Semi-supervised Surface Wave Tomography with Wasserstein Cycle-consistent GAN: Method and Application on Southern California Plate Boundary Region

Ao Cai[1], Hongrui Qiu[1], and Fenglin Niu[1]

[1]Department of Earth, Environmental and Planetary Sciences, Rice University, Houston, TX, USA

Corresponding author: Ao Cai (ao.cai@rice.edu)

Key points:

- A machine learning based method is developed for 1-D Vs inversion to include observed dispersion data into the training process
- The Wasserstein Cycle-GAN algorithm is used to improve training stability and spatial continuity of the output 3-D Vs model
- The final Vs model shows smaller misfits, sharper images of major faults, and large-scale features consistent with the surface geology

Abstract

Machine learning algorithm is applied to shear wave velocity (Vs) inversion in surface wave tomography, where a set of 1-D Vs profiles and the corresponding synthetic dispersion curves are used in network training. Previous studies showed that performances of a trained network depend on the input training dataset with limited diversity and therefore lack generalizability. Here, we present an improved semi-supervised algorithm-based network that takes both model-generated and observed surface wave dispersion data in the training process. The algorithm is termed Wasserstein cycle-consistent generative adversarial networks (Wcycle-GAN). Different from conventional supervised approaches, the GAN architecture extracts feature from the observed surface wave dispersion data that can compensate the limited diversity of the training dataset generated synthetically. The cycle-consistency enforces the reconstruction ability of input data from predicted model using a separate data generating network, while Wasserstein metric provides improved training stability and enhanced spatial smoothness of the output Vs model. We demonstrate improvements by applying the Wcycle-GAN method to 4076 pairs of fundamental mode Rayleigh wave phase and group velocity dispersion curves obtained in Southern California. The final 3-D Vs model from the best trained network shows large-scale features that are consistent with the surface geology. Our Vs model has smaller data misfits, yields better spatial smoothing, and provides sharper images of structures near faults in the top 15 km, suggesting the proposed Wcycle-GAN algorithm has stronger training stability and generalization abilities compared to conventional machine learning methods.

1. Introduction

Machine learning, particularly deep learning (LeCun et al., 2015), has attracted great attentions in geophysical fields, both in active- and passive-source seismology, such as automated seismic image segmentation (e.g., Wu et al., 2019), acoustic impedance inversion (e.g., Das et al., 2019), seismic phase picking (e.g., Ross and Ben-Zion, 2014; Ross et al., 2018; Zhu and Beroza, 2019), and event detection (e.g., Mousavi et al., 2020). The supervised learning such as convolutional neural networks (CNN) based methods have been widely utilized in geophysical studies. The neural networks approach has been proven to be promising in surface wave studies, for instance, extraction of crustal thickness (Devilee et al., 1999; Meier et al.,

2007; Cheng et al., 2019) from surface wave data, and automatic surface wave travel time dispersion picking (e.g., Zhang et al., 2020).

The shear wave velocity (Vs) inversion problem in surface wave tomography, i.e., mapping from surface wave velocity dispersion curves to 1-D Vs depth profiles, is highly nonlinear and underdetermined (e.g., Qiu et al., 2019). Conventional methods, such as linearized inversion (e.g., Herrmann et al., 2013), near-neighbor algorithm (e.g., Wathelet, 2008), and nonlinear Bayesian Markov Chain Monte Carlo method (MCMC; e.g., Roy & Romanowicz, 2017; Shen et al., 2013), are able to provide reliable results in previous studies if an initial model with sufficient accuracy is available. Hu et al. (2020) applied CNN based Vs inversion to Rayleigh wave dispersion data in China and the southern California (SC) plate boundary regions. The results show the effectiveness of the CNN technique and demonstrate the quality of the training dataset can affect accuracy of the output Vs model. In this study, we develop a deep-learning-based method that has the potential to alleviate the dependency on the accuracy of the initial Vs model while preserving the speed of the inversion as demonstrated in Hu et al. (2020).

The workflow of CNN based Vs inversion is shown in Figure 1a. A labeled dataset is split into a training set and a validation set. The "labeled data" usually consists of a known Vs model and its corresponding theoretical dispersion curves (e.g., Hu et al., 2020), and provides learnable examples to supervise the training of networks. The neural network stops updating when the prediction accuracy of the validation set reaches its optimum. The trained network is then applied to the observed dispersion data, later referred to as "unlabeled data", to output the best fitting Vs model. Since only labeled dataset is used in the training process, quality of the Vs model generated from the CNN is dependent on the similarity of the initial model and the true structures (Hu et al., 2020).

Figure 1

In comparison, generative adversarial networks (GAN; Goodfellow et al., 2014) introduce an adversarial network (discriminator) that incorporates both the labeled and unlabeled datasets into the training process (i.e., semi-supervised; Figure 1b), in an effort to alleviate the strong labeled dataset dependency of the CNN. In addition, we introduce Cycle-consistent GAN (Cycle-GAN; Zhu et al., 2017; Yi et al., 2017), in which a data generative network that learns to reconstruct the input data from its predicted model is added. It enforces the model and data generative subnets to be self-consistent and penalizes the reconstruction misfit, consequently reducing the variance of both the forward and backward generative networks. Compared to CNN or GAN, Cycle-GAN

has been proven to generate predictions for seismic trace interpolation (e.g., Kaur and Fomel, 2019) and impedance inversion (e.g., Wang et al., 2019) with better accuracy under the same setup. To further improve training stability (Arjovsky and Bottou, 2017) of the GAN algorithm, we adopt the structure of WGAN-GP, i.e., using Wasserstein distance and adding a gradient penalty (GP; Gulrajani et al., 2017) in the adversarial loss function (Arjovsky et al., 2017). The state-of-the-art hybrid method (hereinafter, Wcycle-GAN) combines the structures of Cycle-GAN and WGAN-GP, and outperforms conventional machine learning algorithms in biomedical translation (McDermott et al., 2018) and seismic impedance inversion (Cai et al., 2020).

In this paper, we demonstrate the application of the Wcycle-GAN method to Vs inversion using dispersion data derived for the SC plate boundary region, one of the most well-studied areas in the world. To better evaluate the seismic hazard in SC, several tomographic velocity models were developed using different datasets with various resolutions. The two Community Velocity Models (CVM), CVM-H15.1 (Shaw et al., 2015) and CVM-S4.26 (Lee et al., 2014), derived via full waveform tomography were widely used as the initial model in previous surface wave tomography studies of this area (e.g., Barak et al., 2015; Berg et al., 2018; Qiu et al., 2019). We first demonstrate the preparation of the training dataset as the combination of labeled dataset generated using the CVM-H15.1 and unlabeled data as the Rayleigh wave velocity dispersion maps from Qiu et al. (2019) in section 2. The network architecture of the Wcycle-GAN designed for this specific dataset and training process are presented in section 3. We then input the unlabeled data to the best trained Wcycle-GAN and obtain the final 3-D Vs model as the output. The final 3-D Vs model and the corresponding data misfits are presented in section 4. Compared to the models generated from conventional CNN algorithm and linearized inversion (Qiu et al., 2019), our Vs model yields smaller data misfit and improved image of structures near major faults. It is important to note that this method is the first machine learning based Vs inversion study that incorporates unlabeled data in the training process, which has the potential to be applied to surface wave dispersion datasets collected at various scales and from regions where subsurface structures are poorly constrained from previous studies.

2. Data

2.1. Rayleigh Wave Phase and Group Velocities – Unlabeled Data

We use the isotropic phase and group velocity maps of fundamental mode Rayleigh waves from Qiu et al. (2019) as the unlabeled dataset, which is used in both the training process and generation of the final 3-D Vs model. Travel times of surface waves reconstructed from ambient noise cross correlations for a seismic network with 346 stations in SC (triangles in Figure 2) are first measured at each station pair over a period range of 2 to 20 s. Eikonal tomography is then applied to resolve isotropic phase and group velocity maps and corresponding uncertainties with a grid size of 0.05°×0.05° (grid lines in Figure 2) for periods between 2.5s and 16s. Details of the Rayleigh wave velocity dispersion maps can be found in Qiu et al. (2019).

In this study, we use velocity dispersions in the period range between 3 s and 16 s to construct the unlabeled data, as the velocity maps at 2.5 s are less robust (i.e., large uncertainties) and only cover a small part of the SC plate boundary region. Dispersion curve and its uncertainty at each grid cell are interpolated and discretized into 17 samples, an interval of 0.5 s from 3 to 6 s and 1 s from 6 to 16 s. Since the uncertainties are estimated from Eikonal tomography by analyzing velocity maps derived for different virtual sources (Section 4 of Qiu et al., 2019), uncertainty values less than 0.05 km/s are set to 0.05 km/s to account for errors from other sources (e.g., dispersion picking, accuracy of the trained network, etc.). Grid cells with a phase or group velocity dispersion curve that has less than 8 sample points are excluded. In total, the unlabeled data consists of 4076 pairs of Rayleigh wave phase (Figure S1) and group velocity (Figure S2) dispersion curves, and the corresponding uncertainties are utilized to calculate the data misfit distribution in section 4.

2.2. Community Velocity Model and Synthetic Dispersion Curves – Labeled Data

We take advantage of the CVM resolved from full waveform tomography in constructing the labeled dataset for training the network. The CVM-H15.1 (later referred to as "CVM-H") is preferred to CVM-S4.26 in the network training because of its inclusion of topography, smaller misfit to observed dispersion data, and model simplicity, as discussed in Qiu et al. (2019). 16480 1-D profiles of Vs, Vp, and density are extracted from the CVM-H with a grid spacing of 0.03º×0.03º for the region covered by the unlabeled data (Figure 2). These 1-D profiles are discretized into 98 layers with a thickness of 0.5 km from 0 km to 49 km (relative to the earth surface) and a half space below 49 km. The study area is confined to a longitude range from 120.2ºW to 114.9ºW and a latitude range from 32.6ºN to 36.0ºN. Each 1-D Vs profile (Figure

Figure 2

S3a) is labeled by the synthetic Rayleigh wave phase and group velocity dispersion curves (Figure S3b). The synthetic velocity dispersion curve is calculated using the Computer Programs in Seismology (CPS) software package (Herrmann, 2013), in which 1-D profiles of Vs, Vp, and density at the target location are inputted.

