

Feasibility of Deep Learning in Shear Wave Splitting analysis using Synthetic-Data Training and Waveform Deconvolution

Megha Chakraborty ^{1,2}, Georg Rumpker ^{* 1,2}, Wei Li ¹, Johannes Faber^{1,3}, Frederik Link ⁴, Nishtha Srivastava ^{1,2}

¹Frankfurt Institute for Advanced Studies, 60438 Frankfurt, Germany, ²Institute of Geosciences, Goethe University Frankfurt, 60438 Frankfurt, Germany, ³Institute for Theoretical Physics, Goethe University Frankfurt, 60438 Frankfurt, Germany, ⁴The Department of Earth & Planetary Sciences, Yale University, New Haven CT 06511, United States

Author contributions: *Conceptualization:* Georg Rumpker and Megha Chakraborty. *Methodology:* Megha Chakraborty, Georg Rumpker, Johannes Faber and Nishtha Srivastava. *Validation:* Wei Li. *Formal Analysis:* Megha Chakraborty. *Data:* Frederik Link. *Writing - Original draft:* Megha Chakraborty. *Writing - Review & Editing:* Megha Chakraborty, Georg Rumpker, Wei Li, Johannes Faber, Frederik Link and Nishtha Srivastava. *Funding acquisition:* Nishtha Srivastava, Georg Rumpker.

Abstract Teleseismic shear-wave splitting analyses are often performed by reversing the splitting process through the application of frequency- or time-domain operations aimed at minimizing the transverse-component energy of waveforms. These operations yield two splitting parameters, ϕ (fast-axis orientation) and δt (delay time). In this study, we investigate the applicability of a baseline recurrent neural network, SWS-Net, for determining the splitting parameters from pre-selected waveform windows. Due to the scarcity of sufficiently labelled real waveform data, we generate our own synthetic dataset to train the model. The model is capable of determining ϕ and δt with a root mean squared error (RMSE) of 9.7° and 0.14 s on noisy synthetic test data. The application to real data involves a deconvolution step to homogenize the waveforms. When applied to data from the USArray dataset, the results exhibit similar patterns to those found in previous studies with mean absolute differences of 9.6° and 0.16 s in the calculation of ϕ and δt , respectively.

Production Editor:
Gareth Funning
Handling Editor:
Lauren Waszek
Copy & Layout Editor:
Kirsty Bayliss

Signed reviewer(s):
Tom Hudson

Received:
September 27, 2023
Accepted:
January 6, 2024
Published:
March 22, 2024

1 Introduction

The analysis of seismic anisotropy serves as a unique tool for investigating the elusive dynamic processes occurring within the Earth's mantle. Inferring vertically and laterally varying anisotropic structures from surface-recorded seismic waveforms can provide vital constraints for geodynamic models of mantle deformation and flow. The study of teleseismic shear-wave splitting (SWS), a technique in use for over three decades, provides key insights about seismic anisotropy, aiding in the analysis of the dynamic processes within Earth's interior (Long and Silver, 2009; Reiss and Rumpker, 2017; Savage, 1999; Silver and Chan, 1991).

Two primary mechanisms contribute to the development of seismic anisotropy in the Earth's mantle: strain-induced lattice preferred orientation (LPO) of upper mantle minerals such as olivine (resulting from differential motion between the lithosphere and asthenosphere, and mantle flow) (Silver and Chan, 1991) and shape preferred orientation due to the presence of vertically aligned fluid-filled fractures, cracks, and microcracks (Holtzman and Kendall, 2010).

When a shear wave enters an anisotropic medium, it is split into two orthogonally polarized components that propagate at different speeds. This phenomenon can be described by two splitting parameters: the fast axis orientation (the polarization direction of the faster wave) ϕ , and the time delay between the two components δt . While ϕ represents the orientation of the anisotropic materials, δt measures the strength of anisotropy and the extent of the anisotropic material. Teleseismic phases are typically employed to investigate the anisotropic properties of the Earth. The most frequently used phases include SKS, SKKS, and PKS, and are collectively referred to as XKS phases. The conversion of these waves at the core-mantle boundary results in polarization in the direction of the back-azimuth (Jia et al., 2021; Liu and Gao, 2013; Reiss and Rumpker, 2017).

