

Using Deep Learning to Derive Shear-Wave Velocity Models from Surface-Wave Dispersion Data

Jing Hu¹, Hongrui Qiu², Haijiang Zhang^{*1,3}, and Yehuda Ben-Zion²

Abstract

We present a new algorithm for derivations of 1D shear-wave velocity models from surface-wave dispersion data using convolutional neural networks (CNNs). The technique is applied for continental China and the plate boundary region in southern California. Different CNNs are designed for these two regions and are trained using theoretical Rayleigh-wave phase and group velocity images computed from reference 1D V_S models. The methodology is tested with 3260 phase–group images for continental China and 4160 phase–group images for southern California. The conversions of these images to velocity profiles take ~23 s for continental China and ~30 s for southern California on a personal laptop with the NVIDIA GeForce GTX 1060 core and a memory of 6 GB. The results obtained by the CNNs show high correlation with previous studies using conventional methods. The effectiveness of the CNN technique makes this fast method an important alternative for deriving shear-wave velocity models from large datasets of surface-wave dispersion data.

Cite this article as Hu, J., H. Qiu, H. Zhang, and Y. Ben-Zion (2020). Using Deep Learning to Derive Shear-Wave Velocity Models from Surface-Wave Dispersion Data, *Seismol. Res. Lett.* **91**, 1738–1751, doi: [10.1785/0220190222](https://doi.org/10.1785/0220190222).

Supplemental Material

Introduction

Surface-wave tomography has been widely used to image Earth structures at various scales in different tectonic regions. Surface-wave dispersion curves are utilized mainly to determine shear-wave velocity (V_S) models, but are also sensitive to density and compressional-wave velocity (V_P) models (Curtis *et al.*, 1998; Zhou *et al.*, 2006; Lin *et al.*, 2009; Liu *et al.*, 2018). In general, surface-wave tomography adopts a two-step approach: group- or phase velocity maps at different periods are first determined, and then a series of 1D V_S models beneath each grid node are inverted using phase and/or group velocity values at that node. A linearized inversion approach requires selecting optimum regularization values and an appropriate initial velocity model to stabilize the inversion (e.g., Herrmann, 2013). Nonlinear methods based on random sampling (Mosegaard and Tarantola, 1995; Sambridge, 1999a,b) have been used to directly invert for 1D V_S models from dispersion curves. This can avoid selecting regularization parameters but may yield solutions based on incomplete sampling and could be time consuming. The increasing rate and size of deployed local and regional seismic arrays (e.g., Lin *et al.*, 2012; Ben-Zion *et al.*, 2015) produce ever larger data volumes of phase and group velocity measurements extracted from natural earthquakes (e.g., Yang and Ritzwoller, 2008), ambient noise (e.g. Shapiro *et al.*, 2005; Roux *et al.*, 2016), and artificial sources (e.g., She *et al.*, 2018). This makes classical 1D V_S inversions extremely laborious and time consuming. Another alternative is to use

artificial neural networks (Devillee *et al.*, 1999; Meier *et al.*, 2007). Compared to the conventional linear or nonlinear methods, once the network is trained, it can be used to map surface-wave dispersion measurements to 1D V_S models directly without inversion.

In recent years, deep-learning techniques, and especially convolutional neural network (CNN) algorithms, have shown significant potential in various seismological applications including event detection (Perol *et al.*, 2018; Yu *et al.*, 2018), phase picking (Ross *et al.*, 2018; Zhu and Beroza, 2018; Wang *et al.*, 2019), earthquake early warning (Li *et al.*, 2018), first-motion polarity determination (Ross *et al.*, 2018), and seismic phase association (McBrearty *et al.*, 2019; Ross *et al.*, 2019). Only a few studies applied neural networks to surface-wave tomography (Devillee *et al.*, 1999; Meier *et al.*, 2007; Cheng *et al.*, 2019). As noted by Meier *et al.* (2007), there are typically three major steps to solve the inverse problem with the neural network method. (1) Assemble a large amount of 1D V_S models (labels) and the corresponding phase and group velocity dispersion curves (inputs) for training the network. (2) Design a neural network

1. Laboratory of Seismology and Physics of Earth's Interior, School of Earth and Space Sciences, University of Science and Technology of China, Hefei, Anhui, China;

2. Department of Earth Sciences, University of Southern California, Los Angeles, California, U.S.A.; 3. Mengcheng National Geophysical Observatory, University of Science and Technology of China, Hefei, Anhui, China

*Corresponding author: zhang11@ustc.edu.cn

© Seismological Society of America

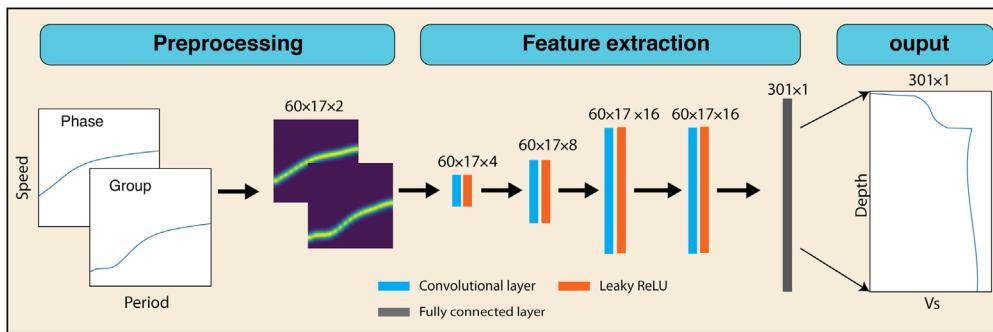


Figure 1. Data preprocessing and the architecture of the convolutional neural network (see the [Application in Continental China](#) section). In the [Application in Southern California](#) section, the number of output elements of convolutional neural network (CNN) is changed from 301 to 99 but other parameter setups are the same. The numbers of trainable parameters of CNNs are 4,916,481 for continental China (see the [Application in Continental China](#) section) and 1,619,639 for Southern California (see the [Application in Southern California](#) section). The color version of this figure is available only in the electronic edition.

structure, which takes phase and group velocities as inputs and outputs 1D V_S models. (3) Train the designed neural network by minimizing the differences between its outputs and the labels. Once a neural network is trained, the best-fitting 1D V_S models are predicted based on the Rayleigh-wave phase and group velocity dispersion curves.

In this study, we present two different CNNs used to perform surface-wave tomography for two different regions including continental China and southern California (Fig. 1), which target different depth ranges (0–150 km for continental China and 0–49.5 km for southern California, respectively). Our analyses differ from previous studies using neural networks (e.g., [Meier et al., 2007](#); [Cheng et al., 2019](#)) in four main aspects. (1) We use CNNs rather than the shallow fully connected neural network adopted in [Meier et al. \(2007\)](#), which can deal with more complicated nonlinear inverse problems. (2) We construct 1D V_S models using finer layers (0.5 km layer thickness) compared to [Meier et al. \(2007\)](#) and [Cheng et al. \(2019\)](#) where the models involved only five major layers (a sedimentary layer, three crustal layers, and an upper mantle layer). (3) We employ both phase and group velocities as [Meier et al. \(2007\)](#) did, whereas [Cheng et al. \(2019\)](#) utilized only phase velocities. (4) Phase and group velocity dispersion curves are converted to 2D images, which are taken as inputs for our CNNs (the preprocessing part in Fig. 1), whereas [Meier et al. \(2007\)](#) concatenated phase and group velocity dispersion curves to create 1D time series as inputs for their fully connected neural network. The pair of phase and group 2D dispersion maps is used as two dependent channels and is easily utilized by 2D convolutional operation in our CNNs to extract features.

Because the variation of different geologically relevant structures is limited, training the CNN on a relatively small amount of datasets containing main geological structures can

be used to perform inversions with the proposed methodology in many regions. For instance, [Eymold and Jordan \(2019\)](#) performed cluster analysis of 1D V_S models in southern California with the k -means algorithm. They classified the entire set of V_S models in the complicated plate boundary region into seven major classes that were shown to correlate with geological structures. Similar analyses in other regions can enable the wide use of CNNs that are trained with just a subset of large datasets.

The reminder of the article is organized as follows. In the

[Methodology](#) section, we describe the methodology and demonstrate the process using an example training dataset from the central western United States. This includes data preprocessing steps, the CNN architecture, and a training process. Next, we apply the method to datasets obtained from continental China (see the [Application to Continental China](#) section) and southern California (see the [Application to Southern California](#) section). For the application in southern California, we use a CNN with a slightly different architecture from the one illustrated in the [Methodology](#) section, which is trained using a dataset generated based on the regional model of [Shaw et al. \(2015\)](#). In the [Discussion and Conclusions](#) section, we discuss and summarize the results.

Methodology

We utilize one of the most widely used deep-learning algorithms, the CNN, to directly invert surface-wave phase and group velocity dispersion curves for isotropic 1D V_S models. Compared to conventional fully connected networks, CNNs are regularized networks with two advantages, sparse connectivity and sharing weights among convolutional layers, which can reduce the machine capacity and improve their generalization ability ([LeCun and Bengio, 1995](#); [LeCun et al., 2015](#); [Goodfellow et al., 2016](#)). First, we take a set of derived 1D V_S models and calculate corresponding theoretical Rayleigh-wave phase and group velocity dispersion curves and corresponding images as the training dataset (preprocessing step in Fig. 1). Then, a designed CNN takes pairs of phase-group dispersion images as inputs and provides outputs of 1D V_S profiles. The differences between the predicted V_S models and corresponding V_S models labeled in the training dataset are minimized to train the CNN (updating the weights of the CNN). The trained CNN can be used to quickly map large

amounts of Rayleigh-wave phase and group dispersion curves to 1D V_S models. Because the output depth range needs to be changed with the period range of the input Rayleigh-wave velocity dispersions, we use two different CNNs for the applications in continental China and southern California. The period ranges are 8–50 s for continental China at depths of 0–150 km (see the [Application to Continental China](#) section) and 2.5–16 s for southern California at depths of 0–49.5 km (see the [Application to Southern California](#) section). For a brief introduction of the data preparation, neural network architecture, and training process, we use a training dataset generated from surface-wave tomography results for the central western United States of [Shen et al. \(2013\)](#) and a test dataset of Rayleigh-wave velocity dispersions for continental China from [Shen et al. \(2016\)](#).