3. Methodology

In contrast to the conventional CNN, GAN incorporates a discriminative network that enables the use of unlabeled data. In CNN applications (Figure 3a), we train a model generative network (G_m) using labeled data only, by iteratively minimizing the point-wise misfit between the translated model (i.e., G_m predictions) and the real Vs model. The misfit is also known as the estimation loss ($\mathcal{L}_{est}$) and can be measured in cross-entropy or least-square format. GAN runs updates of generative and discriminative networks separately in a single iteration (Figure 3b left column). In the first step, the trainable parameters are fixed in G_m and model discriminator (D_m) is updated. The discriminator is renewed to separate the real Vs models and the outputs from model generator. Numerically, this is implemented by forcing D_m to output binary discrimination, where "1" stands for real model samples in the labeled dataset and "0" represents the outputs from G_m. Next, the model discriminator is fixed, and the generator is updated to "fake" the discriminator and score "1" with the translated model. Similar process is conducted for unlabeled data (Figure 3b right column) except when we do not have real Vs models to feed into model generator in the first step. In this way, the discriminative network searches for a transformation to maximize the difference between real and translated models while the generator seeks to minimize it. The corresponding loss function is named adversarial loss ($\mathcal{L}_{adv}$), which can be calculated in cross-entropy (Goodfellow et al., 2014), least-square (Mao et al., 2017) and Wasserstein distance (Arjovsky et al., 2017).

Figure 3

Cycle-GAN (Figure 3c) further extends the GAN algorithm with the concept of "cycle-consistency", by introducing an extra data generative (G_d) and discriminative network (D_d). For simplicity, we separate the algorithm into data cycle (green arrows in Figure 3c) and model cycle (purple arrows in Figure 3c). In the data cycle for the labeled data, besides computation of the adversarial loss, the translated model (i.e., output from G_m) is fed into G_d to reconstruct the original dispersion data (i.e., the input to G_m). The point-wise reconstruction misfit (cycle-consistent loss, $\mathcal{L}_{cyc}$) is minimized during the iterations. Similar to the linearized Vs inversion

where we compute the predicted data from the current best model using known physical relations, in the Cycle-GAN, we compute the reconstructed data but replace the physical modeling with a data generative network. In the model cycle (bottom left of Figure 3c), we generate the translated data from the real Vs model and estimate the adversarial loss using the data discriminator D_d. The translated data is then fed into G_m to generate reconstructed model, and the cycle-consistent loss of the model reconstruction is penalized (Figure 3c left column). The unlabeled data go through similar process in the data cycle (Figure 3c right column). However, since their corresponding Vs models are unknown, there is no model cycle for the unlabeled data.

Our approach to resolve Vs structures from Rayleigh wave velocity dispersion curves is based on a specific Cycle-GAN algorithm that utilizes Wasserstein adversarial loss. We present the details of Wcycle-GAN algorithm-based surface wave tomography as follows.

3.1. Sub Neural Network Structures

The architecture of the proposed Wcycle-GAN consists of four sub neural networks (Figures S3c-S3f) – two generative subnets (G_m and G_d) and two discriminative subnets (D_m and D_d). Different from Hu et al. (2020), for all the subnets, a 1-D rather than 2-D neural network is implemented for network simplicity. Unlike image translation (Isola et al., 2017) or seismic impedance inversion (Cai et al., 2020) problems, surface wave dispersion curves and Vs models have different ranges of values and dimensions. In this study, the input dimension of dispersion data to the neural networks is 17x2 with the phase and group velocities as two separate channels, while the Vs model is 99x1 (Section 2). Considering the difference in Vs model and dispersion data dimensions, we design specific architectures for model and data generative subnets (Figures S3d and S3e). In the model generator, we double the number of filters at each convolutional layer similar to the VGG16 network (Simonyan and Zisserman, 2014). The number of filters at each convolutional layer from shallow to deep is 32, 64, 128, and 256 (Figures S3c-S3f), respectively. Accordingly, in the data generative subnet (G_d), we first upsample the Vs model to the dense feature map with a dimension of 17x256, and sequentially half the number of filters in the following convolutional layers. For both the model and data discriminative subnets (Figures S3c and S3f), we double the number of filters in the convolutional layers and apply a sigmoid activation function in the fully connected layer to output probability values between 0 and 1.

In all the subnets, the convolutional layer uses 1D convolution with kernel size 3x1 and zero padding on the boundary. The stride equals to 1 except in the D_m, where the stride value of 2 is used to reduce trainable parameters. To accelerate the training process, at each convolutional layer, we apply the batch normalization (Ioffe and Szegedy, 2015) after the ReLU (Nair and Hinton, 2010) activation and initialize the weight parameters in the convolutional layers using the He initialization (He et al., 2015). In addition, as suggested by Gulrajani et al. (2017), we replace the batch normalization in the adversarial subnets with the layer normalization (Ba et al., 2016).

3.2. Loss Function

To optimize both the generative and adversarial subnets, the loss function in the Wcycle-GAN is calculated by a combination of the estimation loss, cycle-consistent loss, and adversarial loss, and can be written as

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_1 \mathcal{L}_{cyc} + \lambda_2 \mathcal{L}_{est}, \quad (1)$$

where $\mathcal{L}_{adv}$, $\mathcal{L}_{cyc}$, and $\mathcal{L}_{est}$ stand for the Wasserstein adversarial loss, the cycle-consistent loss, and the estimation loss, respectively. The hyperparameters λ_1 and λ_2 are the weighting factors. We introduce the notations which will be used in the following discussions: $\boldsymbol{m}$ and $\boldsymbol{d}$ stand for the labeled Vs model and synthetic dispersion data pairs, respectively; $\boldsymbol{d}^*$ is the unlabeled real dispersion data; $\boldsymbol{W}_*$ represents the trainable parameters in the networks; $f_{\boldsymbol{W}_*}(*)$ is the neural network operator that generates translated samples using Vs model as input. For instance, $\boldsymbol{W}_{G_m}$ is the trainable parameters in the model generative subnet; $f_{\boldsymbol{W}_{G_m}}(\boldsymbol{m})$ is the output translated dispersion data generated by G_m.

The calculation of Wasserstein adversarial loss can be described as two steps. First, we fix the trainable parameters in the generator G_m and update discriminator D_m using the formula

$$\min_{\boldsymbol{W}_{D_m}} \mathcal{L}_{adv} = -f_{\boldsymbol{W}_{D_m}}(\boldsymbol{d}) - f_{\boldsymbol{W}_{D_m}}(\boldsymbol{d}^*) + f_{\boldsymbol{W}_{D_m}}\left(f_{\boldsymbol{W}_{G_m}}(\boldsymbol{m})\right) + \lambda \mathcal{L}_{gp} \quad (2)$$

The gradient penalty loss $\mathcal{L}_{gp}$ enforces the discriminator to be 1-Lipschitz continuous, which is the assumed to optimize the Wasserstein GAN (Arjovsky et al., 2017). Detailed implementation of $\mathcal{L}_{gp}$ can be found in Gulrajani et al. (2017). In practice, the weighting factor λ should be large enough to avoid exploding gradient (Gulrajani et al., 2017). In this study, we set $\lambda = 100$ to

ensure good numerical stabilities (Cai et al., 2020). In the second step, the D_m is fixed and G_m is updated via

$$\min_{\boldsymbol{W}_{G_m}} \mathcal{L}_{adv} = f_{\boldsymbol{W}_{D_m}}(\boldsymbol{d}) + f_{\boldsymbol{W}_{D_m}}(\boldsymbol{d}^*) - f_{\boldsymbol{W}_{D_m}}\left(f_{\boldsymbol{W}_{G_m}}(\boldsymbol{m})\right). \tag{3}$$

Note that the computation of Wasserstein adversarial loss is slightly different from that of the conventional adversarial loss. Corresponding mathematical derivations of Wasserstein adversarial loss can be found in Arjovsky et al. (2017).

The cycle consistency loss (Zhu et al., 2017) measures the reconstruction errors with the expression

$$\mathcal{L}_{cyc}(\boldsymbol{W}_{G_m}, \boldsymbol{W}_{G_d}) = E\left(\boldsymbol{d}^*, f_{\boldsymbol{W}_{G_d}}(f_{\boldsymbol{W}_{G_m}}(\boldsymbol{d}^*))\right) + E\left(\boldsymbol{d}, f_{\boldsymbol{W}_{G_d}}(f_{\boldsymbol{W}_{G_m}}(\boldsymbol{d}))\right), \tag{4}$$

for the data cycle and

$$\mathcal{L}_{cyc}(\boldsymbol{W}_{G_m}, \boldsymbol{W}_{G_d}) = E\left(\boldsymbol{m}, f_{\boldsymbol{W}_{G_m}}(f_{\boldsymbol{W}_{G_d}}(\boldsymbol{m}))\right), \tag{5}$$

for the model cycle. $E(*,*)$ stands for a measurement of the difference between two samples, and in this proposed method it is computed by mean-square error (MSE). Using the labeled data as an example, the $\mathcal{L}_{cyc}$ is computed as the difference between the input data $\boldsymbol{d}$ and the reconstructed data $f_{\boldsymbol{W}_{G_d}}(f_{\boldsymbol{W}_{G_m}}(\boldsymbol{d}))$. The reconstructed data is the output after the original data consequently passed through the model (G_m) and data (G_d) generative subnets. We also penalize the estimation loss in the Wcycle-GAN algorithm to constrain the fitting in the labeled dataset, by computing the MSE between the translated samples and ground truth in the model and data domain,

$$\mathcal{L}_{est}(\boldsymbol{W}_{G_m}, \boldsymbol{W}_{G_d}) = E\left(\boldsymbol{m}, f_{\boldsymbol{W}_{G_m}}(\boldsymbol{d})\right) + E\left(\boldsymbol{d}, f_{\boldsymbol{W}_{G_d}}(\boldsymbol{m})\right). \tag{6}$$

The complete loss functions can be found in the supplementary materials.

