteworthy that the posterior mean closely aligns with the model predicted by the network (Fig. 20b). Fig. 26(b) shows an increase in uncertainties below 12 m depth, with the largest values observed in the interbedded strata between 14 and 18 m depth. This suggests that the velocity inversion that occurs at that depth has a limited influence on the seismic records. The MC outcomes are consistent with the sensitivity kernel computed around the V_s estimated by the network trained with the estimated wavelet (Fig. 26c), thereby indicating that the model parameters located in the first 12 m depth are the most informed by the data.

4 DISCUSSION

Classical approaches to surface wave inversion do not account for lateral variations. Both deterministic and probabilistic 2-D FWI methods overcome this limitation. The deterministic approach heavily relies on the starting model and may encounter cycle skipping issues, while also lacking in estimating model uncertainties. On the other hand, probabilistic approaches are not reliant on the starting model and include uncertainty estimations, albeit they come with high computational costs. In this context, we were motivated to integrate deep learning techniques to devise an alternative strategy that mitigates initial model dependency and uncertainty estimation challenges while maintaining modest computational demands. To overcome these problems, we proposed a data-driven *S*-velocity estimation method by combining neural networks with the DCT compressional technique and incorporating a network MC sampling strategy to estimate the uncertainty affecting the solution.

The implemented inversion using deep learning is characterized by an affordable computational cost, good generalization capabilities, and provides accurate S-velocity predictions with low data misfit. However, the velocity models forming the training data set are inherently limited and cannot encompass all potential geological conditions and velocity ranges. This limitation highlights the importance of a training data set that accurately represents typical near-surface geological scenarios, along with associated data that reflects realistic kinematics typically found in field data acquisitions. For example, some preliminary tests with velocity models generated using tractable analytical distributions (i.e. Gaussian) have led to non-realistic subsurface geometries in the training set, thereby compromising the results (Rincón 2023).

Our results demonstrated the robustness of the trained network against erroneous noise assumptions and estimated wavelets. However, any changes in acquisition settings, topography, or significant variations in the wavelets would necessitate retraining the network. In our case, this process required a total effective time of 4.6 hr using the previously mentioned servers. One potential avenue for future research to avoid the necessity for retraining involves the

exploration of transfer learning techniques (TL). TL is a machine-learning technique used to apply the knowledge gained from a previously trained machine-learning model to a new problem, thereby avoiding the need for complete retraining and making the process more efficient (Weiss *et al.* 2016).

By integrating the DCT-compression technique into our deep-learning framework, we effectively reduced the dimensionality of the parameter space, resulting in a 74 per cent decrease in computational cost compared to training in the full domain. Additionally, the memory requirement was significantly reduced by 98 per cent. However, the trade-off of using this compression technique lies in the proper selection of the number of DCT coefficients. Although we meticulously selected the pair of retained coefficients to account for the specified threshold of 98 per cent of the EV for the entire data set, it is possible that certain geometrical characteristics, such as thin layers and sharp contacts, may be lost due to DCT compression. In some cases, it may be necessary to retain a higher number of DCT coefficients to account for these velocity variations, resulting in an increase in the computational cost during the training phase. In this context, it could be argued that compressions performed using non-linear methods (e.g. convolutional autoencoders) might be more efficient than DCT in preserving these characteristics (Aleardi *et al.* 2022). Furthermore, since DCT can be either mono- or multi-dimensional, the implemented method can also be adapted for 3-D surface wave FWI. This adaptation would require the generation of plausible 3-D geological models for the data set and the integration of a more computationally efficient elastic forward operator. While there would be a notable increase in computational costs for both data set generation and the training phase, addressing cycle skipping in 3-D FWI poses greater challenges, thereby emphasizing the importance of a well-defined starting model.

Using the MC method, we estimated the uncertainty affecting our solution through 10 000 simulations within a negligible timeframe. The posterior mean and standard deviation can be utilized to interpret our findings. We observed good agreement between the uncertainty and the local approximation of the sensitivity. This uncertainty analysis demonstrates that the V_s below 12 m depth is poorly constrained by the observation and for this reason, we expect lower precision and accuracy below that depth. However, the high-quality predictions offered by the proposed approach below 12 m (i.e. for those parameters poorly informed by the data), seem to demonstrate that the network successfully learned the inverse operator from the training set. Future work could explore other deep learning-derived methodologies for estimating solution uncertainty, such as using MC Dropout methods.