Data preparation and preprocessing

The data diversity is important for training a neural network ([Deng et al., 2009](#)). We first extract 6803 1D V_S models for the central western United States from the surface to a depth of 150 km with each layer thickness of 0.5 km (Fig. S1, available in the supplemental material to this article) based on the surface-wave tomography of [Shen et al. \(2013\)](#). Corresponding V_P models are computed above 120 km from V_S using the relationship of [Brocher \(2005\)](#) and a fixed $V_P/V_S = 1.79$ from 120 to 150 km ([Kennett et al., 1995](#)). Density is also computed from V_S following the empirical relation of [Brocher \(2005\)](#). We then generate corresponding theoretical Rayleigh-wave phase and group velocity dispersion curves for periods in the 8–50 s range (8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 35, 40, and 50 s) via the Computer Programs in Seismology (CPS) software package ([Herrmann, 2013](#)). Considering the good performance of CNNs on image processing, we transform the dispersion curves to energy images (preprocessing part in Fig. 1) via a Gaussian function $g_T(v_0)$

$$g_T(v_0) = e^{-(v-v_0)^2/r}, \quad (1)$$

in which v_0 is the value of phase velocity or group velocity at period T , v is a constant array of 60 elements that varies from 2 to 5 km/s with an interval of 0.05 km/s, and r is the radius of the Gaussian function. The velocity range and interval can be changed according to the specific training dataset, and here we set r as 0.1 km/s. After converting dispersion curves into images, we obtain 6803 pairs of phase-group dispersion images with a dimension of 60 × 17 (height = 60; width = 17). Those transformation parameters will be discussed further in the [Application to Continental China](#) section.

The CNN architecture

The CNN used in this study comprises four convolutional layers and one fully connected layer (Fig. 1). The CNN has two input channels that take phase and group dispersion images

and outputs a best-fitting 1D V_S model. For the continental China case, the input images have dimensions of 60 × 17 and the output 1D V_S profile is discretized into 301 layers. The number of filters for each layer is set following the VGG16 network ([Simonyan and Zisserman, 2014](#)). We double the number of filters for convolutional layers and keep the last two convolutional layers the same. The numbers of filters at each convolutional layer are, from shallow to deep, 4, 8, 16, and 16. For each convolutional layer, we set the kernel size equal to 3 and stride equal to 1, and apply a zero-padding operation at each convolutional layer to keep the spatial resolution after convolution ([Simonyan and Zisserman, 2014](#)). To further avoid the vanishing gradient problems, activation function leaky Rectified Linear Unit (ReLU) ($f(x) = x$, if $x > 0$; otherwise $0.01x$) is applied at each activation layer located right after the convolutional layer (red bars in Fig. 1), which can accelerate training convergence compared to ReLU ([Maas et al., 2013](#)). [Xu et al. \(2015\)](#) compared rectified activations widely used in CNNs, and there was no significant effect on the performance of CNNs. Because the inputs dimension is small, we do not employ a pooling layer that is often included in a conventional CNN architecture.

Mapping phase-group dispersion images to 1D V_S profiles belongs to a regression task. The weights of the neural network are optimized to minimize root mean square error between predictions and labels using the training dataset. The loss function used to train the CNN is defined as

$$J = \frac{1}{M} \sum_{i=1}^M \sqrt{\frac{1}{N} \sum_{j=1}^N (\tilde{m}_j^i - m_j^i)^2}, \quad (2)$$

in which M is the number of V_S models for a batch, N is the number of layers of a V_S model, \tilde{m}_j^i and m_j^i are the predicted and ground-truth V_S values for i th V_S model at j th layer, respectively.

Training

We randomly split 80% of the whole dataset as the training dataset and 20% as the validation dataset. The validation dataset is excluded from the training process and only used to guide parameter tuning and avoid overfitting. The maximum number of epochs is set to 600 to ensure the training process converges. For each epoch, we randomly shuffle the whole input dataset to decrease the risk of creating batches that are not representative of the overall training dataset. We use the adaptive moment estimation (Adam) optimizer with a learning rate of 1×10^{-5} and other parameters set by default to minimize the loss function ([Kingma and Ba, 2014](#)). We initialize weights with uniform distribution and use a batch size of 64 considering a trade-off between efficiency and generalization performance ([Keskar et al., 2016](#)). The training parameter setups are the same for both the [Application to Continental China](#) and [Application to Southern California](#) sections.

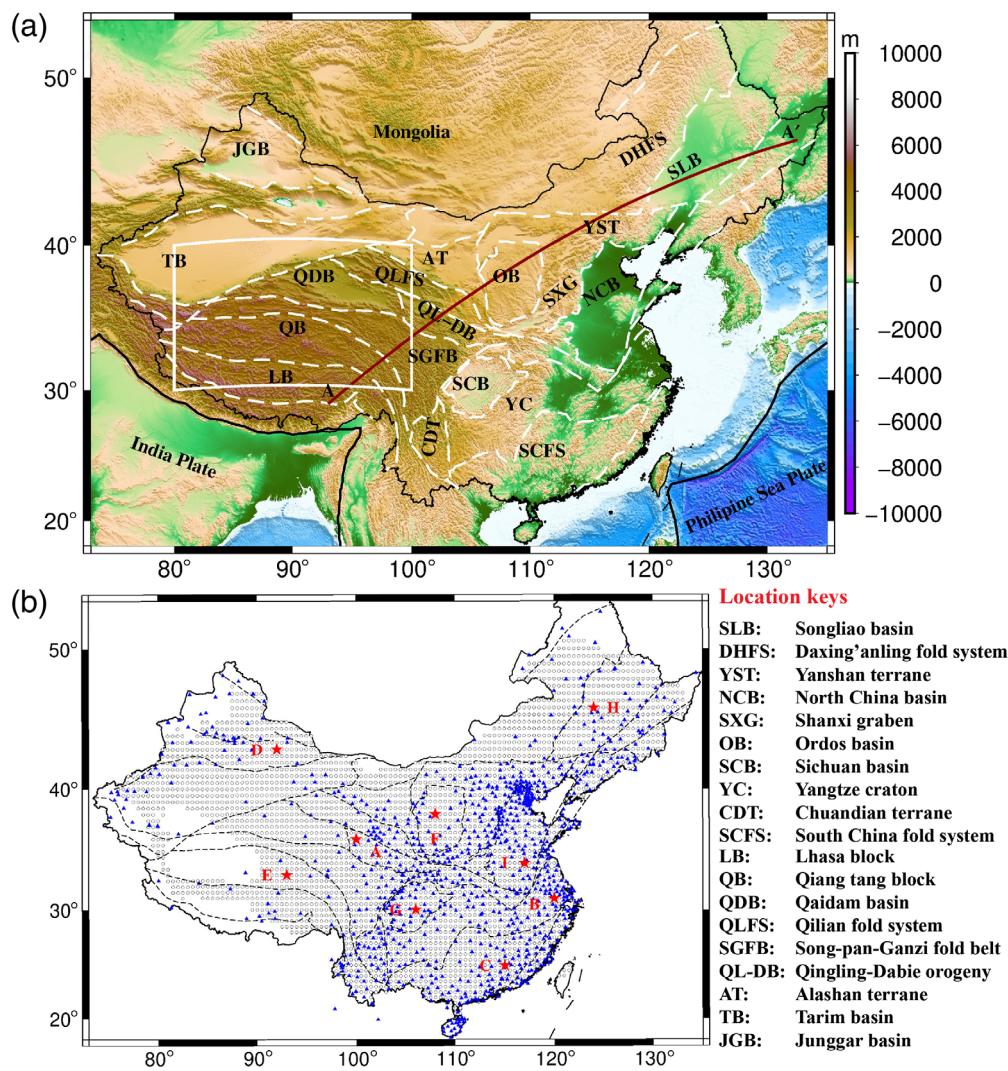


Figure 2. (a) Map view of the surface topography and major tectonic features of China and surrounding area. White dashed lines outline the main tectonic units and basins, and black bold lines indicate the plate boundary. The dark red line denotes A-A' profile (shown in Fig. 4). The white box outlines the region in which the 675 1D V_S models of [Shen et al. \(2016\)](#) are extracted to build up the training dataset for *Test2*. Location keys are shown in the bottom right. (b) Map view of the area covered by the test dataset (circles). Red stars mark grid nodes that are used to demonstrate the comparison between the observed and predicted Rayleigh-wave dispersion curves in Figure 5. Blue triangles are permanent stations used to extract dispersion measurements in [Shen et al. \(2016\)](#). For the map view of all stations used to extract dispersion measurements, please refer to [Shen et al. \(2016\)](#). The color version of this figure is available only in the electronic edition.

Application to continental China

In this section, we conduct two experiments, named *Test1* and *Test2*, to study the impact of the training dataset on the CNN performance. For *Test1* and *Test2*, we employ two different training datasets to separately train the CNNs with same control parameters (learning rate, batch size, and so on). Then, the trained CNNs of *Test1* and *Test2* are used to invert V_S models using the actual Rayleigh-wave dispersion curves measured in continental China. The predicted V_S models of *Test1* and *Test2* are evaluated by comparing with the V_S model of [Shen et al. \(2016\)](#).