3.3. Training Neural Networks and Evaluation

Before feeding the dispersion data and Vs model into the neural network, we apply linear transformations (see supplementary materials) to normalize them into the interval range of [-1, 1] to speed up the convergence of the training process. Outputs of the neural network in the data and model domains are transformed back to its original amplitude according to their linear transformation relations before computing misfits. For a comparative study, we apply both the

conventional 1-D CNN and the proposed Wcycle-GAN method to the Vs inversion at SC region, and the structure of CNN is the same as the model generative subnet (G_m) in the Wcycle-GAN. For the training process of both CNN and Wcycle-GAN (Figure S4), the iteration stops when the root-mean-square (RMS) misfit between the predicted and true shear velocities in the labeled data is below 0.07 km/s,

$$E_{RMS} = \sqrt{\frac{1}{N_{batch}} \sum_{i=1}^{N_{batch}} \left\| V_{s\_i}^{pred} - V_{s\_i}^{label} \right\|_2^2} \quad , \tag{7}$$

where N_{batch} is the number of Vs models in a batch, $V_{s\_i}^{pred}$ and $V_{s\_i}^{label}$ are the predicted Vs and true models in the labeled data, respectively. For the hyperparameter selection, we choose $\lambda_1 = 5$ and $\lambda_2 = 3$ for training the Wcycle-GAN. The training batch size is 160 for the labeled data and 80 for the unlabeled data. We use Adam (Kingma and Ba, 2014) for optimization with a learning rate of 5×10^{-5} and other parameters as default. For the CNN training, the neural networks could further lower its RMS misfit of the labeled data at later epochs, which may result in overfitting.

Finally, we apply the trained generative networks (G_m) to the observed dispersion data and output the final Vs model. To evaluate the performance of models obtained from different methods, we compute the chi-square misfit between the predict data calculated using the final Vs model and the observed dispersion data at every grid point:

$$\chi = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left[\frac{d_i^{pred} - d_i^{obs}}{\sigma_i^{obs}} \right]^2}, \tag{8}$$

where N=17x2 is the number of observed dispersion data points, d_i^{pred} and d_i^{obs} are the theoretical and observed dispersion wave speed (i.e., phase and group velocities) at the i^{th} data point, and σ_i^{obs} is the corresponding data uncertainty. A good data fitting is achieved when the normalized χ^2 misfit is close to 1 (Bevington, 1969; Zelt et al., 2003).

4. Results

The advantages of the proposed Wcycle-GAN method are demonstrated using surface wave dispersion data obtained from the SC plate boundary region. We first present the 3-D Vs model obtained from Wcycle-GAN method and compare it with that of Qiu et al. (2019) and the surface geology (section 4.1). Then, models derived from different machine learning algorithms (e.g.,

CNN) are compared to illustrate the advantages of incorporating unlabeled data into the network training process (section 4.2).

4.1. Output 3-D Vs Model

For training the Wcycle-GAN, the results converge after 1700 epochs. The trained network is applied to the observed dispersion data and generate the final 3-D Vs model by assembling all the 1-D Vs predictions. Because of the limited period range (i.e., 3-16s) of the input Rayleigh wave dispersion curves, the Vs model resolved beyond the 3-20 km depth range are not well constrained (Qiu et al., 2019). Therefore, we only focus on the Vs models at depths of 3-15 km. Depth slices at the depth of 5 km and 10 km for the initial model (CVM-H) and differences between the initial and final models are presented in Figure S5. The largest differences between our final model and the CVM-H are found underneath the basins and near the Salton Trough in the top 3-10 km, consistent with that in Qiu et al. (2019).

Figure 4 shows the depth slices of the Vs model resolved at 5 km and 10 km from various methods (Figure S6 for depth slices at 3 km and 15 km). At shallow depths (e.g., in the top 3-7 km; Figures 4c and S6a-b), we can clearly see a good agreement between our final model (Figures 4c and 4g) and the surface geology, such as low velocity anomalies at Southern Central valley, LA Basin, Ventura Basin, and the Salton Trough; areas with high velocity in the Peninsular Ranges (e.g., Berg et al., 2018; Lee et al., 2014; Tape et al., 2010). It is important to note that our model shows the low velocity zone better within the junction between the San Jacinto Fault (SJF) and San Andreas Fault (SAF) compared to the CVM-H (Figure S5a-b).

Figure 4

At greater depths (e.g., below 10 km; Figures 4g and S6c-d), a sharp velocity contrast from west to east in the Peninsular Ranges is observed, which is related to the Hemet stepover (Marliyani et al., 2013). Clearer velocity contrasts across major fault systems, such as Elsinore Fault (EF), SJF and SAF are depicted in the map views of the final Vs model (Figures 4g and S6c-d), suggesting the derived Vs model yields higher resolutions compared to the CVM-H. These observations agree well with the large-scale features found in the Vs model of Qiu et al. (2019). In addition, the differences between the two models at different depth slices, which are shown in Figure S7, are rather small. The consistent observation of largest velocity updates beneath basin, coherent large-scale velocity structures, together with small model differences suggest a cross-validation of both the Wcycle-GAN and the Eikonal tomography model.

Unlike the conventional linearized Vs inversion (e.g., Qiu et al., 2019), in which an extra spatial filtering is applied to achieve a smoothed 3-D Vs model, our final Vs model in map view suggests that the Wcycle-GAN method inherently guarantees a spatial smoothness that is similar to those of the surface wave velocity dispersion maps (Figure S1). The proposed Wcycle-GAN method shows potential to improve lateral consistency of the neighboring 1-D models, which is a significant drawback in current dispersion-curve based 1-D Vs inversion. We note that, while presenting the Vs model in map view better shows the large-scale features that are consistent with the surface geology, it is hard to demonstrate variations in structures at depth, such as geometry of the major fault systems (e.g., width of low-velocity zone and dipping fault). Thus, in section 5, we further illustrate three depth cross sections (blue lines in Figure 2) of our final Vs model for a detailed discussion of the resolved fault structures.

Figure 5 shows histograms of the chi-square misfit of the dispersion data computed following equation 8 for Vs models obtained from different methods. To calculate the misfit, the compressional velocity (Vp) model by assuming the same Vp/Vs ratio as the CVM-H and the density model same as the CVM-H are used. Map views of χ misfits are depicted in Figure S8. The misfits are lower using the Wcycle-GAN model than using the Vs model of Qiu et al. (2019) in the Salton Trough region, suggesting our final Vs model is more reasonable in the area. The average misfit of the Wcycle-GAN based model (0.949; Figure 5c) is slightly smaller than 1, suggesting the final Vs model is of good fit to the observed dispersion data. Although the average misfit value of our model is a bit higher than that (0.864; Figure 5d) of Qiu et al. (2019), we note the misfit values are also sensitive to the input Vp and density models, which might not be accurate as we assume the Vp/Vs ratio and density to the same as those of CVM-H.

Figure 5

4.2. Comparison with the Conventional CNN Algorithm

In this section, we compare the Vs model from the Wcycle-GAN method with that derived from the conventional CNN algorithm. The training parameters (e.g., batch size, learning rate) and stopping criteria are the same as illustrated in section 3.3. For the CNN case, 120 epochs are needed to achieve a convergent training. Training the Wcycle-GAN takes longer time than the CNN method due to extra efforts on training the adversarial networks. But the Wcycle-GAN method still provides sufficient efficiency as the 1700 epochs only took ~12 hours using a single NVIDIA GeForce RTX 2080 graphic card. After the training process, for both the CNN and

Wcycle-GAN, it only takes ~30 s to generate the 3-D Vs model using 4076 pairs of group and phase velocity dispersion curves, demonstrating their efficiencies in model predictions.

Figures 4a and 4e present depth slices of the Vs model derived from the CNN method at 5 km and 10 km, respectively, while the data misfit histogram is shown in Figure 5a. Compared with results from the proposed Wcycle-GAN method (Figures 4c, 4g, and 5c), the Vs model from CNN is less smooth and continuous, and shows much higher average misfit values, suggesting results from the CNN method are less stable and robust. This is likely due to the limited diversity provided in the labeled dataset generated synthetically. In addition, the Wasserstein metric used in the Wcycle-GAN improves the long-wavelength features recovery in the network training, resulting in an enhanced spatial smoothness of the output 3-D Vs model. Similar property of Wasserstein metric has been observed in near surface seismic velocity estimation using full-waveform inversion (Yang et al., 2018). The better accuracy in fitting the observed dispersion data and spatial continuity of the Vs model from the Wcycle-GAN method demonstrates the effectiveness of the proposed method by incorporating advanced loss function, cycle consistency, and unlabeled data into the training process.

5. Discussions

We suggest the proposed Wasserstein Cycle-GAN to be a robust data-driven method. On one hand, the Wasserstein adversarial loss with gradient penalty provides good training stability and convergence characteristic comparing with cross-entropy or least-squares. Figure S9 shows the comparative study of using different metrics for adversarial loss. Using the least square loss may result in underfitting to the labeled data as the incorrect prediction of the velocity jump at Moho depth. Both cross-entropy and least-square adversarial loss can result in strong artifacts and negative velocity gradient in the Vs predictions using unlabeled data. In comparison, Wasserstein loss results in high model prediction quality using either labeled or unlabeled data. The Vs model from the Wcycle-GAN method is smoother and laterally more continuous, compared to models derived from supervised method (Section 4.2). On the other hand, the proposed method incorporates the observed dispersion data into the training process that improves the generalization ability of the trained network. In addition, for the weighting factors in the loss function, changes in hyperparameter λ_1 and λ_2 has relatively small effects on the final derived

Vs model, but a future study of the effects of the two hyperparameters would be beneficial for optimizing the Wcycle-GAN method.

To further discuss the importance of incorporating unlabeled data in the training process, we perform a third experiment, in which the same Wcycle-GAN structure is used but trained without the unlabeled data. The weighting factor λ_2 of the loss function (equation 1) is set to 10, different from section 3.2, since only the labeled data is used for training. Figures 4b and 4f present map views of the output Vs model from such experiment. Strong local velocity jumps and artificial lateral heterogeneities are seen in the model, comparing with the Vs model map views in Figures 4c and 4g. Training the Wcycle-GAN without unlabeled data results in larger data misfits (~2.3 in average) that are shown clearly both in histogram (Figure 5b) and map view (Figure S8), compared to those of the proposed Wcycle-GAN method (0.949). Therefore, incorporating the unlabeled data into the training process is essential for providing robust and reliable Vs model when using machine learning based methods to solve the Vs inversion problem.