We finally point out some additional potential applications of our approach. One possible avenue is to use the deep learning proposed model as the initial model in a probabilistic FWI framework. We anticipate that this could accelerate the sampling of the posterior distribution by reducing the number of iterations required to complete the *burn-in* period, which consists of the initial iterations of a MCMC inversion, during which the corresponding velocity models are excluded from the numerical estimation of the statistical properties of the PPD. Additionally, in extensive geophysical campaigns where the wavelet is estimated and array settings remain constant, this method has the potential to offer immediate geologically plausible velocity model predictions. This capability can aid in optimizing both acquisition strategies through informed decision-making based on the results.

The implemented method could potentially be enhanced through an unsupervised formulation, wherein we establish a loop connecting the forward modelling and CNN to update the weights and biases based on the data misfit (Jin *et al.* 2021) instead of the model misfit. Such optimization necessitates a transfer learning approach, as the proposed models must satisfy the Courant–Friedrichs–Lewy conditions to simulate the predicted data (Courant *et al.* 1967). This proposition leads to a semi-supervised framework, requiring one forward computation for each sample data set at each epoch. Furthermore, different network architectures, such as DeepONets, could be explored, as this architecture has demonstrated high performance in effectively approximating complex, nonlinear operators, thereby enabling better generalizations (Lu *et al.* 2021; Zhu *et al.* 2023).

5 CONCLUSIONS

We have introduced a data-driven approach for V_s-model prediction that integrates deep learning with DCT. The implemented deep learning inversion method effectively addressed surface wave FWI, showing high generalization capabilities and accurately proposing velocity models with low data misfits and almost no evidence of cycle skipping. The inclusion of DCT-compression accelerates the training phase and reduces the dimensionality of the input and outputs of the network. Our approach also provides a comprehensive estimation of uncertainties affecting the proposed models in a computationally efficient manner. Furthermore, the velocity model predictions in real data applications aligned accurately with borehole information and guaranteed good data matching and absence of cycle skipping. We also demonstrated that the inclusion of the DCT compression within the deep-learning framework not only reduces the computational workload for the training phase but, more importantly, guarantees quite stable predictions when the estimated wavelet is close to the propagating wavelet.

ACKNOWLEDGMENTS

We express our gratitude to Nicola Bienati from ENI for his valuable assistance and insightful comments, which greatly enhanced the quality of this work. We also extend our thanks to the Associate Editors and the two anonymous reviewers for their constructive feedback, which significantly contributed to improving the original manuscript. The seismic data processing was carried out using the ProMax software by Landmark Graphics Corporation, which we gratefully acknowledge. This research was supported by EU funding within the MUR NRRP "National Recovery and Resilience Plan" (Project CUP I51J23000450007, No. 226).

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests of personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The data presented in this study are available on request from the corresponding authors.

REFERENCES

Abbas, A., Vantassel, J.P., Cox, B.R., Kumar, K. & Crocker, J., 2023. A frequency-velocity CNN for developing near-surface 2D vs images from

linear-array, active-source wavefield measurements, *Comput. Geotech.,* **156,** 105305

Aleardi, M., 2020. Combining discrete cosine transform and convolutional neural networks to speed up the Hamiltonian Monte Carlo inversion of pre-stack seismic data, *Geophys. Prospect.,* **68**(9), 2738–2761.

Aleardi, M., Pierini, S. & Sajeva, A., 2019. Assessing the performances of recent global search algorithms using analytic objective functions and seismic optimization problems, *Geophysics,* **84**(5), R767–R781.

Aleardi, M., Salusti, A. & Pierini, S., 2020. Transdimensional and Hamiltonian Monte Carlo inversions of Rayleigh-wave dispersion curves: a comparison on synthetic datasets, *Near Surf. Geophys.,* **18**(5), 515–543.

Aleardi, M., Vinciguerra, A. & Hojat, A., 2021. A convolutional neural network approach to electrical resistivity tomography, *J. appl. Geophys.,* **193,** 104434

Aleardi, M., Vinciguerra, A., Stucchi, E. & Hojat, A., 2022. Stochastic electrical resistivity tomography with ensemble smoother and deep convolutional autoencoders, *Near Surf. Geophys.,* **20**(2), 160–177.