The training dataset of *Test1* comprises 6803 1D V_S models (Fig. S2) extracted from the central western United States tomographic model of [Shen et al. \(2013\)](#) and the corresponding theoretical Rayleigh-wave phase and group dispersion images (the [Data Preparation and Preprocessing](#) section). For the training dataset of *Test2*, we augment *Test1* training dataset with additional 675 1D V_S models (Fig. S2) extracted from the Tibet region (Fig. 2a) results of [Shen et al. \(2016\)](#) and the corresponding theoretical dispersion images. [Shen et al. \(2016\)](#) measured the Rayleigh-wave group and phase velocity dispersion curves for a period range of 8–50 s in continental China. Those dispersion measurements were used to determine a 3D V_S model for the top 150 km of continental China via a Bayesian Monte Carlo inversion. Here, we use the dispersion measurements of [Shen et al. \(2016\)](#) as the test dataset for *Test1* and *Test2*. Both phase and group velocity dispersion curves are required to be within the period range of 8–50 s at each grid node. Figures S3 and S4 show phase and group velocities at periods of 10, 30, and 40 s from [Shen et al. \(2016\)](#), respectively, together with their corresponding uncertainties for all available grid nodes. We linearly interpolate those phase

and group velocity dispersion curves and generate phase–group dispersion images based on equation (1). A total of 3260 pairs of phase–group velocity dispersion images are produced for the test dataset of continental China, which cover most of continental China (Fig. 2b).

The training losses as a function of epochs show a dramatic decrease (from ~4 to ~0.15 km/s) after the first 15 epochs for both tests (Fig. S5a,b). After 600 epochs, the final losses converge to ~0.05 km/s, and we take the CNNs at the epoch 600 as the final trained CNNs for both tests. These trained CNNs

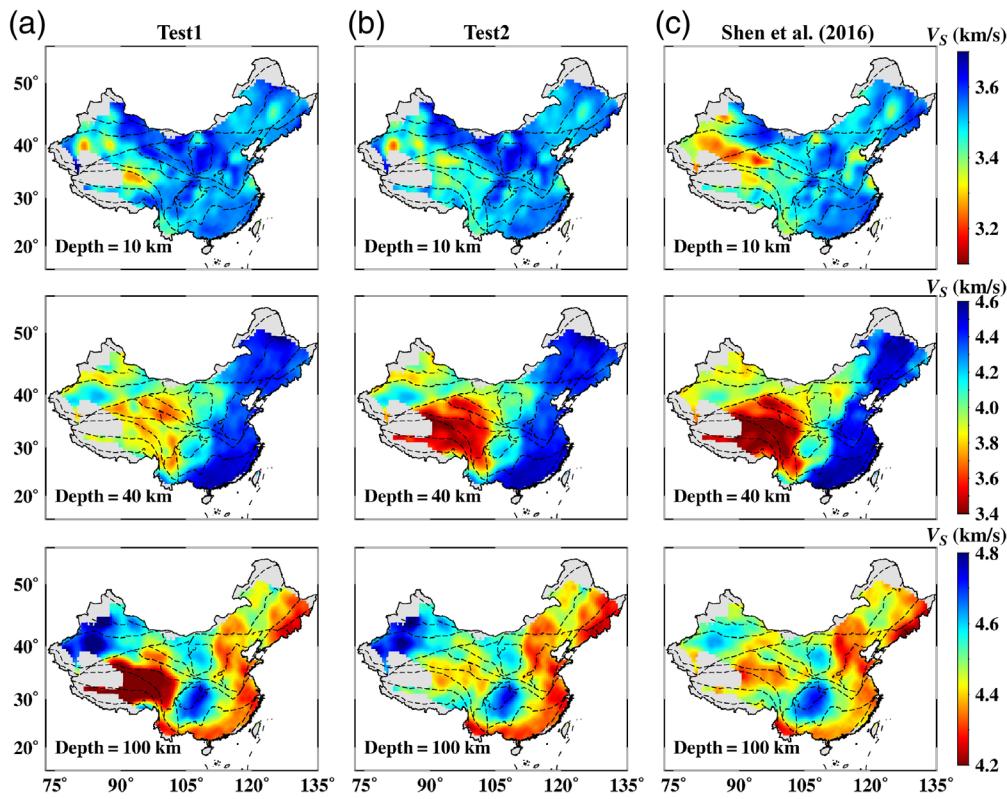


Figure 3. Comparison of V_S depth slices obtained from (a) *Test1*, (b) *Test2*, and (c) *Shen et al. (2016)* at 10 (top), 40 (center), and 100 km (bottom), respectively. Thick dashed lines delineate the tectonic units. The color version of this figure is available only in the electronic edition.

are then used to estimate 1D V_S models from dispersion measurements for continental China. For both *Test1* and *Test2*, the wall time to train the CNN is ~ 1.5 hr and the wall time to predict 3260 1D V_S models using the trained CNN is ~ 23 s. The CNN-based inversion is much more computationally efficient than the Bayesian Monte Carlo inversion used in *Shen et al. (2016)*, which usually requires more than 200 computing hours to sequentially invert the same test dataset given that it needs ~ 4 min for each single grid node. Those tests are on a personal laptop with the NVIDIA GeForce GTX 1060 core and a memory of 6 GB. The same evaluation rules are for the [Application to Southern California](#) section.

Although the exact V_S distribution of continental China is unknown, we can take the 3D V_S model of *Shen et al. (2016)* as the baseline model and evaluate results from *Test1* and *Test2*. The CNN training dataset for *Test2* includes V_S information from the Tibet region, which is significantly different compared to the V_S model of the central western United States in *Shen et al. (2013)*. Figure 3 shows comparisons of V_S distributions given by *Test1*, *Test2*, and *Shen et al. (2016)* at depths of 10, 40, and 100 km. The results of *Test1* and *Test2* show high similarity to the baseline 3D V_S model in eastern, southern, northern, and northeast China, especially for several main tectonic units including SCB, OB, NCB (see keys in

Fig. 2). For JGB and TB, the V_S values from *Test1* and *Test2* are systematically larger than those of the *Shen et al. (2016)* for all three depths. In the Tibet region, however, V_S values from *Test1* are larger than those of the baseline model at the depths of 40 and 100 km, whereas V_S values from *Test2* are close to those of *Shen et al. (2016)* at all three depths. This is because the 1D V_S profiles in Tibet region are significantly different from those of the training dataset used to train the CNN in *Test1* (Fig. S2).

Furthermore, we compare V_S models from *Test1* and *Test2* to results of *Shen et al. (2016)* along a vertical cross section A-A' at a depth range of 5–120 km (Fig. 4), which crosses SLB, OB, and Tibet region. We exclude the top 5 km V_S structures due to the lack of sensitivity at shallow depths as illustrated in *Shen et al. (2016)*. Overall, the V_S

distributions of all three cross sections are similar except for the Tibet region. In general, the crustal thickness (the areas marked by red-yellow color in Fig. 4) decreases from Tibet to SLB for all three cross sections, which is consistent with previous imaging results (*He et al., 2014; Xin et al., 2019*).

To quantify the quality of the model given by the CNN, we examine the data fitting between the input and model-predicted dispersion curves. For each location, the data misfit is defined as

$$\text{Misfit} = \left[\frac{1}{N} \sum_{i=1}^N \left(\frac{d_i^{\text{obs}} - d_i^{\text{cal}}}{\sigma_i} \right)^2 \right]^{\frac{1}{2}}, \quad (3)$$

in which d_i^{obs} is the observed Rayleigh-wave phase or group velocity, d_i^{cal} is the value predicted by the forward model, σ_i is the uncertainty of the phase or group velocity, i is the index of the discrete phase and group velocity measurements, and N is the number of dispersion measurements.

Figure 5 shows the resulting dispersion fittings at two representative grid nodes located in Tibet region and OB for both tests. The 1D V_S profiles (*Test1*, *Test2*, and baseline model of *Shen et al., 2016*) and corresponding dispersion curves are also shown for comparison. For each grid node, we perform 200 bootstrap tests (shaded areas of right column plots in Fig. 5).

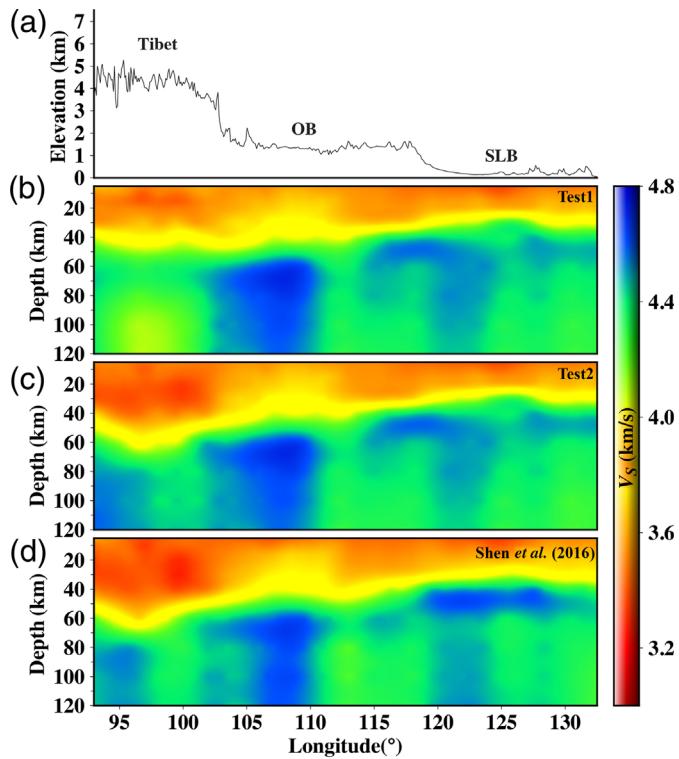


Figure 4. Comparison of V_s vertical cross sections along profile A–A' for *Test1*, *Test2*, and *Shen et al.* (2016). (a) Topography along A–A' profile. OB and SLB represent Ordos and Songliao basins, respectively (Fig. 2a). (b) V_s model of *Test1*; (c) V_s model of *Test2*; and (d) V_s model of *Shen et al.* (2016). The color version of this figure is available only in the electronic edition.