We also note that our Wcycle-GAN method requires less amount of labeled data in the training. To demonstrate this, we reduce the amount of labeled data by down sampling with a grid spacing of 0.1ºx0.1º (originally 0.03ºx0.03º). This results in a selection of 1890 out of the originally 16480 labeled data, which is even much less than the number (4076) of observed dispersion curves. Figures 6a and 6c show the depth slices of the Vs model from the Wcycle-GAN method trained with down sampled labeled data. The resulting Vs models are similar between the methods trained using a reduced and the full labeled datasets. Figure 5e shows the data misfit of the Vs model from the network trained with reduced labeled dataset. There is only a small increase in the mean misfit, i.e., from 0.949 to 1.10, compared to that of results trained with the full labeled dataset. It is important to note that the average misfit value 1.1 is still much smaller than those of the supervised methods (Figures 5a and 5b). The result suggests the redundancy in the labeled data and further demonstrates the strength of the proposed Wcycle-GAN method in resolving high accuracy Vs model using small amount of labeled data. This can also save time spent on training as it takes only ~4 hours after reducing the amount of the labeled dataset by almost a factor of 10.

Figure 6

An extension to the proposed Wcycle-GAN algorithm is incorporating the location (i.e., longitude and latitude) as prior information in the training process, which can further enhance the accuracy in the application of Vs inversion. Map views of the Vs model, derived from the

proposed method after incorporating the latitude and longitude of both the labeled and unlabeled data in the training process, at 5 km and 10 km are presented in Figures 6b and 6d, respectively. Details of how to incorporate location information into a machine learning network training can be found in supplementary materials. The Vs models resolved from networks trained with and without the input of location information are nearly identical to each other at a large scale (e.g., tens of kilometers; Figures 4c, 4f, 6b, and 6d). The data misfits (~0.9 in Figure 5f) are slightly smaller after incorporating the location information into the training process. Therefore, we show the cross sections of the Vs model resolved from the network trained with location information incorporated in Figure 7 to infer structures of the major fault systems. We note that the incorporation of location information for both the labeled and unlabeled data will have greater impact on the results when applying to the Vs inversion at regional or global scales.

We show the cross sections DD', EE' and FF' (blue lines in Figure 2), the same as those shown in Figure 1 of Qiu et al. (2019), of the final Vs model between 3 km and 20 km to infer the structures of EF, SJF, and SAF at depth. In the profile DD', the low velocity zone indicates both the SJF and SAF are nearly vertical. This is consistent with the fault geometry near San Gorgonio Pass (SGP) from the Community fault model in SC (CFMv5; Plesch et al., 2007). Besides, we observe a pronounced low-velocity body (dashed circle, Figure 7) between depths of 15-20 km, which is consistent with the results of Qiu et al. (2019) (Figure S10c). This low velocity anomaly at great depth, with ~5-7% lower velocities compared to the surrounding media, is likely related to the large damage volume beneath the SGP estimated in Ben-Zion and Zaliapin (2019).

Figure 7

In profile EE', we observe a broad (~5-km-wide) flower-shaped (i.e., width decreases with depth) fault damage zone with ~2-3% average velocity reduction for the SAF in the top 8-10 km that is clearly dipping towards the northeast. The estimated dipping angle of SAF in profile EE' is ~60º. This dipping angle is consistent with the observation in Qiu et al. (2019), but the flower-shaped fault damage zone is less clear in their results (Figure S10g). Besides, the low velocity anomaly beneath the Eastern California Shear Zone (ECSZ) is slightly deeper than that in Qiu et al. (2019). Similarly, the SAF is highlighted by a flower-shaped low-velocity zone that is dipping towards the northeast with a similar angle (~60º) in the top 10 km. Different from EE', the low velocity zone is more pronounced (~4-5%) in FF', likely indicating the rocks inside the fault zone are more damaged in the southwest.

The flower-shaped fault zone structures in EE' and FF' are consistent with the model of Fuis et al. (2016) derived for the southern section of the SAF by jointly inverting gravity and magnetic data. In addition, the observed ~60º dipping angle in both EE' and FF' agrees well with the previous the estimation from magnetic data (~65º; Fuis et al., 2012). It is important to note that the model of Qiu et al. (2019) is subject to the choice of damping parameter in and spatial smoothing after the Vs inversion. Therefore, through the cross section comparisons, we again demonstrate the robustness of our Vs model from the Wcycle-GAN model and confirm with a different method that the flower-shaped damage zone and fault dipping towards northeast observed for the southern section of the SAF in Qiu et al. (2019) are reliable. These features have important implications, such as a better understanding of strong ground motions produced by earthquakes that will occur on the SAF.

6. Conclusions

We implement the Wcycle-GAN method to the Vs inversion in surface wave tomography, by incorporating unlabeled data into the network training process. The proposed method shows an improved prediction accuracy, better training stability, and only requires a small amount of labeled data, compared to CNN-based method. We demonstrate these improvements by using the fundamental mode Rayleigh wave velocity dispersion data derived in the Southern California plate boundary region. The final Vs model obtained from the proposed method show clearer images of structures near faults in the top 15 km, specifically the low velocity damage zone centered on the southern section of the San Andreas fault that is dipping ~60º to the northeast. In addition, integrating longitude and latitude information into the Wcycle-GAN algorithm further improves the prediction accuracy as well as the spatial continuity of the final Vs model, particularly in the cross sections. For future studies, we would like to investigate the potential of this method by reducing the amount of labeled data through leveraging random sampling or sampling strategy based on clustering analysis (Eymold & Jordan, 2019).

Acknowledgements

The Rayleigh wave velocity dispersion data used in this study are derived in Qiu et al. (2019). The labeled dataset is extracted from the Southern California Earthquake Center (SCEC) Community Velocity Model of Shaw et al. (2015; CVMH). The Wcycle-GAN is implemented

using the deep-learning framework of TensorFlow. The training and prediction processes are conducted using a single NVIDIA GeForce RTX 2080 GPU with a memory of 8GB. Fruitful discussions with Dr. Jing Hu at University of Science and Technology of China (USTC) are well appreciated. We are grateful to the Department of Earth, Environmental and Planetary Sciences, Rice University, for supporting this study and A. Cai's PhD research.

References

Arjovsky, M., & Bottou, L. (2017). Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein gan. *arXiv preprint arXiv:1701.07875*.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Barak, S., Klemperer, S. L., & Lawrence, J. F. (2015). San A ndreas F ault dip, P eninsular R anges mafic lower crust and partial melt in the S alton T rough, S outhern C alifornia, from ambient-noise tomography. *Geochemistry, Geophysics, Geosystems*, *16*(11), 3946-3972.

Barmin, M. P., Ritzwoller, M. H., & Levshin, A. L. (2001). A fast and reliable method for surface wave tomography. *Pure and Applied Geophysics*, *158*(8), 1351–1375. https://doi.org/10.1007/PL00001225

Ben-Zion, Y., & Zaliapin, I. (2019). Spatial variations of rock damage production by earthquakes in southern California. *Earth and Planetary Science Letters*, *512*, 184-193.

Berg, E. M., Lin, F. C., Allam, A., Qiu, H., Shen, W., & Ben-Zion, Y. (2018). Tomography of Southern California via Bayesian joint inversion of Rayleigh wave ellipticity and phase velocity from ambient noise cross-correlations. *Journal of Geophysical Research: Solid Earth*, *123*, 9933–9949. https://doi.org/10.1029/2018JB016269

Bevington, P. R., (1969), Data reduction and error analysis for the physical sciences: McGraw-Hill.

Brocher, T. M., (2005), Empirical relations between elastic wavespeeds and density in the Earth's crust: Bulletin of the seismological Society of America, 95(6), 2081-2092.

Cai, A., Di, H., Li, Z., Maniar, H., Abubakar, A. (2020). Wasserstein cycle-consistent generative adversarial network for improved seismic impedance inversion: Example on 3D SEAM model. In *SEG Technical Program Expanded Abstracts 2020* (pp. 1274-1278). Society of Exploration Geophysicists.

Cheng, X., Liu, Q., Li, P., & Liu, Y. (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.*, (2), 61-71.

Das, V., Pollack, A., Wollner, U., & Mukerji, T. (2019). Convolutional neural network for

seismic impedance inversion. *Geophysics*, *84*(6), R869-R880.

Devilee, R. J. R., Curtis, A., & Roy-Chowdhury, K. (1999). An efficient, probabilistic neural network approach to solving inverse problems: Inverting surface wave velocities for Eurasian crustal thickness. *Journal of Geophysical Research: Solid Earth*, *104*(B12), 28841-28857.

Eymold, W. K., & Jordan, T. H. (2019). Tectonic regionalization of the southern California crust from tomographic cluster analysis. *Journal of Geophysical Research: Solid Earth*, *124*(11), 11840-11865.

Fang, H., Zhang, H., Yao, H., Allam, A., Zigone, D., Ben-Zion, Y., et al. (2016). A new algorithm for three-dimensional joint inversion of body wave and surface wave data and its application to the Southern California plate boundary region. *Journal of Geophysical Research: Solid Earth*, *121*, 3557–3569. https://doi.org/10.1002/2015JB012702

Fuis, G. S., Scheirer, D. S., Langenheim, V. E., & Kohler, M. D. (2012). A new perspective on the geometry of the San Andreas fault in southern California and its relationship to lithospheric structure. *Bulletin of the Seismological Society of America*, *102*(1), 236-251.

Fuis, G. S., Bauer, K., Goldman, M. R., Ryberg, T., Langenheim, V. E., Scheirer, D. S., ... & Graves, R. W. (2017). Subsurface geometry of the San Andreas fault in southern California: Results from the Salton Seismic Imaging Project (SSIP) and strong ground motion expectations. *Bulletin of the Seismological Society of America*, *107*(4), 1642-1662.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., (2014), Generative adversarial nets: Advances in Neural Information Processing Systems, 2672-2680.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville A., (2017), Improved training of Wasserstein GANs: Advances in Neural Information Processing Systems, 5767-5777.

He, K., Zhang, X., Ren, S. and Sun, J., (2015), Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification: Proceedings of the IEEE International Conference on Computer Vision, 1026-1034.