Alkhalifah, T., 2016. Full-model wavenumber inversion: an emphasis on the appropriate wavenumber continuation, *geophysics,* **81**(3), r89–r98.

Berti, S., Aleardi, M. & Stucchi, E. 2024a. A Bayesian approach to elastic full-waveform inversion: application to two synthetic near surface models, *Bull. Geophys. Oceanogr.,* **65**(2), 291–308.

Berti, S., Aleardi, M. & Stucchi, E., 2024b. A probabilistic full waveform inversion of surface waves, *Geophys. Prospect.,***72,** 3448–3473,

Berti, S., Aleardi, M. & Stucchi, E., 2024c. A computationally efficient Bayesian approach to full-waveform inversion, *Geophys. Prospect.,* **72**(2), 580–603.

Biondi, E., Stucchi, E. & Mazzotti, A., 2014. Nonstretch normal moveout through iterative partial correction and deconvolution, *Geophysics,* **79**(4), V131–V141.

Bohlen, T. & Saenger, E.H., 2006. Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves, *Geophysics,* **71**(4), T109–T115.

Bohlen, T., 2002. Parallel 3-D viscoelastic finite difference seismic modelling, *Comput. Geosci.,* **28**(8), 887–899.

Britanak, V., Yip, P.C. & Rao, K.R., 2010. *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms And Integer Approximations*. Elsevier.

Choi, Y. & Alkhalifah, T., 2012. Application of multi-source waveform inversion to marine streamer data using the global correlation norm, *Geophys. Prospect.,* **60**(4), 748–758.

Courant, R., Friedrichs, K. & Lewy, H., 1967. On the partial difference equations of mathematical physics, *IBM J. Res. Dev.,* **11**(2), 215–234.

Dal Moro, G., 2019. Surface wave analysis: improving the accuracy of the shear-wave velocity profile through the efficient joint acquisition and Full Velocity Spectrum (FVS) analysis of Rayleigh and Love waves, *Explor. Geophys.,* **50**(4), 408–419.

de la Varga, M., Schaaf, A. & Wellmann, F., 2019. GemPy 1.0: open-source stochastic geological modeling and inversion, *Geosci. Model. Dev.,* **12**(1), 1–32.

Deng, C. *et al.* 2022. OpenFWI: Large-scale multi-structural benchmark datasets for full waveform inversion, *Adv. Neural Inform. Process. Syst.,* **35,** 6007–6020.

Dhara, A. & Sen, M.K., 2022. Physics-guided deep autoencoder to overcome the need for a starting model in full-waveform inversion, *Leading Edge,* **41**(6), 375–381.

Dumoulin, V. & Visin, F., 2016. A guide to convolution arithmetic for deep learning,arXiv preprint arXiv:1603.07285

Forbriger, T., 2003. Inversion of shallow-seismic wavefields: I. Wavefield transformation, *Geophys. J. Int.,* **153**(3), 719–734.

Forbriger, T., Groos, L. & Schäfer, M., 2014. Line-source simulation for shallow-seismic data. Part 1: theoretical background, *Geophys. J. Int.,* **198**(3), 1387–1404.

Gal, Y. & Ghahramani, Z., 2016, Dropout as a bayesian approximation: representing model uncertainty in deep learning, in *International Conference on Machine Learning,* pp. 1050–1059, PMLR.

Garofalo, F. *et al.* 2016. InterPACIFIC project: Comparison of invasive and non-invasive methods for seismic site characterization. Part I: Intra-comparison of surface wave methods, *Soil Dyn. Earthq. Eng.,* **82,** 222–240.

Gebraad, L., Boehm, C. & Fichtner, A., 2020. Bayesian elastic full-waveform inversion using Hamiltonian Monte Carlo, *J. geophys. Res.,* **125**(3), e2019JB018428.

Hansen, T.M. & Cordua, K.S., 2017. Efficient Monte Carlo sampling of inverse problems using a neural network-based forward—applied to GPR crosshole traveltime inversion, *Geophys. J. Int.,* **211**(3), 1524–1533.

Hu, W., Jin, Y., Wu, X. & Chen, J., 2020. Physics-guided self-supervised learning for low frequency data prediction in FWI, in *SEG Technical Program Expanded Abstracts 2020,* pp. 875–879, Society of Exploration Geophysicists.