For each test, perturbations ($\delta\nu$) are imposed to the corresponding phase and group dispersion curves at 8–50 s. Perturbations are selected randomly from the uniform probability distribution $\delta\nu \sim U(-\sigma, +\sigma)$, in which σ is the corresponding uncertainty at each period. The perturbated phase and group velocity curves are converted to 2D colormaps (equation 1), which are used as inputs to the CNN to predict a 1D V_s profile. The V_s profile uncertainties (gray lines of right column plots in Fig. 5) are calculated as the standard deviation of all the bootstrap results. The misfits at the E th grid node (upper panels of Fig. 5a,b) for *Test1*, *Test2*, and the baseline model are 9.29, 1.06, and 1.05, respectively. The misfits at the F th grid node (bottom panels of Fig. 5a,b) for *Test1*, *Test2*, and the baseline model are 0.87, 0.73, and 1.93, respectively. Both models from *Test1* and *Test2* at the F th grid node in OB yield smaller misfit values than the model of *Shen et al.* (2016), whereas the model from *Test1* fails to fit the dispersion measurements at the E th grid node in the Tibet region. This indicates that the CNNs give inaccurate estimate of 1D V_s models when a biased V_s distribution is assumed in the training process. More examples of such dispersion fitting are shown in Figures S6 and S7.

The spatial distribution and histogram of dispersion misfits for *Test1*, *Test2*, and baseline model of *Shen et al.* (2016) are shown in Figure 6a,b. The average misfit values are 2.92 for *Test1*, 1.97 for *Test2*, and 1.89 for the baseline model (Fig. 6b). The average misfit value of the baseline model is larger than ~0.76 reported in *Shen et al.* (2016). This is because *Shen et al.* (2016) used a different scaling relationship to obtain V_p and density values from V_s , and investigated a larger amount of data than that used in this study.

As shown in Figure 6a, regions in the eastern part of continental China show generally small misfit values (<2), suggesting the dispersion curves are well fitted for both *Test1* and *Test2*, whereas the dispersion data are poorly fitted in *Test1* for the Tibet region. In the Chuandian terrane (CDT) region, the misfit of *Test2* is less than that of *Test1*, indicating that the CNN of *Test2* provides better estimations of the V_s structure there. Because the V_s distribution of the CDT region is not included in the training datasets for both tests, this likely suggests that the diversity of 1D V_s profiles in the Tibet region is sufficient to represent structure complexities in the CDT region. The misfits for the TB region (Fig. 6a) are relatively larger than other regions due to the larger uncertainties of phase and group velocity measurements (right column plots of Figs. S3 and S4).

We also investigate the impact of the selection of the training dataset on the CNN performance. To perform a statistical analysis, we split the dataset in the same way as that of the **Training** section. The randomly splitting dataset process is performed 15 times to produce different training datasets, which are then subsequently used to train 15 different CNNs, respectively. The trained CNNs are independently used to estimate 1D V_s models with the entire test dataset of *Test1*. The standard deviation of V_s at each layer is calculated (Fig. S8a,b). The mean standard deviation of V_s values from those CNNs is ~0.06 km/s (Fig. S8c), suggesting that the V_s model given by CNN is insensitive to the selection of the training dataset when 1D V_s profiles are sufficiently accurate.

We investigate further the effect of the size of Gaussian radius r and the number of elements of array v , and the map height in equation (1) on the CNN performance. We again use *Test1* for the demonstration. Here, we retain the training parameters and the CNN structure the same for both test datasets. We set r to be 0.01, 0.1, and 0.5 km/s and the map height to be 10, 60, and 100 to create nine new training datasets, respectively. Correspondingly, nine CNNs are trained separately and are applied to *Test1* test dataset. It is found that there is only a slight difference when using different r values with the map height being 60 (Table S1 and Fig. S9). Therefore, we suggest that $r = 0.1$ and a map height = 60 are appropriate for the applications done in this article.

Finally, we also investigate the effect of the number of layers outputted from the network on the performance of the CNN. The larger number of layers is, the more complex CNN is due

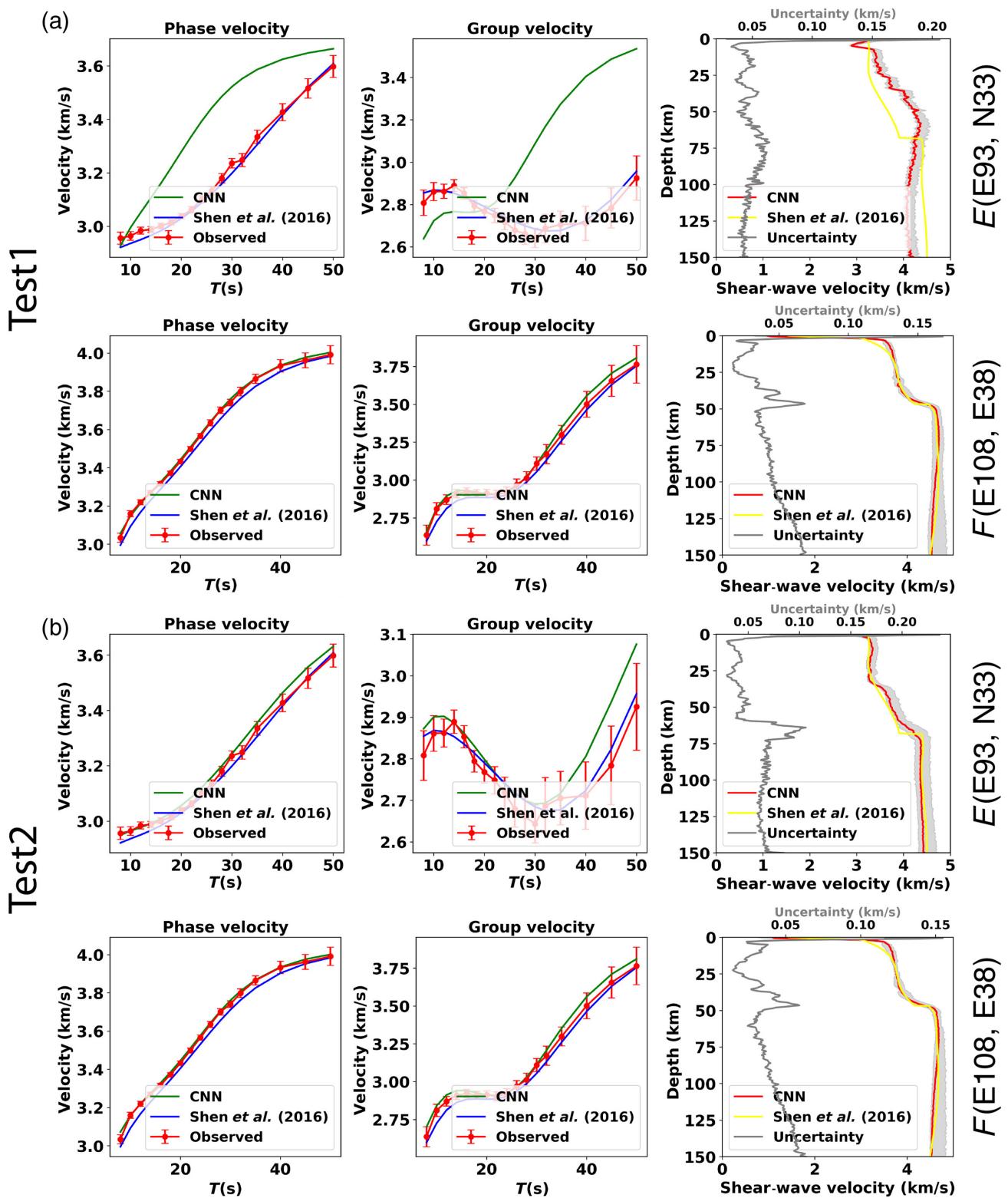


Figure 5. Comparison of observed and predicted dispersion curves for (a) *Test1* and (b) *Test2* at two selected nodes as shown in Figure 3. Left and middle panels show the comparisons of Rayleigh-wave group and phase dispersion curves, respectively. The red line in each panel represents the observed dispersion curve and error bars indicate a range of one standard deviation about each respective mean value. Green and blue lines depict

the predicted dispersion curves from *Test1* and [Shen et al. \(2016\)](#), respectively. Right panels illustrate the comparison of 1D V_s profiles obtained from the CNN-based method (red line) and [Shen et al. \(2016\)](#) (yellow line). Gray shaded areas mark 200 bootstrap results. Gray lines denote the V_s profile uncertainty with one standard deviation of those bootstrap results. The color version of this figure is available only in the electronic edition.