Herrmann, R. B. (2013). Computer programs in seismology: An evolving tool for instruction and research. *Seismological Research Letters*, *84*(6), 1081-1088.

Hu, J., Qiu, H., Zhang, H., & Ben-Zion, Y., (2020), Using Deep Learning to Derive Shear-Wave Velocity Models from Surface-Wave Dispersion Data, Seismological Research Letters (2020) 91 (3): 1738–1751. https://doi.org/10.1785/0220190222

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1125-1134).

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444.

Lee, E.-J., Chen, P., Jordan, T. H., Maechling, P. B., Denolle, M. A. M., & Beroza, G. C. (2014). Full-3-D tomography for crustal structure in Southern California based on the scattering-integral and the adjoint-wavefield methods. *Journal of Geophysical Research: Solid Earth, 119*, 6421–6451. https://doi.org/10.1002/2014JB011346

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., & Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2794-2802).

Marliyani, G. I., Rockwell, T. K., Onderdonk, N. W., & McGill, S. F. (2013). Straightening of the Northern San Jacinto Fault, California, as Seen in the Fault-Structure Evolution of the San Jacinto Valley StepoverStraightening of Northern SJF as Seen in Fault-Structure Evolution of San Jacinto Valley Stepover. *Bulletin of the Seismological Society of America*, *103*(3), 2047-2061.

McDermott, M. B., Yan, T., Naumann, T., Hunt, N., Suresh, H., Szolovits, P., & Ghassemi, M. (2018). Semi-supervised biomedical translation with cycle wasserstein regression GANs. In Thirty-Second AAAI Conference on Artificial Intelligence.

Meier, U., Curtis, A., & Trampert, J. (2007). Global crustal thickness from neural network inversion of surface wave data. *Geophysical Journal International*, *169*(2), 706-722.

Mousavi, S. M., Ellsworth, W. L., Zhu, W., Chuang, L. Y., & Beroza, G. C. (2020). Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking. *Nature Communications*, *11*(1), 1-12.

Nair, V., and Hinton, G., E., (2010), Rectified linear units improve restricted boltzmann machines. In ICML, 2010.

Kaur, H., Pham, N., & Fomel, S. (2019). Seismic data interpolation using CycleGAN. In *SEG Technical Program Expanded Abstracts 2019* (pp. 2202-2206). Society of Exploration Geophysicists.

Plesch, A., Shaw, J. H., Benson, C., Bryant, W. A., Carena, S., Cooke, M., ... & Hauksson, E. (2007). Community fault model (CFM) for southern California. *Bulletin of the Seismological Society of America*, *97*(6), 1793-1802.

Qiu, H., Lin, F. C., & Ben-Zion, Y. (2019). Eikonal tomography of the Southern California plate boundary region. *Journal of Geophysical Research: Solid Earth*, *124*(9), 9755-9779.

Ross, Z. E., & Ben-Zion, Y. (2014). Automatic picking of direct P, S seismic phases and fault zone head waves. *Geophysical Journal International*, *199*(1), 368-381.

Ross, Z. E., Meier, M. A., & Hauksson, E. (2018). P wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth, 123*(6), 5120-5129.

Roy, C., & Romanowicz, B. A. (2017). On the implications of a priori constraints in transdimensional Bayesian inversion for continental lithospheric layering. *Journal of Geophysical Research: Solid Earth*, *122*(12), 10-118.

Share, P. E., Ben-Zion, Y., Ross, Z. E., Qiu, H., & Vernon, F. L. (2017). Internal structure of the

San Jacinto fault zone at Blackburn Saddle from seismic data of a linear array. *Geophysical Journal International*, *210*(2), 819–832. https://doi.org/10.1093/gji/ggx191

Shaw, J. H., Plesch, A., Tape, C., Suess, M. P., Jordan, T. H., Ely, G., Hauksson, E., Tromp, J., Tanimoto, T., Graves, R., et al. (2015). Unified structural representation of the southern California crust and upper mantle, Earth Planet. Sci. Lett. 415, 1–15.

Shen, W., Ritzwoller, M. H., Schulte-Pelkum, V., & Lin, F. C. (2013). Joint inversion of surface wave dispersion and receiver functions: a Bayesian Monte-Carlo approach. *Geophysical Journal International*, *192*(2), 807-836.

Simonyan, K., and Zisserman, A., (2014). Very deep convolutional networks for large-scale image recognition, available at https://arxiv.org/abs/1409.1556 (last accessed January 2020).

Tape, C., Liu, Q., Maggi, A., & Tromp, J. (2010). Seismic tomography of the southern California crust based on spectral-element and adjoint methods. *Geophysical Journal International*, *180*(1), 433-462.

Wang, Y., Ge, Q., Lu, W., & Yan, X. (2019). Seismic impedance inversion based on cycle-consistent generative adversarial network. In *SEG Technical Program Expanded Abstracts 2019* (pp. 2498-2502). Society of Exploration Geophysicists.

Wathelet, M. (2008). An improved neighborhood algorithm: Parameter conditions and dynamic scaling. Geophysical Research Letters, 35, L09301. https://doi.org/10.1029/2008GL033256.

Wu, X., Liang, L., Shi, Y., and Fomel, S. (2019). FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation: Geophysics, 84(3), IM35-IM45.

Yang, Y., Engquist, B., Sun, J. & Hamfeldt, B.F., 2018. Application of optimal transport and the quadratic Wasserstein metric to full-waveform inversion, *Geophysics*, **83**(1), R43–R62.

Yi, Z., Zhang, H., Tan, P., and Gong, M., 2017, Dualgan: Unsupervised dual learning for image-to-image translation: Proceedings of the IEEE International Conference on Computer Vision, 2849-2857.

Zelt, C. A., K. Sain, J. V. Naumenko, and D. S. Sawyer, 2003, Assessment of crustal velocity models using seismic refraction and reflection tomography: Geophysical Journal International, **153**(3), 609–626.

Zhang, X., Jia, Z., Ross, Z. E., & Clayton, R. W. (2020). Extracting dispersion curves from ambient noise correlations using deep learning. *IEEE Transactions on Geoscience and Remote Sensing*.

Zhu, W., & Beroza, G. C. (2019). PhaseNet: a deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, *216*(1), 261-273.

Zhu, J. Y., Park, T., Isola, P., and Efros, A. A., 2017, Unpaired image-to-image translation using cycle-consistent adversarial networks: Proceedings of the IEEE International Conference on Computer Vision, 2223-2232.

Zigone, D., Ben-Zion, Y., Campillo, M., & Roux, P. (2015). Seismic tomography of the Southern California plate boundary region from noise-based Rayleigh and Love waves.

Pure and Applied Geophysics, *172*(5), 1007–1032. https://doi.org/10.1007/s00024-014-0872-1

Figure captions

Figure 1. The flowchart for (a) convolutional neural network (CNN) and (b) generative adversarial network (GAN) algorithms. The part of chart outlined by the blue dashed rectangular is further explained in Figure 3.

Figure 2. Map of the Southern California plate boundary region. The thick black lines depict surface traces of major faults, coastlines, and state boundaries. The yellow triangles are seismic stations used in Qiu et al. (2019) to derive the Rayleigh wave velocity dispersion maps with a grid size of 0.05°×0.05° (grid lines). Three cross sections (i.e., DD' to FF'; blue lines) of the final Vs model are presented in Figure 7. The cross sections DD' to FF' are of the same locations as those in Qiu et al. (2019). SAF – San Andreas Fault; SJF – San Jacinto Fault; EF – Elsinore Fault; ECSZ – Eastern California Shear Zone.

Figure 3. The algorithm comparison between convolutional neural network (CNN), generative adversarial network (GAN), and Wasserstein Cycle-GAN (Wcycle-GAN). The suffix m and d represents shear velocity model and dispersion data, respectively. CNN (a) computes point-wise misfit (estimation loss: $\mathcal{L}_{est}$) between real samples and translated samples generated by a model generative network (G_m). The GAN (b) introduces an adversarial network (D_m) and computes the difference between distributions of real and generated samples using adversarial loss ($\mathcal{L}_{adv}$), by updating generator and discriminator separately in a single iteration. The Wcycle-GAN (c) uses Wasserstein metric for adversarial loss in (b). Besides, a data generative subnet (G_d) is incorporated to learn the modeling of velocity model to dispersion data, together with a corresponding data discriminator (D_d). The use of G_d enables an extra constraint, the cycle consistent loss ($\mathcal{L}_{cyc}$), which is estimated by the misfit between the input real sample and reconstructed sample. The complete Wcycle-GAN penalty function is a linear combination of three types of the loss function ($\mathcal{L}_{est}$, $\mathcal{L}_{adv}$, and $\mathcal{L}_{cyc}$).

Figure 4. Comparison of depth slices for the output 3-D Vs models from four different methods. Depth slices at 5 km (left column) and 10 km (right column) for (a), (e) CNN-based model; (b), (f) Wcycle-GAN (WCGAN) based model but without using the unlabeled data in training; (c), (g) the proposed Wcycle-GAN based model; (d), (h) the Eikonal tomography

model from Qiu et al., (2019), respectively. Black lines delineate the coastline and light grey lines depict the surface traces of the major faults in southern California.

Figure 5. Chi-square misfit histograms for Vs models derived from six different methods: (a) CNN-based method; (b) Wcycle-GAN (WCGAN) based model but without using the unlabeled data in training; (c) Wcycle-GAN based model with full labeled data and unlabeled data; (d) model from Qiu et al. (2019); (e) Wcycle-GAN based model but using down sampled 1890 label data; (f) Wcycle-GAN based method with location information added as extra channels in the network training.

Figure 6. Depth slices of shear velocity model at 5 km (left column) and 10 km (right column) for (a), (c) Wcycle-GAN (WCGAN) based model but using down sampled 1890 unlabeled data and (b), (d) Wcycle-GAN based model with location information added as extra prior information in the network training (WCGAN + Position).

Figure 7. Cross sections (blue lines in Figure 2) of the Vs model resolved from the Wcycle-GAN network trained with location information. Colors in panels on the left show the velocity values, whereas velocity perturbations, relative to the 1-D average Vs depth profile, in percentage are illustrated on the right. The black curve depicts an exaggerated topography variation. The black dashed line in each profile represents the inferred fault planes for SJF in DD' and SAF in EE' and FF'. The dashed ellipse in DD' outlines a low velocity anomaly that is likely associated with rock damaged inferred in Ben-Zion & Zaliapin (2019). EF = Elsinore Fault; SJF = San Jacinto Fault; ECSZ = Eastern California Shear Zone.