Ivanov, I.I., McKenzie, B.S., Zhou, L., Tadokoro, C.E., Lepelley, A., Lafaille, J.J., Cua, D.J. & Littman, D.R., 2006. The orphan nuclear receptor RORγt directs the differentiation program of proinflammatory IL-17+ T helper cells, *Cell,* **126**(6), 1121–1133.

Jin, P., Zhang, X., Chen, Y., Huang, S.X., Liu, Z. & Lin, Y., 2021. Unsupervised learning of full-waveform inversion: Connecting CNN and partial differential equation in a loop, arXiv preprint arXiv:2110.07584.

Kazei, V., Ovcharenko, O., Plotnitskii, P., Peter, D., Zhang, X. & Alkhalifah, T., 2021. Mapping full seismic waveforms to vertical velocity profiles by deep learning, *Geophysics,* **86**(5), R711–R721.

Kendall, A. & Gal, Y., 2017. What uncertainties do we need in bayesian deep learning for computer vision?, *Adv. Neural Inform. Process. Syst.,* **30**.

Kotsi, M., Malcolm, A. & Ely, G., 2020. Uncertainty quantification in time-lapse seismic imaging: a full-waveform approach, *Geophys. J. Int.,* **222**(2), 1245–1263.

Kumar, J. &, . . . , pp..

Lochbühler, T., Breen, S.J., Detwiler, R.L., Vrugt, J.A. & Linde, N., 2014. Probabilistic electrical resistivity tomography of a CO_2 sequestration analog, *J. appl. Geophys.,* **107,** 80–92.

Long, J., Shelhamer, E. & Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp.3431–3440, IEEE.

Lu, L., Jin, P., Pang, G., Zhang, Z. & Karniadakis, G.E., 2021. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nat. Mach. Intell.,* **3**(3), 218–229.

Maraschini, M. & Foti, S., 2010. A Monte Carlo multimodal inversion of surface waves, *Geophys. J. Int.,* **182**(3), 1557–1566.

Miller, G.F., Pursey, H. & Bullard, E.C., 1955. On the partition of energy between elastic waves in a semi-infinite solid, *Proc. R. Soc. Lond., A,* **233**(1192), 55–69.

Mosegaard, K. & Tarantola, A., 2002. Probabilistic approach to inverse problems, *Int. Geophys. Ser.,* **81**(A), 237–268.

Mosser, L., Purves, S. & Naeini, E.Z., 2020, Uncertainty quantification for deep learning in geoscience applications, in *EAGE/AAPG Digital Subsurface for Asia Pacific Conference,* Vol. **2020,** pp. 1–3, European Association of Geoscientists & Engineers.

Osband, I., 2016, Risk versus uncertainty in deep learning: bayes, bootstrap and the dangers of dropout, in MIT Press. *NIPS Workshop on Bayesian Deep Learning* Vol. **192**.

Ovcharenko, O., Kazei, V., Kalita, M., Peter, D. & Alkhalifah, T., 2019. Deep learning for low-frequency extrapolation from multioffset seismic data, *Geophysics,* **84**(6), R989–R1001.

Pang, Y., Nie, L., Liu, B., Liu, Z. & Wang, N., 2020. Multiscale resistivity inversion based on convolutional wavelet transform, *Geophys. J. Int.,* **223**(1), 132–143.

Park, C.B., Miller, R.D. & Xia, J., 1999. Multichannel analysis of surface waves, *Geophysics,* **64**(3), 800–808.

Richardson, A., 2018. Seismic full-waveform inversion using deep learning tools and techniques,arXiv preprint arXiv:1801.07232

Richart, F.E., Hall, J.R. & Woods, R.D., 1970. *Vibration of Soils and Foundations,* Prentice Hall.

Rincón, F., 2023, Accelerating full-waveform inversion of surface waves through a properly trained Neural Network and Discrete Cosine Transform, *MSc thesis*, Univ. Pisa.

Ronneberger, O., Fischer, P. & Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation, in *Proceedings of the MICCAI 2015: 18th International Conference on Medical Image Computing and Computer-Assisted Intervention,* Munich, Germany, 5–9 October 2015, Part III, 234–241, Springer International Publishing.

Sajeva, A., Aleardi, M. & Mazzotti, A., 2017. Genetic algorithm full-waveform inversion: uncertainty estimation and validation of the results, *Boll. Geof. Teorica Appl.,* **58**(IV), 395–414.