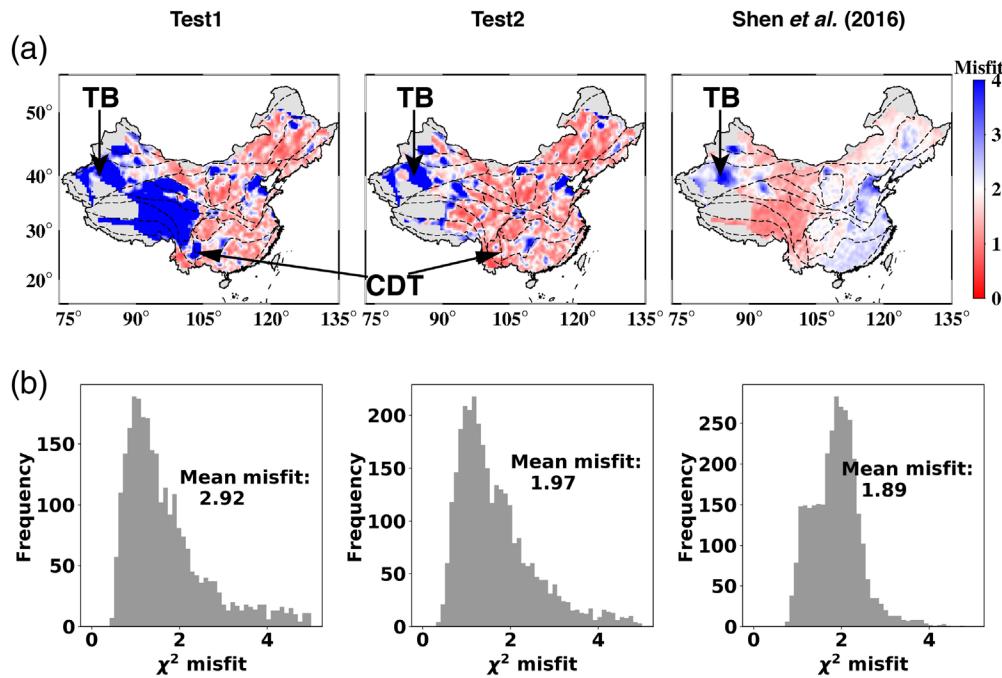


Figure 6. (a) Dispersion misfit maps and (b) the corresponding histograms for (left) *Test1*, (middle) *Test2*, and (right) *Shen et al. (2016)*. We do not compare the misfit maps of *Shen et al. (2016)* with those of *Test1* and *Test2* due to a difference in V_p and density setups between our study and *Shen et al. (2016)*. The color version of this figure is available only in the electronic edition.

to increasing the number of unknown parameters of the last fully connected layer. Here, for *Test1*, we set layer thickness to be 0.1, 0.25, 0.5, and 2 km, which corresponds 1501, 701, 301, and 76 layers for a depth range of 0–150 km, respectively. Table S2 and Figure S10 show that there are no prominent changes regarding the mean misfit of the *Test1* test dataset for the layer thickness of 0.1–0.5 km except for 2 km. It appears that a model with 2 km layer thickness is not fine enough to represent the actual V_s structures, and the complexity of CNN is extremely low comparing to other cases using thinner layers. We retain 0.5 km thickness for each layer in the *Application to southern California* section, considering its relatively low complexity of the CNN and high computational efficient.

Application to southern California

Assuming the same initial 3D V_s model, can the CNN-based tomography perform as well as the traditional methods (e.g., Herrmann, 2013; *Shen et al.*, 2013)? The southern California region provides a good opportunity to answer this question as several dense seismic networks and seismic velocity models (*Lee et al.*, 2014; *Shaw et al.*, 2015) are available in this plate boundary region. Recently, *Qiu et al.* (2019) performed Eikonal tomography for southern California using more than 300 seismic stations and provided isotropic Rayleigh-wave phase and group velocity maps with a grid interval of $0.05^\circ \times 0.05^\circ$ over a period range of 2.5–16 s (Fig. 7). These Rayleigh-wave velocity

maps were jointly inverted at each grid node using the Southern California Earthquake Center Community Velocity Model of *Shaw et al.* (2015; hereinafter, CVMH) as an initial model. The CPS (Herrmann, 2013) was used to obtain a set of 1D V_s profiles for the top 50 km, which are later assembled to construct a pseudo-3D V_s model.

To perform a direct comparison between the CNN-based model and the results of *Qiu et al.* (2019), we use the same Rayleigh-wave phase and group velocity maps of *Qiu et al.* (2019) and depth discretization (from the surface to a depth of 49.5 km with 0.5 km layer thickness). The CNN architecture is the same as that described in the *Application to Continental China* section except for the output dimension size (99 – the number of

layers). For the training dataset, we extract 24,554 1D V_s profiles from the CVMH and generate the corresponding theoretical phase and group velocity dispersion curves for Rayleigh waves over a period range of 2.5–16 s. The V_p and density are also extracted from the CVMH. We then convert the phase and group dispersion curves to 2D images following the processing steps described in the *Data Preparation and Preprocessing* section (equation 1). A velocity range of 1.0–4.5 km/s is used to construct the dispersion images. For the test dataset, phase and group dispersion images are generated using the Rayleigh-wave velocity maps of *Qiu et al.* (2019). Figures S11 and S12 show phase and group velocity maps at periods of 3, 7, and 11 s, respectively, together with their corresponding uncertainties. In total, we have 4160 pairs of Rayleigh-wave phase and group dispersion images as the input test dataset.

The training parameter setups (maximum number of epochs, batch size, learning rate, and so on) are the same as those used in the *Application to Continental China* section. The final loss converges to ~ 0.06 km/s (Fig. S5c). Convergent results are achieved after 600 epochs without overfitting, and the trained network at the epoch 600 is used to estimate 1D V_s models from the input dispersion images. It takes ~ 4.5 hr to train the new CNN and ~ 30 s to generate all the 1D V_s models, whereas more than 30 hr are required to sequentially invert the same dataset in *Qiu et al.* (2019), giving that a single grid inversion with 100 iterations needs ~ 30 s.

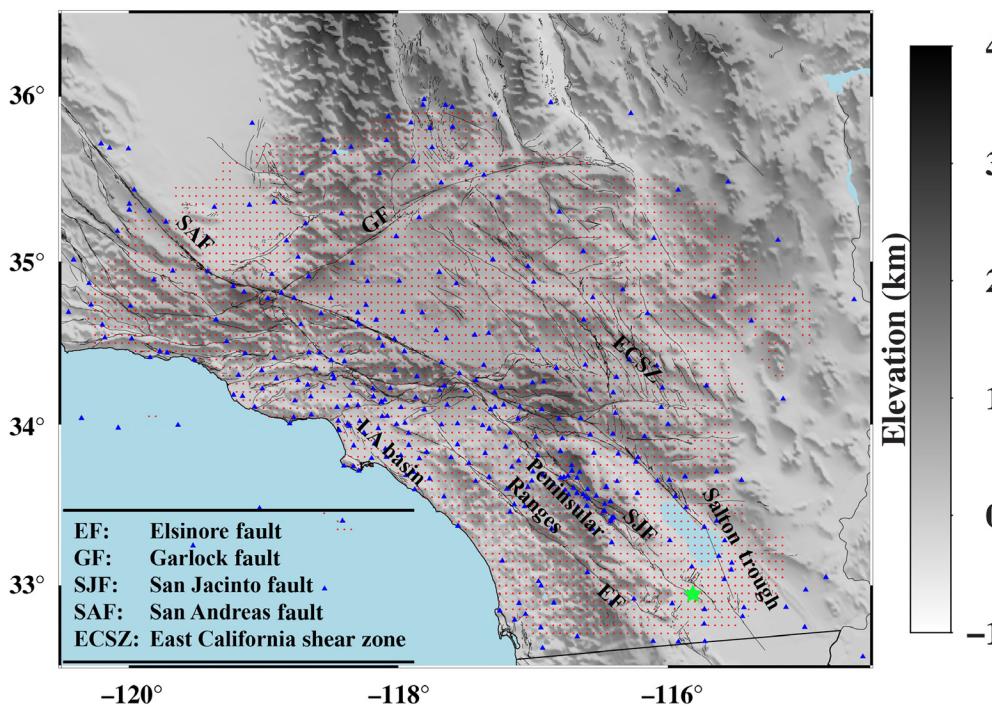


Figure 7. Map view of the southern California plate boundary region. Red dots represent the spatial distribution of the observed Rayleigh-wave phase and group dispersion curves used in this study. Thin black lines denote the main fault surface traces in southern California. The gray-shaded background depicts the topography. Blue triangles mark stations used in Qiu et al. (2019). The lime star marks the selected node that is used to show the comparison between the observed and predicted Rayleigh-wave dispersion curves in Figure 10. The color version of this figure is available only in the electronic edition.