Figure 1.

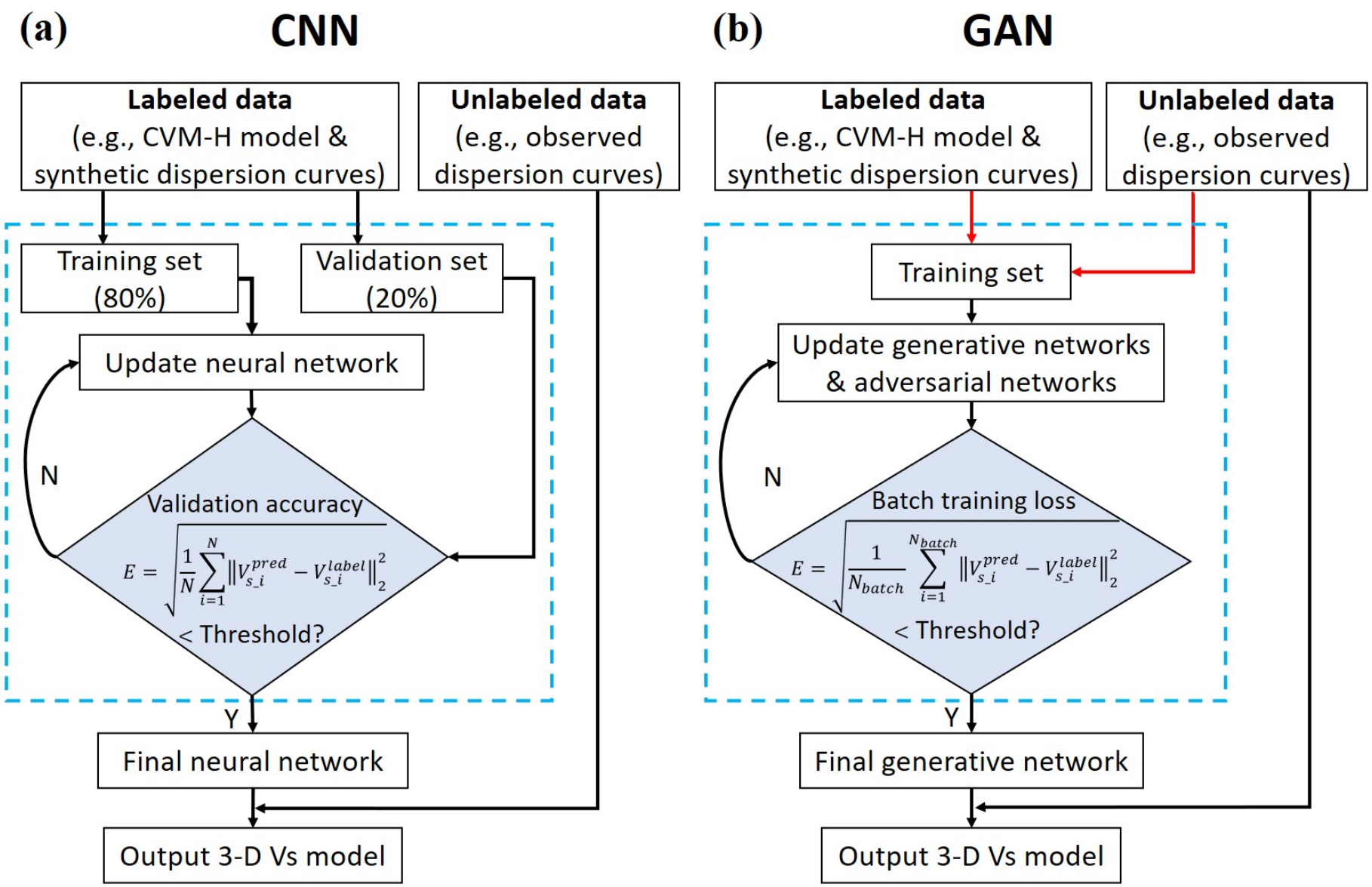

(a)
CNN
Labeled data
(e.g., CVM-H model &
synthetic dispersion curves)
Unlabeled data
(e.g., observed
dispersion curves)
Training set
(80%)
Validation set
(20%)
Update neural network
N
Validation accuracy
$E = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\|V_{s\_i}^{pred} - V_{s\_i}^{label}\|_2^2}$
< Threshold?
Y
Final neural network
Output 3-D Vs model
(b)
GAN
Labeled data
(e.g., CVM-H model &
synthetic dispersion curves)
Unlabeled data
(e.g., observed
dispersion curves)
Training set
Update generative networks
& adversarial networks
N
Batch training loss
$E = \sqrt{\frac{1}{N_{batch}}\sum_{i=1}^{N_{batch}}\|V_{s\_i}^{pred} - V_{s\_i}^{label}\|_2^2}$
< Threshold?
Y
Final generative network
Output 3-D Vs model

Figure 2.

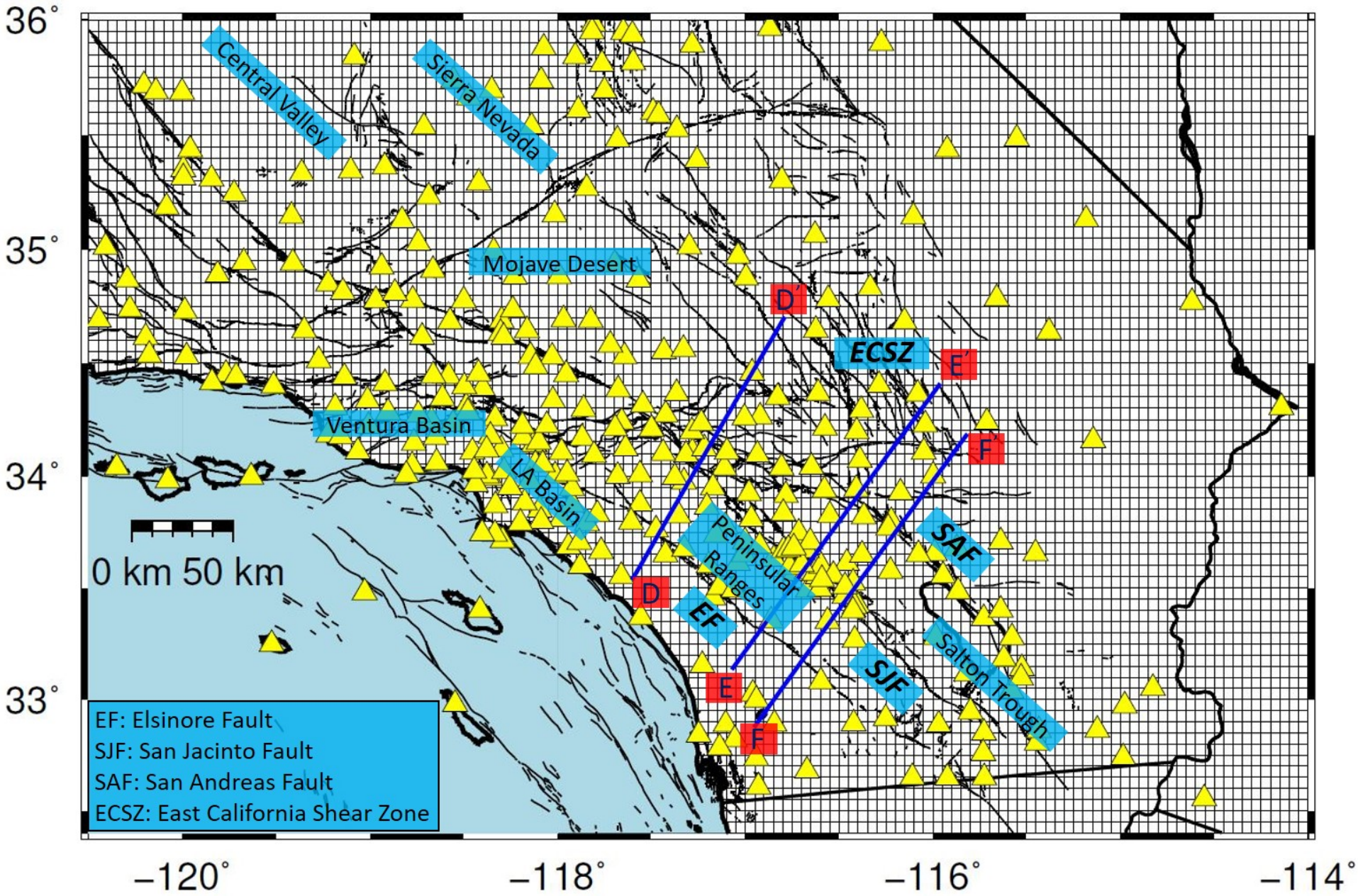
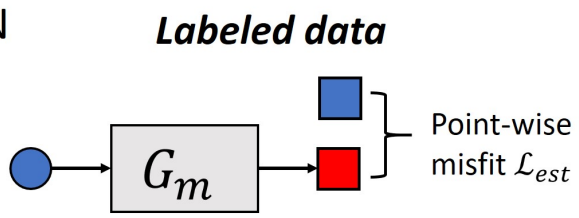

36°
35°
34°
33°
−120°
−118°
−116°
−114°
Central Valley
Sierra Nevada
Mojave Desert
Ventura Basin
LA Basin
Peninsular Ranges
ECSZ
SAF
EF
SJF
Salton Trough
D
D'
E
E'
F
F'
0 km 50 km
EF: Elsinore Fault
SJF: San Jacinto Fault
SAF: San Andreas Fault
ECSZ: East California Shear Zone

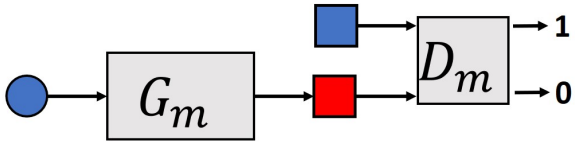
Figure 3.