Schäfer, M., Groos, L., Forbriger, T. & Bohlen, T., 2014. Line-source simulation for shallow-seismic data. Part 2: full-waveform inversion—a synthetic 2-D case study, *Geophys. J. Int.,* **198**(3), 1405–1418.

Schmidt, H. & Jensen, F.B., 1985. A full wave solution for propagation in multilayered viscoelastic media with application to Gaussian beam reflection at fluid–solid interfaces, *J. acoust. Soc. Am.,* **77**(3), 813–825.

Socco, L.V. & Boiero, D., 2008. Improved Monte Carlo inversion of surface wave data, *Geophys. Prospect.,* **56**(3), 357–371.

Sun, H. & Demanet, L., 2020. Extrapolated full-waveform inversion with deep learning, *Geophysics,* **85**(3), R275–R288.

Tarantola, A., 1984. Inversion of seismic reflection data in the acoustic approximation, *Geophysics,* **49**(8), 1259–1266.

Vantassel, J.P., Kumar, K. & Cox, B.R., 2022. Using convolutional neural networks to develop starting models for near-surface 2-D full waveform inversion, *Geophys. J. Int.,* **231**(1), 72–90.

Virieux, J. & Operto, S., 2009. An overview of full-waveform inversion in exploration geophysics, *Geophysics,* **74**(6), WCC1–WCC26.

Virieux, J., 1986. P-SV wave propagation in heterogeneous media: Velocity-stress finite-difference method, *Geophysics,* **51**(4), 889–901.

Wathelet, M., Chatelain, J.L., Cornou, C., Giulio, G.D., Guillier, B., Ohrnberger, M. & Savvaidis, A., 2020. Geopsy: a user-friendly open-source tool set for ambient vibration processing, *Seismol. Res. Lett.,* **91**(3), 1878–1889.

Wathelet, M., Jongmans, D. & Ohrnberger, M., 2004. Surface-wave inversion using a direct search algorithm and its application to ambient vibration measurements, *Near Surf. Geophys.,* **2**(4), 211–221.

Weiss, K., Khoshgoftaar, T.M. & Wang, D., 2016. A survey of transfer learning, *J. Big Data,* **3,** 1–40.

Wu, Y. & McMechan, G.A., 2019. Parametric convolutional neural network-domain full-waveform inversion, *Geophysics,* **84**(6), R881–R896.

Wu, Y., Lin, Y. & Zhou, Z., 2018, InversionNet: accurate and efficient seismic waveform inversion with convolutional neural networks, in *SEG International Exposition and Annual Meeting,* pp. 2096–2100, SEG.

Xia, J., Miller, R.D. & Park, C.B., 1999. Estimation of near-surface shear-wave velocity by inversion of Rayleigh waves, *Geophysics,* **64**(3), 691–700.

Xia, Y. *et al.* 2003. One-dimensional nanostructures: synthesis, characterization, and applications, *Adv. Mater.,* **15**(5), 353–389.

Yang, F. & Ma, J., 2019. Deep-learning inversion: a next-generation seismic velocity model building method, *Geophysics,* **84**(4), R583–R599.

Yust, M.B., Cox, B.R., Vantassel, J.P., Hubbard, P.G., Boehm, C. & Krischer, L., 2023. Near-surface 2d imaging via fwi of das data: an examination on the impacts of fwi starting model, *Geosciences,* **13**(3), 63.

Zhang, Z. & Alkhalifah, T., 2022. Regularized elastic full-waveform inversion using deep learning, in *Advances in Subsurface Data Analytics,* pp. 219–250, Elsevier.

Zhang, Z. & Lin, Y., 2020. Data-driven seismic waveform inversion: a study on the robustness and generalization, *IEEE Trans. Geosci. Remote Sens.,* **58**(10), 6900–6913.

Zhao, Z. & Sen, M.K., 2021. A gradient-based Markov chain Monte Carlo method for full-waveform inversion and uncertainty analysis, *Geophysics,* **86**(1), R15–R30.