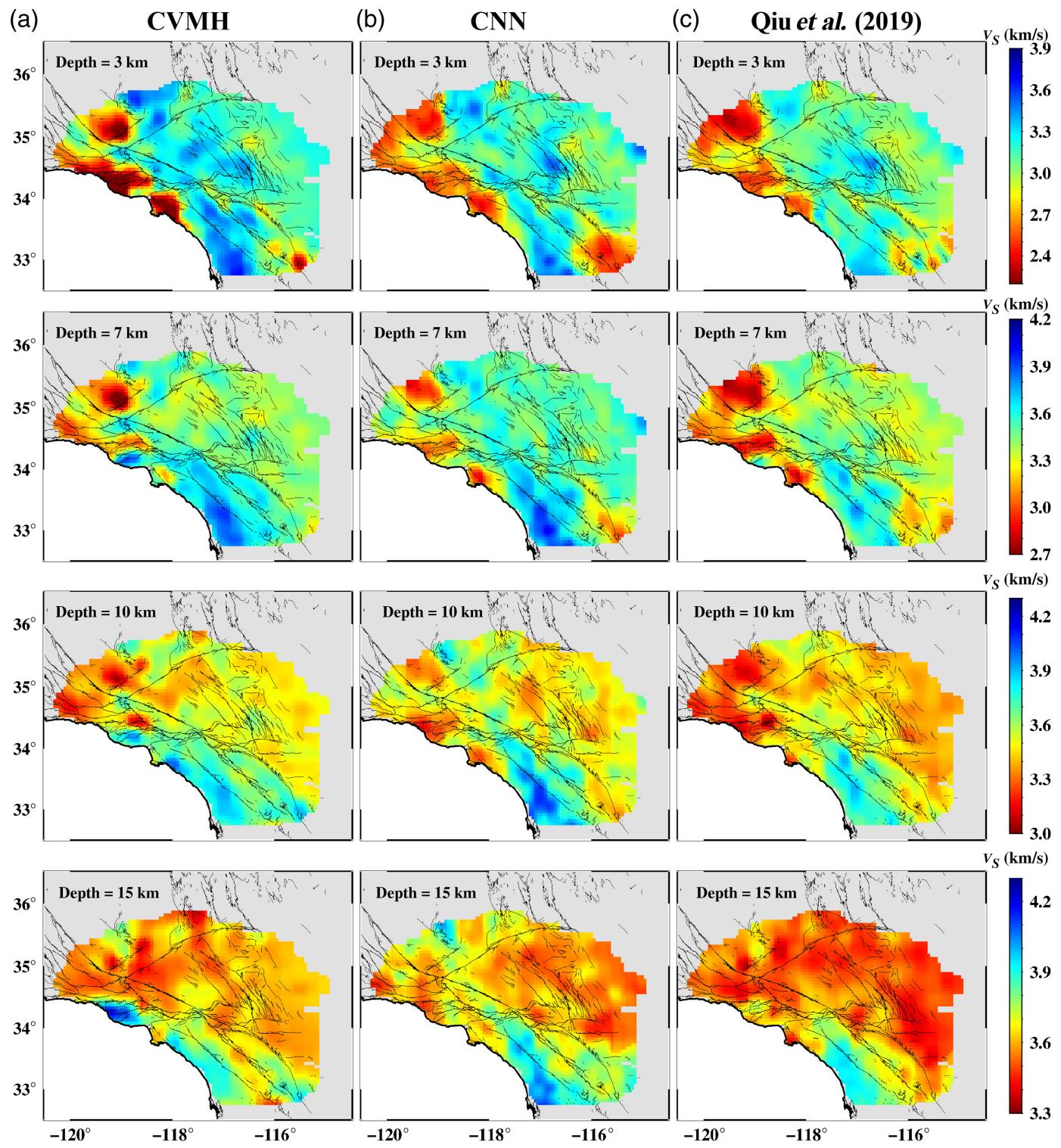
Because of a lack of Rayleigh-wave velocity data at periods shorter than 3 s or longer than 16 s, we cannot resolve the structures in the top 3 km and below 20 km (Qiu et al., 2019). For this reason, Figure 8 only shows the comparison of V_S depth slices between the CNN-based model, results of Qiu et al. (2019), and the initial model of CVMH at depths of 3–15 km. The derived CNN-based model overall shows consistent features compared to Qiu et al. (2019). However, the velocities in the Peninsular Ranges and Salton trough region from the CNN model are higher and considerably lower than those of Qiu et al. (2019), respectively. The large-scale geological features observed in the CNN model are consistent with those of the CVMH model, whereas the variations in velocity values are much larger in the CNN-based model. Figure 9 shows the spatial distributions of Rayleigh-wave velocity dispersion misfit (equation 3) for the CVMH, CNN-based model, and Qiu et al. (2019). The average misfits are 4.54, 1.49, and 1.72 for the CVMH, CNN-based model, and Qiu et al. (2019), respectively. In Qiu et al. (2019), the V_P/V_S ratio and Moho depth are fixed in the inversion and the average misfit is ~0.6, which is smaller than the value obtained in this study. The larger misfit from the CNN-based model is likely caused by different scaling relations between

V_P and V_S used to compute theoretical Rayleigh-wave dispersion curves. In general, large misfits are seen in the Salton trough region relative to other regions in the results of Qiu et al. (2019). However, the misfit values in the Salton trough region is similar to the other regions from the CNN-based model, suggesting the dispersion data there are likely better fitted by the CNN model than Qiu et al. (2019) (Fig. 10). The average misfit of the initial model CVMH is a factor of ~3 larger than that of the CNN-based result, suggesting the CNN-based model significantly improves the fitting of the input dispersion data than the model used to train it. Overall, the CNN-based V_S model fits the input Rayleigh-wave dispersion data better than CVMH and similar to the results of Qiu et al. (2019).

Discussion and Conclusions

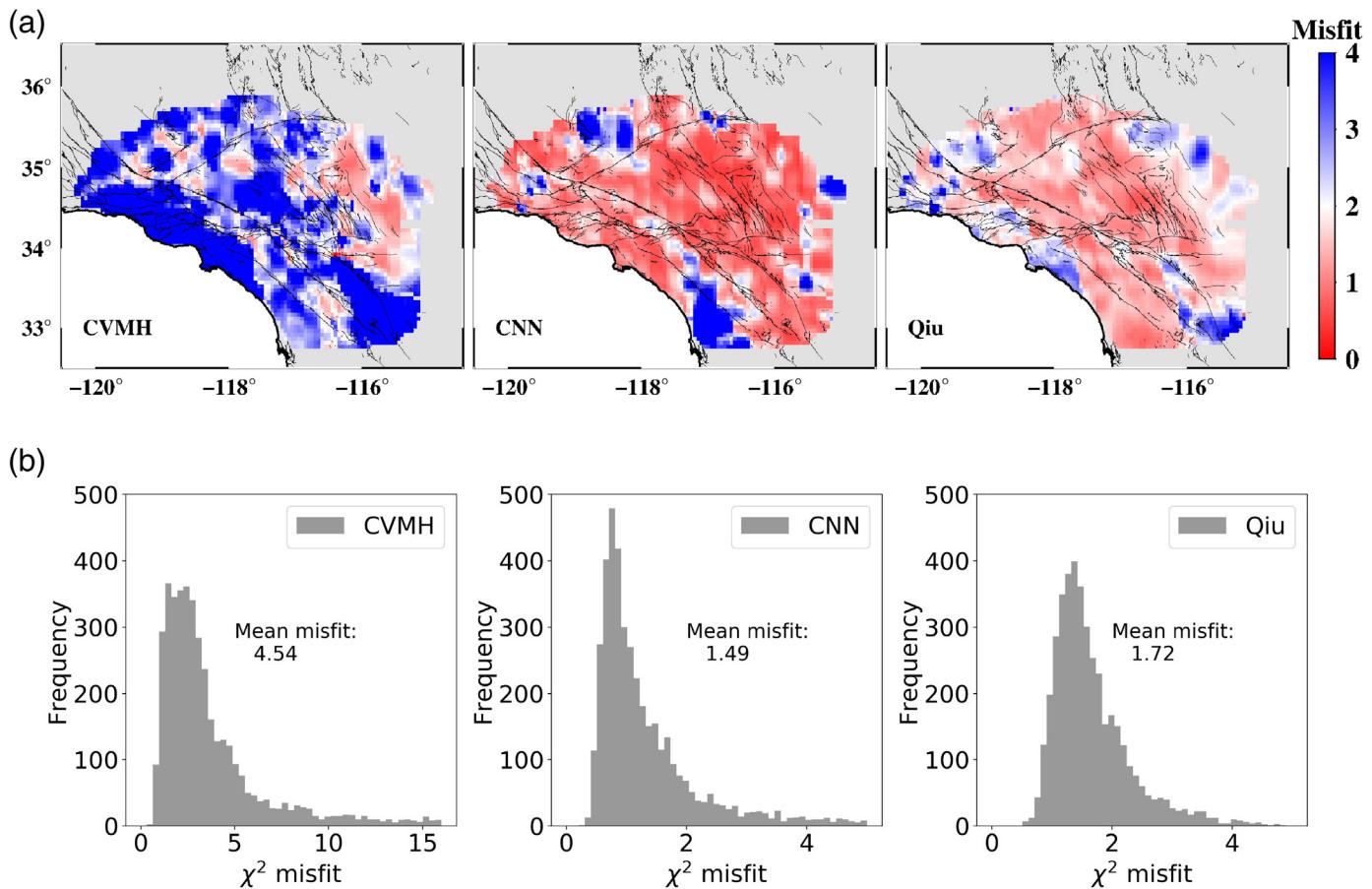
The ongoing significant increase in the number of seismic stations produces big datasets that require fast processing methods. In this study, we demonstrate that properly trained CNNs provide highly effective tools for converting surface-wave dispersion measurements directly to shear-wave velocity models. The CNNs bypass the need to select carefully an appropriate initial model and inversion parameters, and they replace time-consuming nonlinear inversions with rapidly direct mapping of phase and group velocity dispersion curves to model velocity results.

The effectiveness of the proposed CNN-based technique is tested on two different datasets associated with continental China and southern California. Compared with the earlier method of Meier et al. (2007), we use deep neural network that can better represent highly complex velocity models. Different from Cheng et al. (2019) that used deep neural network for surface-wave tomography from phase velocities, we utilize both phase and group velocities as inputs. In addition, in contrast to Meier et al. (2007) and Cheng et al. (2019) that generated dataset based on a reference model with a small number of layers, we use training datasets associated with V_S models derived in previous studies with many layers that are 0.5 km



thick each. The application of the CNN to continental China shows great potential for deriving V_S models using a relatively small training dataset (*Test1* in the [Application to Continental China](#) section). The analysis also demonstrates that increasing diversity in the training dataset enhances the performance of the CNNs (*Test2* in the [Application to Continental China](#) section). The comparison between V_S models from *Test1* and *Test2* and the V_S model from [Shen et al. \(2016\)](#) shows that if the training

Figure 8. Comparison of the V_S depth slices extracted from the (a) Southern California Earthquake Center Community Velocity Model of [Shaw et al. \(2015; CVMH\)](#), (b) CNN-based model, and (c) results of [Qiu et al. \(2019\)](#) at 3, 7, 10, and 15 km. Thin black lines delineate surface traces of main faults in southern California. The thick line represents the coastal line. The color version of this figure is available only in the electronic edition.



dataset does not contain potential structures for the real dataset, the CNN may perform worse than the conventional inversions methods. Fortunately, studies of tomographic cluster analysis indicate that there are many common velocity structures in Earth (Stankiewicz *et al.*, 2010; Lekic and Romanowicz, 2011; Lekic *et al.*, 2012; Braeuer and Bauer, 2015; Flament *et al.*, 2017; Eymold and Jordan, 2019). This allows applications of CNN algorithms trained with a limited subset of different datasets to be used in many geographical regions.