a. CNN

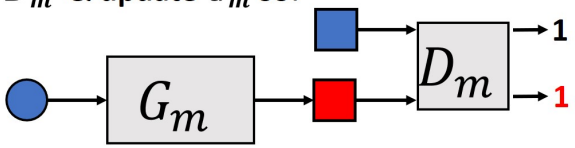


b. GAN: $\mathcal{L}_{adv}$

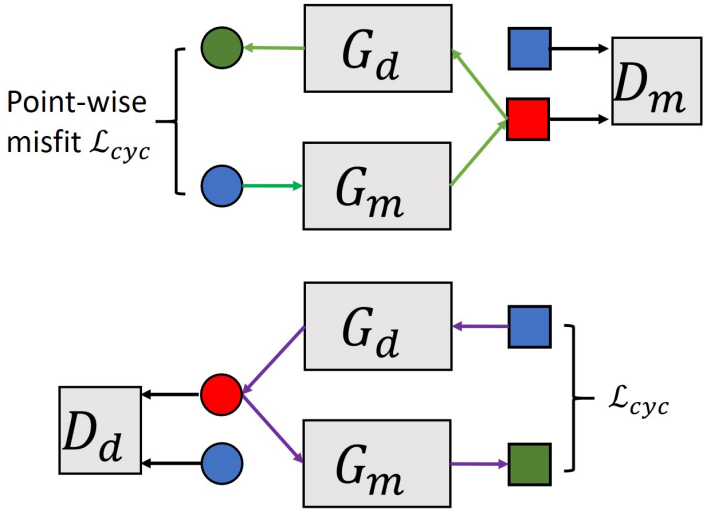
(1) Fix G_m & update D_m so:



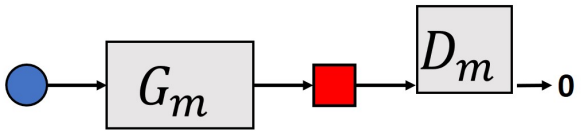
(2) Fix D_m & update G_m so:

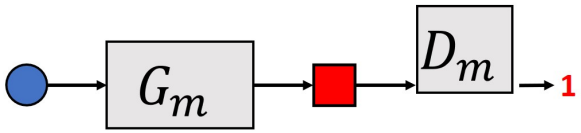


c. Wasserstein Cycle-GAN



Unlabeled data





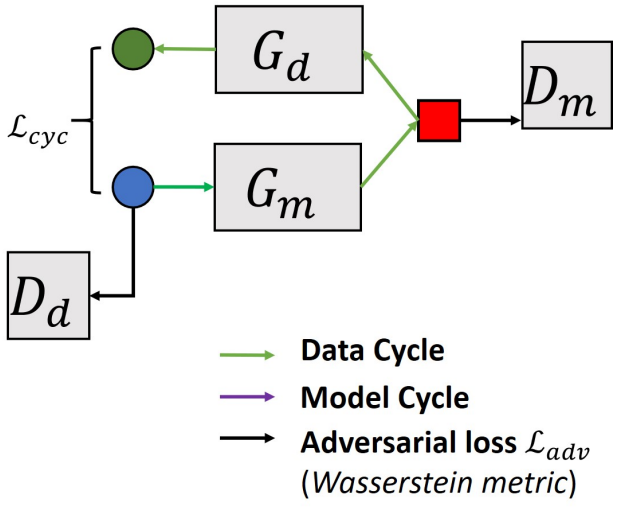


Figure 4.

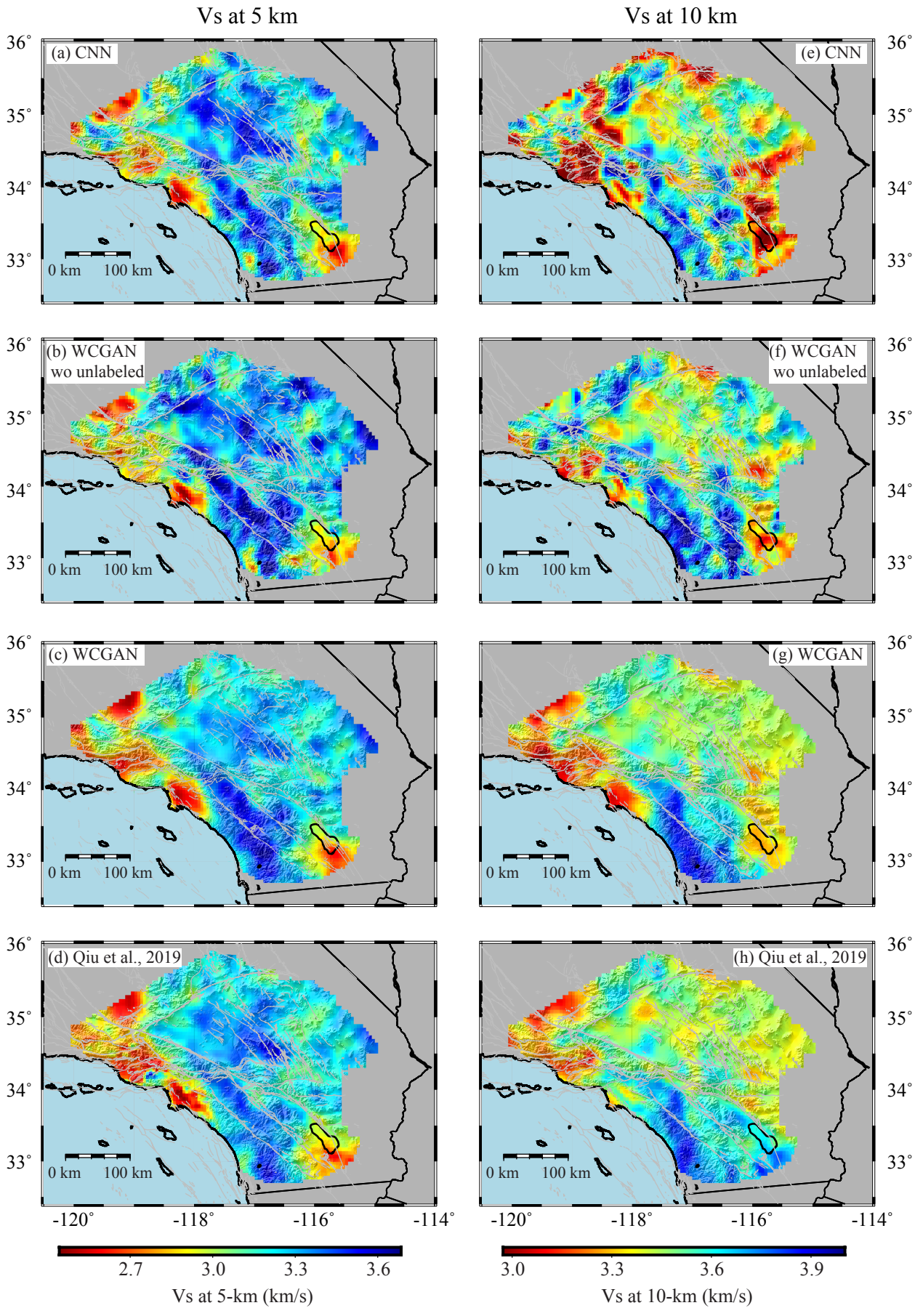

Vs at 5 km
Vs at 10 km
(a) CNN
(b) WCGAN
wo unlabeled
(c) WCGAN
(d) Qiu et al., 2019
(e) CNN
(f) WCGAN
wo unlabeled
(g) WCGAN
(h) Qiu et al., 2019
0 km 100 km
36°
35°
34°
33°
-120°
-118°
-116°
-114°
2.7 3.0 3.3 3.6
Vs at 5-km (km/s)
3.0 3.3 3.6 3.9
Vs at 10-km (km/s)

Figure 5.