Zhu, M., Feng, S., Lin, Y. & Lu, L., 2023. Fourier-DeepONet: Fourier-enhanced deep operator networks for full waveform inversion with improved accuracy, generalizability, and robustness, *Comput. Meth. Appl. Mech. Eng.,* **416,** 116300

Zhu, W., Xu, K., Darve, E. & Beroza, G.C., 2021. A general approach to seismic inversion with automatic differentiation, *Comput. Geosci.,* **151,** 104751

APPENDIX

In this appendix, we present the results of the MASW inversion for both synthetic and real cases obtained through the Geopsy algorithms. To simplify and avoid misinterpretation of the higher modes, we solely considered the fundamental mode for the inversion. We conducted the inversion for both off-end shots and selected the one yielding better result compared to the true model in the synthetic case and the borehole information in the real case.

Fig. A1 illustrates the results of the inversion for the synthetic case. In Fig. A1(a), a representation of explored models is provided, with colors denoting the computed misfit values corresponding to each associated dispersion curve. These curves are further detailed in Fig. A1(c). It is worth noting that beyond a depth of 18 m, models within a velocity range of 180–350 $\mathrm{m\,s^{-1}}$ explains the observed data, so all of them can be considered plausible solutions. Additionally, Fig. A1(b) presents the velocity spectrum alongside the selected dispersion points and the predicted dispersion curve associated with the model solution. Importantly, the predicted dispersion curve closely aligns with the selected dispersion points.

Fig. A2 presents the results of the MASW inversion for the real case. In Fig. A2(a), the ensemble of explored models is depicted, with colors representing the misfit values computed with each associated dispersion curve, which are further illustrated in Fig. A2(c). Note that beyond 16 m, all proposed models featuring a layer boundary within the range of 14–22 m effectively explains the observed dispersion curve, exhibiting a low misfit. Fig. A2(b) presents the velocity spectrum with the picked dispersion points and the predicted dispersion curve associated with the model solution, note that the predicted dispersion curve effectively matches the picked dispersion points. It is noteworthy that the predicted dispersion curve closely aligns with the picked dispersion points.