Both the CVMH model and results of Qiu *et al.* (2019) have large dispersion misfits in the Salton trough region. The significant improvement in fitting dispersion measurements in that region (Figs. 9 and 10) using CNN suggests that the CNN-based results are less affected by the initial model than those of classical inversion schemes (Qiu *et al.*, 2019). It is striking that the surface-wave data misfit based on the CVMH is larger than the CNN-based model, given that the training dataset is computed based on CVMH. This is likely because the CNN can learn the implicit relations between V_S models and dispersion curves and may have the ability to store the models associated with the training dataset. When applied to the test dataset, it could use the learned relations to predict V_S models or select the best model among various candidates to fit the test dataset.

For the actual application of the CNN-based method, as in Hou *et al.* (2019), we can first use conventional methods to

Figure 9. (a) Misfit maps and (b) misfit histograms of the CVMH, CNN-based model, and model of Qiu *et al.* (2019). The color version of this figure is available only in the electronic edition.

invert for the coarser V_S model on sparse grid nodes in a large survey region. We can then create a training dataset based on the coarser model to further train a CNN, which is subsequently applied to the whole region with denser grid nodes. This strategy can save much time compared to running a large number of repeated inversions for dense grid nodes and also alleviate artifacts.

Our applications of CNNs to surface-wave tomography employ dispersion curves data over different period ranges for continental China and southern California, targeting a depth range of 0–150 km for continental China and 0–50 km for southern California. The results are consistent with previous studies using conventional methods, but our method is computationally far more efficient. This advantage will become increasingly important as training datasets accumulate and vast new datasets are recorded. Future applications can include monitoring structures (e.g., fault zones, volcanos, and reservoirs) in real time with CNN-based time-dependent tomography using all results at the monitored locations as training datasets.

The model uncertainty associated with the CNN results are difficult to estimate. This is a common problem in deep

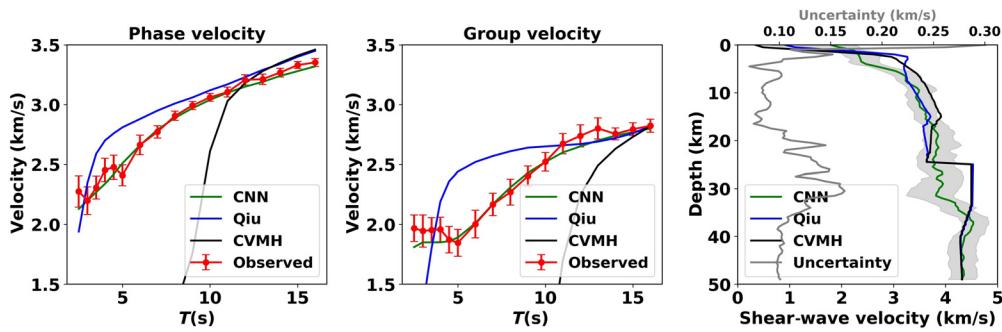


Figure 10. Comparison of the observed and predicted Rayleigh-wave dispersion curves for the CNN-based model, model of Qiu *et al.* (2019), and CVMH at the selected node marked by lime star in Figure 7. Left and middle panels show the comparisons of the observed and predicted Rayleigh-wave group and phase dispersion curves, respectively. The red curve in each panel represents the observed dispersion curve and error bars indicate a range of one standard deviation about each respective mean value. 1D V_S models used to generate predicted dispersion curves are illustrated in the right panel. Green, blue, black, and gray lines depict the dispersion curves predicted by the CNN-based model, Qiu *et al.* (2019), CVMH, and CNN-based model uncertainty, respectively. The shaded area is 200 CNN-based V_S models by bootstrap test performed in the same way as Figure 5 in the Application in Continental China section. The misfit values are 0.78 for the CNN-based model, 3.3 for the model of Qiu *et al.* (2019), and 16.2 for the CVMH model. The color version of this figure is available only in the electronic edition.

learning, which impacts the ability to interpret the results (Lakshminarayanan *et al.*, 2017). We obtain low bound uncertainty estimates of the results using an ensemble of perturbed test datasets created according to the associated uncertainties. An improved procedure may implement validation, where in each iteration forward calculation results based on the CNN-derived models are iteratively evaluated with respect to the real data. Additional future improvements include incorporating receiver function results (Bodin *et al.*, 2012; Shen *et al.*, 2013) and lateral discontinuities between geological units as constraints in the output models.

Data and Resources

The convolutional neural network (CNN) is implemented using the deep-learning framework of PyTorch (Paszke *et al.*, 2017, 2019). The training and prediction processes are performed on a laptop with a NVIDIA GeForce GTX 1060 core and a memory of 6 GB. For Application to Continental China section, scripts and training and test datasets are available at [https://github.com/jhsa26/SurfTomo CNN](https://github.com/jhsa26/SurfTomo_CNN) (last accessed January 2020). The real surface-wave dispersion measurements are from Shen *et al.* (2016) and Qiu *et al.* (2019). The V_S models of training dataset are from Shen *et al.* (2013) and Southern California Earthquake Center Community Velocity Model of Shaw *et al.* (2015; CVMH) of Southern California Earthquake Center (SCEC). Figures were prepared using Generic Mapping Tools (GMT; Wessel and Smith, 1998) and Matplotlib (Hunter, 2007). Supplemental material for this article includes tables comparing mean misfits for different Gaussian widths to convert dispersion curves and different layer thicknesses, and various figures showing V_S labels used for training, phase and group velocity maps of continental China and southern California, training loss curves, and misfit distributions.

Acknowledgments

This study was supported by National Natural Science Foundation of China (U1839205), National Key Research and Development Program of China (Grant 2018YFC1504102), China Scholarship Council (CSC), and the Department of Energy (Award DE-SC0016520). The article benefitted from constructive comments by two anonymous reviewers. The authors thank Weisen Shen for providing his surface-wave dispersion data and V_S models of central western United States and China. The authors thank Hailiang Xin for providing his Generic Mapping Tools (GMT) scripts to plot Figures 2 and 4.

References

- Ben-Zion, Y., F. L. Vernon, Y. Ozakin, D. Zogone, Z. E. Ross, H. Meng, M. White, J. Reyes, D. Hollis, and M. Barklage (2015). Basic data features and results from a spatially dense seismic array on the San Jacinto fault zone, *Geophys. J. Int.* **202**, no. 1, 370–380.
- Bodin, T., M. Sambridge, H. Tkalcic, P. Arroucau, K. Gallagher, and N. Rawlinson (2012). Transdimensional inversion of receiver functions and surface wave dispersion, *J. Geophys. Res.* **117**, no. B2, doi: [10.1029/2011JB008560](https://doi.org/10.1029/2011JB008560).
- Braeuer, B., and K. Bauer (2015). A new interpretation of seismic tomography in the southern Dead Sea basin using neural network clustering techniques, *Geophys. Res. Lett.* **42**, no. 22, 9772–9780.
- Brocher, T. M. (2005). Empirical relations between elastic wavespeeds and density in the Earth's crust, *Bull. Seismol. Soc. Am.* **95**, no. 6, 2081–2092.
- Cheng, X., Q. Liu, P. Li, and Y. Liu (2019). Inverting Rayleigh surface wave velocities for crustal thickness in eastern Tibet and the western Yangtze craton based on deep learning neural networks, *Nonlinear Process. Geophys.* **26**, no. 2, 61–71.
- Curtis, A., J. Trampert, R. Snieder, and B. Dost (1998). Eurasian fundamental mode surface wave phase velocities and their relationship with tectonic structures, *J. Geophys. Res.* **103**, no. B11, 26,919–26,947.
- Deng, J., W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei (2009). Imagenet: A large-scale hierarchical image database, *2009 IEEE Conf. on Computer Vision and Pattern Recognition*, 248–255.
- Devilee, R. J. R., A. Curtis, and K. Roy-Chowdhury (1999). An efficient, probabilistic neural network approach to solving inverse problems: Inverting surface wave velocities for Eurasian crustal thickness, *J. Geophys. Res.* **104**, no. B12, 28,841–28,857.
- Eymold, W. K., and T. H. Jordan (2019). Tectonic regionalization of the southern California crust from tomographic cluster analysis, *J. Geophys. Res.* **124**, no. 11, 11,840–11,865.