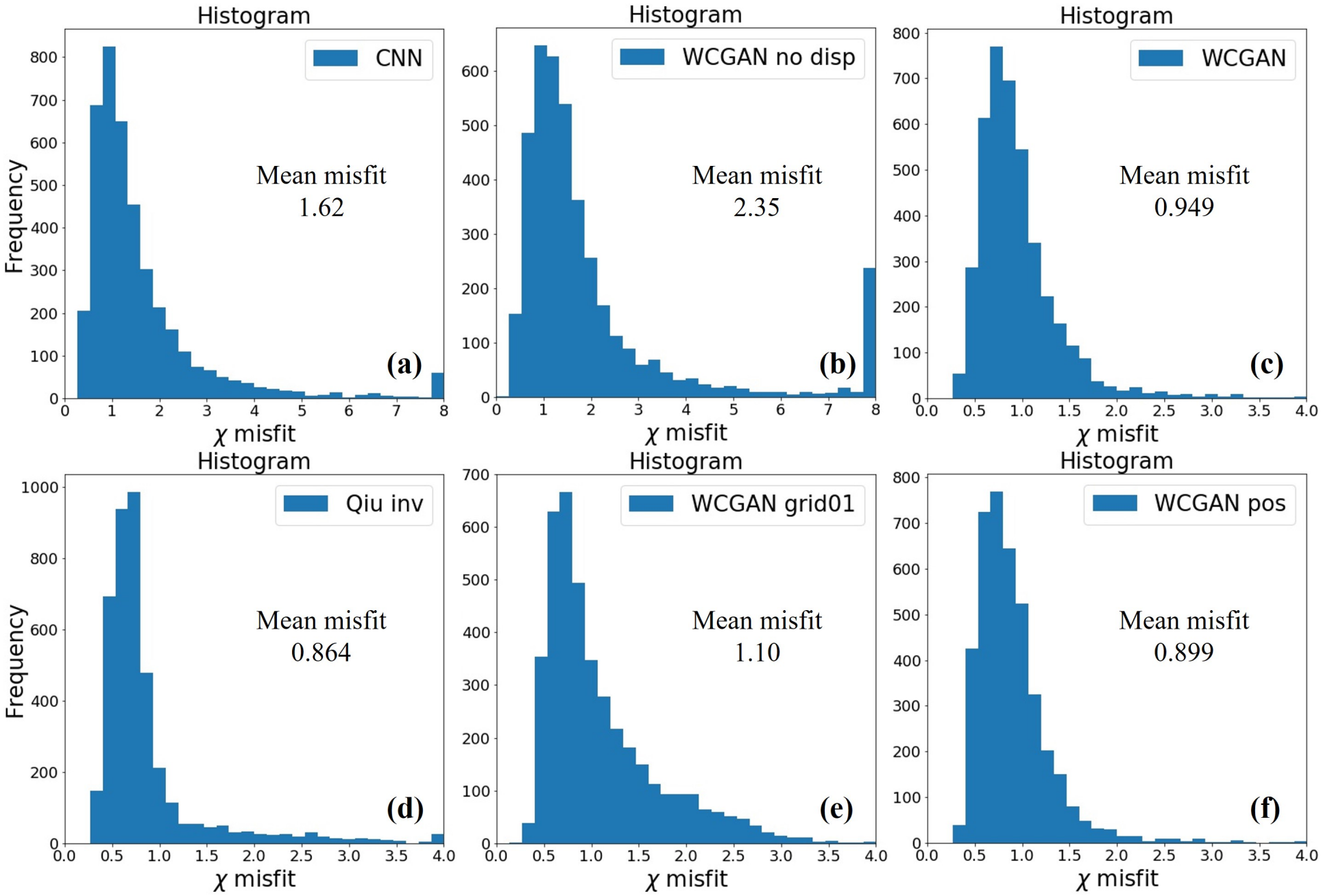

Histogram
CNN
Mean misfit
1.62
(a)
Frequency
χ misfit
Histogram
WCGAN no disp
Mean misfit
2.35
(b)
χ misfit
Histogram
WCGAN
Mean misfit
0.949
(c)
χ misfit
Histogram
Qiu inv
Mean misfit
0.864
(d)
Frequency
χ misfit
Histogram
WCGAN grid01
Mean misfit
1.10
(e)
χ misfit
Histogram
WCGAN pos
Mean misfit
0.899
(f)
χ misfit

Figure 6.

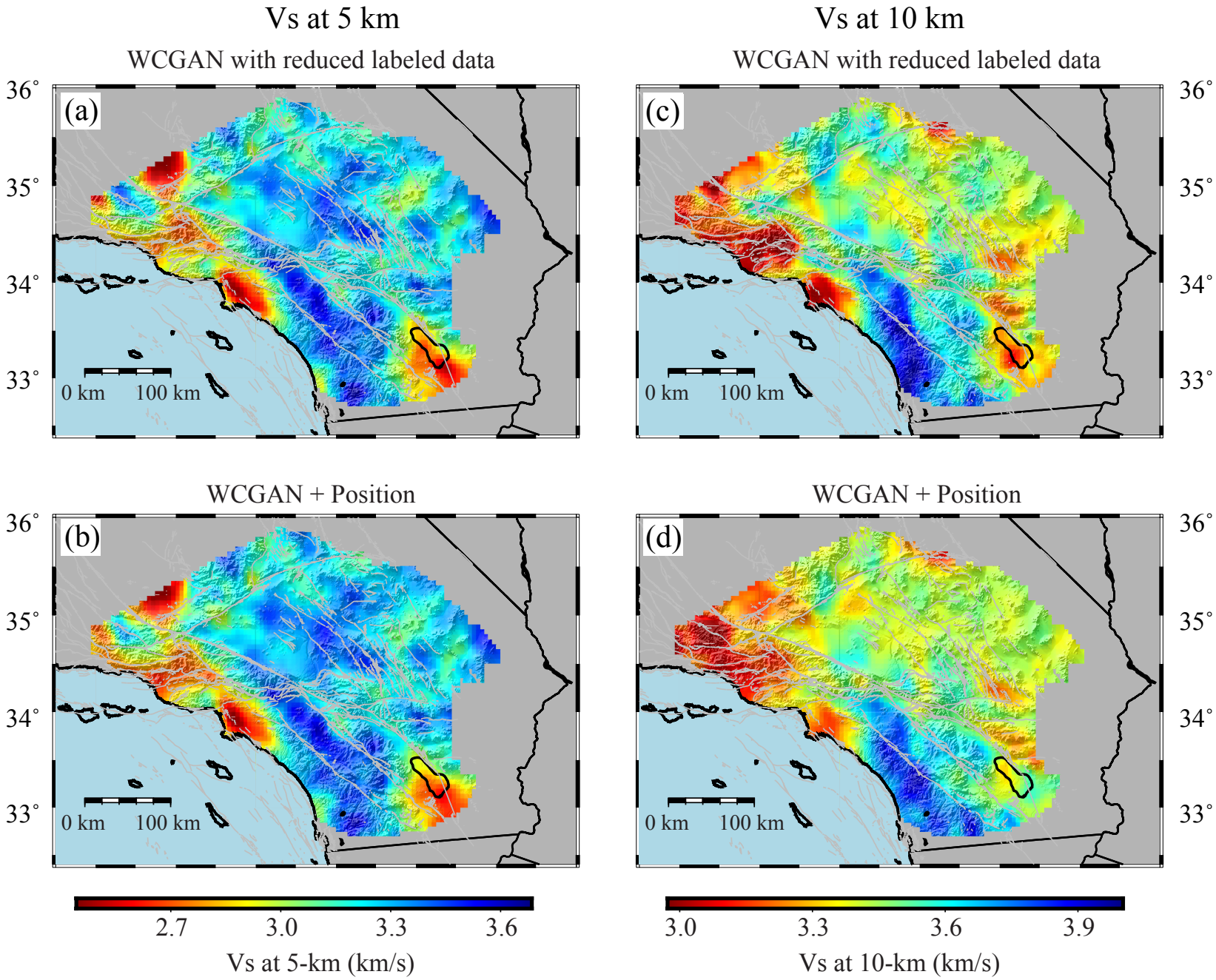
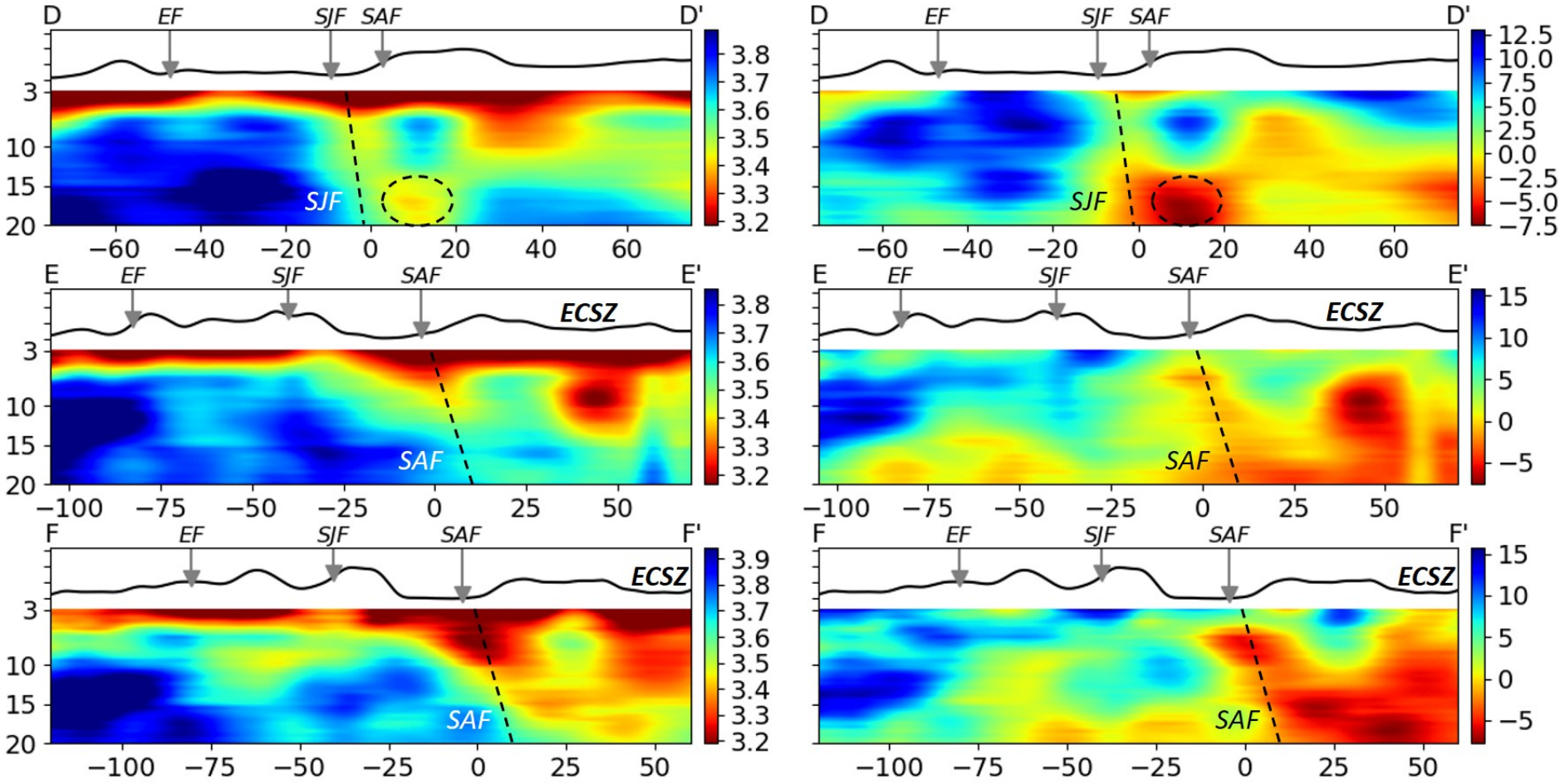

Vs at 5 km
WCGAN with reduced labeled data
Vs at 10 km
WCGAN with reduced labeled data
(a)
(c)
WCGAN + Position
WCGAN + Position
(b)
(d)
36°
35°
34°
33°
0 km
100 km
2.7
3.0
3.3
3.6
Vs at 5-km (km/s)
3.0
3.3
3.6
3.9
Vs at 10-km (km/s)

Figure 7.


D
EF
SJF
SAF
D'
SJF
3
10
15
20
−60
−40
−20
0
20
40
60
3.8
3.7
3.6
3.5
3.4
3.3
3.2
12.5
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
E
E'
ECSZ
SAF
−100
−75
−50
−25
0
25
50
15
10
5
0
−5
F
F'
3.9