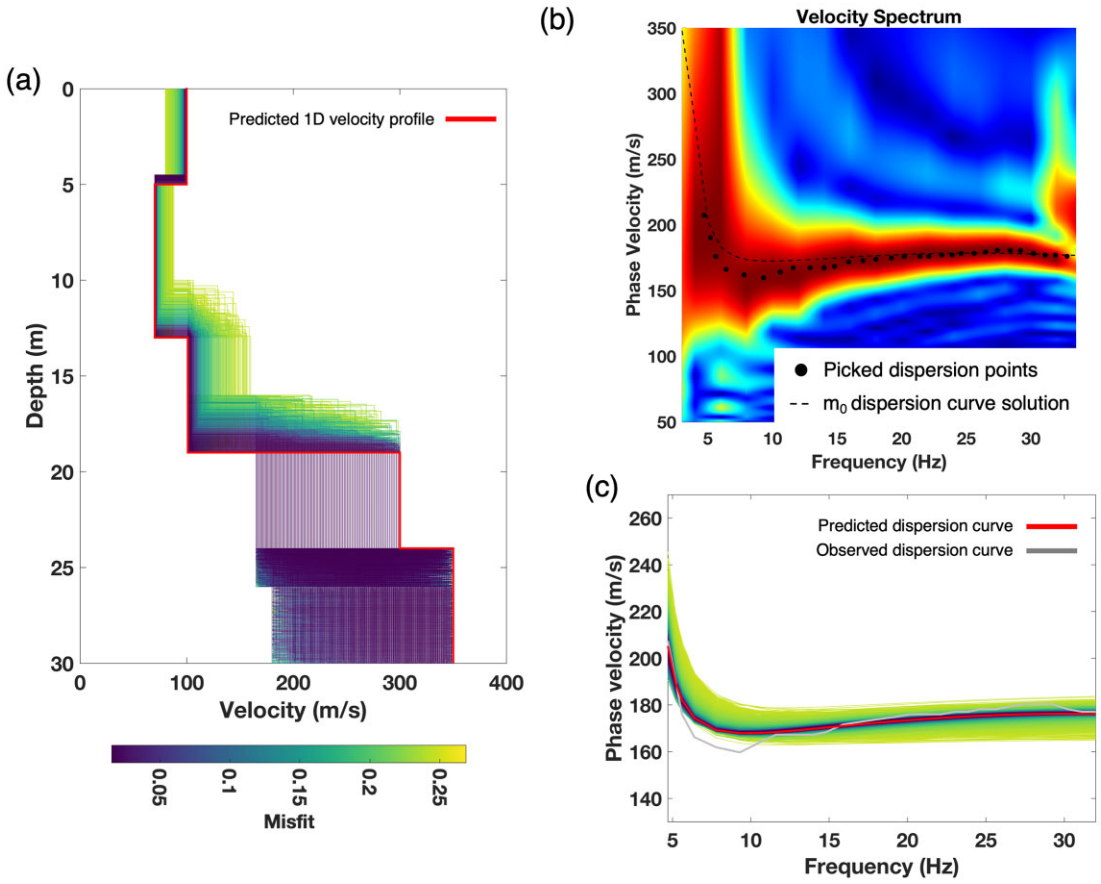

Figure A1. MASW inversion results for the synthetic case. (a) Sampled 1-D velocity models. (b) Velocity spectrum with the picked dispersion points and the predicted dispersion curve. (c) Sampled dispersion curves derived from the velocity models in panel (a).

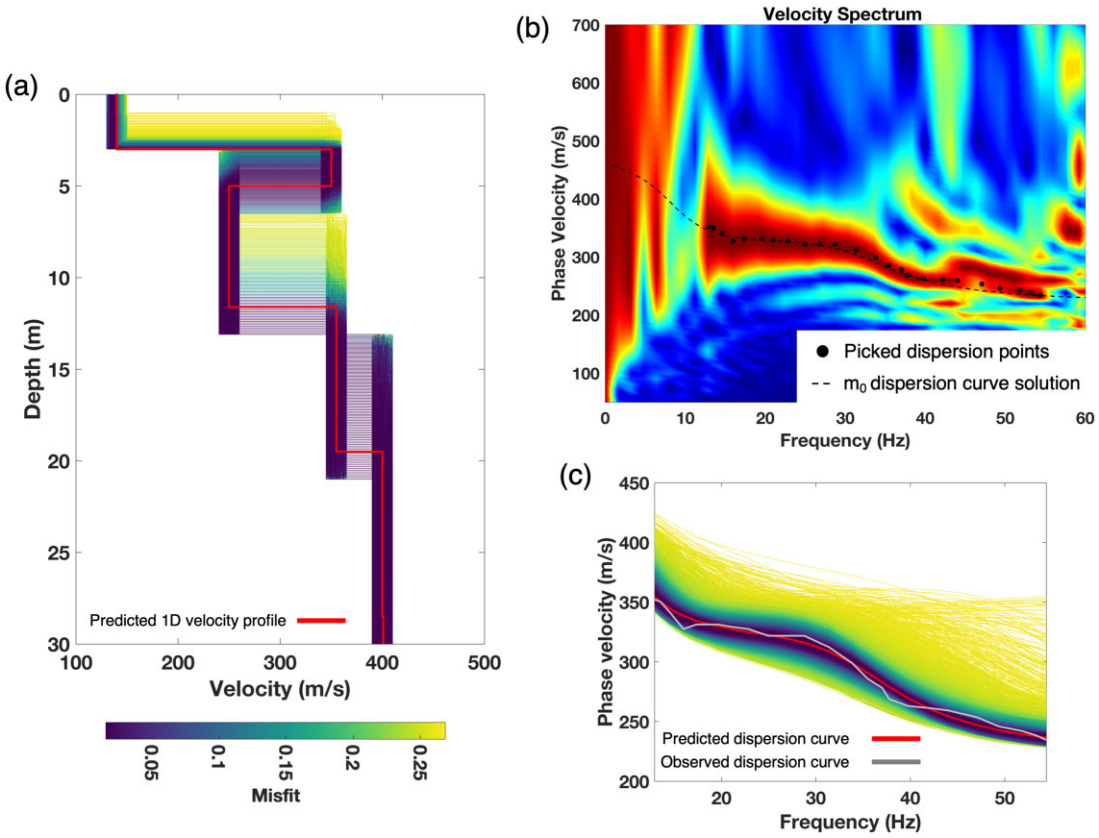

Figure A2. MASW inversion results for the real case. (a) Sampled 1-D velocity models. (b) Velocity spectrum with the picked dispersion points and the predicted dispersion curve. (c) Sampled dispersion curves derived from the velocity models in panel (a).