- Flament, N., S. Williams, R. D. Müller, M. Gurnis, and D. J. Bower (2017). Origin and evolution of the deep thermochemical structure beneath Eurasia, *Nat. Comm.* **8**, 14,164.
- Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*, MIT Press, Cambridge, Massachusetts.
- He, R., X. Shang, C. Yu, H. Zhang, and R. D. Van der Hilst (2014). A unified map of Moho depth and V_P/V_S ratio of continental China by receiver function analysis, *Geophys. J. Int.* **199**, no. 3, 1910–1918.
- Herrmann, R. B. (2013). Computer programs in seismology: An evolving tool for instruction and research, *Seismol. Res. Lett.* **84**, no. 6, 1081–1088.
- Hou, S., S. Angio, A. Clowes, I. Mikhalev, H. Hoeber, and S. Hagedorn (2019). Learn to invert: Surface wave inversion with deep neural network, *81st EAGE Conf. and Exhibition 2019 Workshop Programme*, London, United Kingdom, 3–6 June 2019.
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment, *Comput. Sci. Eng.* **9**, no. 3, 90.
- Kennett, B. L., E. R. Engdahl, and R. Buland (1995). Constraints on seismic velocities in the Earth from traveltimes, *Geophys. J. Int.* **122**, no. 1, 108–124.
- Keskar, N. S., D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang (2016). On large-batch training for deep learning: Generalization gap and sharp minima, available at <https://arxiv.org/abs/1609.04836> (last accessed January 2020).
- Kingma, D. P., and J. Ba (2014). Adam: A method for stochastic optimization, available at <https://arxiv.org/abs/1412.6980> (last accessed January 2020).
- Lakshminarayanan, B., A. Pritzel, and C. Blundell (2017). Simple and scalable predictive uncertainty estimation using deep ensembles, available at <https://arxiv.org/abs/1612.01474> (last accessed January 2020).
- LeCun, Y., and Y. Bengio (1995). Convolutional networks for images, speech, and time series, *Handb. Brain Theory Neural Netw.* **3361**, no. 10, 1995.
- LeCun, Y., Y. Bengio, and G. Hinton (2015). Deep learning, *Nature* **521**, no. 7553, 436.
- Lee, E. J., P. Chen, T. H. Jordan, P. B. Maechling, M. A. Denolle, and G. C. Beroza (2014). Full-3-D tomography for crustal structure in southern California based on the scattering-integral and the adjoint-wavefield methods, *J. Geophys. Res.* **119**, no. 8, 6421–6451.
- Lekic, V., and B. Romanowicz (2011). Tectonic regionalization without a priori information: A cluster analysis of upper mantle tomography, *Earth Planet. Sci. Lett.* **308**, nos. 1/2, 151–160.
- Lekic, V., S. Cottaar, A. Dziewonski, and B. Romanowicz (2012). Cluster analysis of global lower mantle tomography: A new class of structure and implications for chemical heterogeneity, *Earth Planet. Sci. Lett.* **357**, 68–77.
- Li, Z., M. A. Meier, E. Hauksson, Z. Zhan, and J. Andrews (2018). Machine learning seismic wave discrimination: Application to earthquake early warning, *Geophys. Res. Lett.* **45**, no. 10, 4773–4779.
- Lin, F. C., M. H. Ritzwoller, and R. Snieder (2009). Eikonal tomography: Surface wave tomography by phase front tracking across a regional broad-band seismic array, *Geophys. J. Int.* **177**, no. 3, 1091–1110.
- Lin, F. C., B. Schmandt, and V. C. Tsai (2012). Joint inversion of Rayleigh wave phase velocity and ellipticity using USArray: Constraining velocity and density structure in the upper crust, *Geophys. Res. Lett.* **39**, no. 12, doi: [10.1029/2012GL052196](https://doi.org/10.1029/2012GL052196).
- Liu, Y., H. Zhang, H. Fang, H. Yao, and J. Gao (2018). Ambient noise tomography of three-dimensional near-surface shear-wave velocity structure around the hydraulic fracturing site using surface microseismic monitoring array, *J. Appl. Geophys.* **159**, 209–217.
- Maas, A. L., A. Y. Hannun, and A. Y. Ng (2013). Rectifier nonlinearities improve neural network acoustic models, *Proc. ICML* **30**, no. 1, 3 pp.
- McBrearty, I. W., A. A. Delorey, and P. A. Johnson (2019). Pairwise association of seismic arrivals with convolutional neural networks, *Seismol. Res. Lett.* **90**, no. 2A, 503–509.
- Meier, U., A. Curtis, and J. Trampert (2007). Global crustal thickness from neural network inversion of surface wave data, *Geophys. J. Int.* **169**, no. 2, 706–722.
- Mosegaard, K., and A. Tarantola (1995). Monte Carlo sampling of solutions to inverse problems, *J. Geophys. Res.* **100**, no. B7, 12,431–12,447.
- Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga (2017). Automatic differentiation in PyTorch, *31st Conf. on Neural Information Processing Systems*, Long Beach, California, 4 pp.
- Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. (2019). PyTorch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems*, 8024–8035.
- Perol, T., M. Gharbi, and M. Denolle (2018). Convolutional neural network for earthquake detection and location, *Sci. Adv.* **4**, no. 2, e1700578.
- Qiu, H., F.-C. Lin, and Y. Ben-Zion (2019). Eikonal tomography of the Southern California plate boundary region, *J. Geophys. Res.* **124**, doi: [10.1029/2019JB017806](https://doi.org/10.1029/2019JB017806).
- Ross, Z. E., M. A. Meier, and E. Hauksson (2018). P wave arrival picking and first-motion polarity determination with deep learning, *J. Geophys. Res.* **123**, no. 6, 5120–5129.
- Ross, Z. E., Y. Yue, M. A. Meier, E. Hauksson, and T. H. Heaton (2019). PhaseLink: A deep learning approach to seismic phase association, *J. Geophys. Res.* **124**, no. 1, 856–869.
- Roux, P., L. Moreau, A. Lecointre, G. Hillers, M. Campillo, Y. Ben-Zion, D. Zigone, and F. Vernon (2016). A methodological approach towards high-resolution surface wave imaging of the San Jacinto Fault Zone using ambient-noise recordings at a spatially dense array, *Geophys. J. Int.* **206**, no. 2, 980–992.
- Sambridge, M. (1999a). Geophysical inversion with a neighbourhood algorithm—I. Searching a parameter space, *Geophys. J. Int.* **138**, no. 2, 479–494.
- Sambridge, M. (1999b). Geophysical inversion with a neighbourhood algorithm—II. Appraising the ensemble, *Geophys. J. Int.* **138**, no. 3, 727–746.
- Shapiro, N. M., M. Campillo, L. Stehly, and M. H. Ritzwoller (2005). High-resolution surface-wave tomography from ambient seismic noise, *Science* **307**, no. 5715, 1615–1618.
- Shaw, J. H., A. Plesch, C. Tape, M. P. Suess, T. H. Jordan, G. Ely, E. Hauksson, J. Tromp, T. Tanimoto, R. Graves, et al. (2015). Unified structural representation of the southern California crust and upper mantle, *Earth Planet. Sci. Lett.* **415**, 1–15.
- She, Y., H. Yao, Q. Zhai, F. Wang, and X. Tian (2018). Shallow crustal structure of the middle-lower Yangtze River region in eastern China from surface-wave tomography of a large volume airgun-shot experiment, *Seismol. Res. Lett.* **89**, no. 3, 1003–1013.

- Shen, W., M. H. Ritzwoller, D. Kang, Y. Kim, F. C. Lin, J. Ning, W. Wang, Y. Zheng, and L. Zhou (2016). A seismic reference model for the crust and uppermost mantle beneath China from surface wave dispersion, *Geophys. J. Int.* **206**, no. 2, 954–979.
- Shen, W., M. H. Ritzwoller, and V. Schulte-Pelkum (2013). A 3-D model of the crust and uppermost mantle beneath the Central and Western US by joint inversion of receiver functions and surface wave dispersion, *J. Geophys. Res.* **118**, no. 1, 262–276.
- Simonyan, K., and A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition, available at <https://arxiv.org/abs/1409.1556> (last accessed January 2020).
- Stankiewicz, J., K. Bauer, and T. Ryberg (2010). Lithology classification from seismic tomography: Additional constraints from surface waves, *J. Afr. Earth Sci.* **58**, no. 3, 547–552.
- Wang, J., Z. Xiao, C. Liu, D. Zhao, and Z. Yao (2019). Deep-learning for picking seismic arrival times, *J. Geophys. Res.* **124**, no. 7, 6612–6624.
- Wessel, P., and W. Smith (1998). New improved version of generic mapping tools released, *Eos Trans. AGU* **79**, no. 579.
- Xin, H., H. Zhang, M. Kang, R. He, L. Gao, and J. Gao (2019). High-resolution lithospheric velocity structure of continental China by double-difference seismic travel-time tomography, *Seismol. Res. Lett.* **90**, no. 1, 229–241.
- Xu, B., N. Wang, T. Chen, and M. Li (2015). Empirical evaluation of rectified activations in convolutional network, available at <https://arxiv.org/abs/1505.00853> (last accessed January 2020).
- Yang, Y., and M. H. Ritzwoller (2008). Teleseismic surface wave tomography in the western US using the Transportable Array component of USArray, *Geophys. Res. Lett.* **35**, no. 4, doi: [10.1029/2007GL032278](https://doi.org/10.1029/2007GL032278).
- Yu, Y., J. Lin, L. Zhang, G. Liu, J. Hu, Y. Tan, and H. Zhang (2018). Identification of seismic wave first arrivals from earthquake records via deep learning, *International Conf. on Knowledge Science, Engineering and Management*, Springer, Cham, Switzerland, 274–282.
- Zhou, Y., G. Nolet, F. A. Dahlen, and G. Laske (2006). Global upper-mantle structure from finite-frequency surface-wave tomography, *J. Geophys. Res.* **111**, no. B4, doi: [10.1029/2005JB003677](https://doi.org/10.1029/2005JB003677).
- Zhu, W., and G. C. Beroza (2018). PhaseNet: A deep-neural-network-based seismic arrival-time picking method, *Geophys. J. Int.* **216**, no. 1, 261–273.

Manuscript received 12 August 2019

Published online 19 February 2020