Several software codes have been developed to determine the splitting parameters ϕ and δt through grid search or correlation approaches. Examples of such codes can be found in the works of Silver and Chan (1991) such as Liu and Gao (2013); Savage et al. (2010); Teanby et al. (2004); Wüstefeld et al. (2008); Wüstefeld et al. (2010); Hudson et al. (2023). (Semi-)automatic ap-

*Corresponding author: rumpker@geophysik.uni-frankfurt.de

proaches were recently suggested by Reiss and Rümper (2017) and Link et al. (2022).

In this paper, we present a baseline model that demonstrates the potential of Deep Learning for the analysis of shear-wave splitting. In a recent study, Zhang and Gao (2022) utilized a Convolutional Neural Network (CNN) for waveform classification to automatically select reliable SWS measurements. However, to the best of our knowledge, a comprehensive analysis to infer anisotropic splitting parameters using deep learning has not yet been presented. Here, we introduce a deep learning model called SWSNet (Shear-Wave Splitting Network) to determine the splitting parameters from pre-selected waveform windows which are used by Link et al. (2022) for their analysis. Due to the lack of sufficient labelled data, the model is trained on synthetic data, simulated under the assumption of a single anisotropic layer (as is the case with traditional methods). A series of deconvolution and reconvolution steps are applied to both the real data and the synthetic data to ensure maximum resemblance. We demonstrate that SWSNet can produce results comparable to those of previous studies such as Liu et al. (2014) when applied to real data from the USArray and obtain mean absolute differences of 9.6° and 0.16 s in the calculation of ϕ and δt , respectively.

The major contributions of this paper can be summarised as follows: (i) to the best of our knowledge this is the first work to explore the applicability of deep learning in determining splitting parameters from waveforms; (ii) as we do not have sufficient labelled real data, we use synthetic data to train our model; (iii) a novel deconvolution and reconvolution approach is applied to remove the source and path effects from the real data to bridge the gap between ideal synthetic waveforms and real waveforms.

2 Methods and Results

For our study we use a supervised learning approach, which is a machine learning paradigm that relies on labelled data for training a model. The Deep Learning model we use learns to map the waveforms to the corresponding labels (in our case ϕ and δt) by minimising the difference between the true and predicted labels defined by the loss function.

In principle, labelled waveform data from shear-wave splitting analyses is available from publications and data archives (see e.g., Barruol et al., 2009). However, for our purposes, the amount of available data is limited, and the labelling may not be as uniform as would be required for efficient training. In order to overcome this limitation, we will use synthetic data as an alternative. Ideally, the generated synthetic waveforms will mimic the properties and characteristics of real data.

2.1 Modeling shear-wave splitting

In our approach, we consider waveform effects due to a single anisotropic layer, which is characterized by a horizontal symmetry axis (referred to as the “fast axis” and oriented at an angle ϕ measured clockwise from

North). A vertically incident shear wave splits into horizontally polarized fast and slow components, where the fast component aligns parallel to the symmetry axis, while the slow component is oriented perpendicular to it. Generally, these two quasi-shear waves propagate at different speeds, resulting in a separation by the delay time, δt , as they travel through the layer. A graphical representation of the coordinate systems used is given in Figure S1.

The equations to describe shear-wave splitting in layered structures have recently been summarized by Rümper et al. (2023). In the frequency domain, the radial and transverse displacement components, after passing through the layer, can be expressed as

$$\begin{pmatrix} u_1^{(r)} \\ u_1^{(t)} \end{pmatrix} = \begin{pmatrix} \cos \theta + i \sin \theta \cos 2\alpha & i \sin \theta \sin 2\alpha \\ i \sin \theta \sin 2\alpha & \cos \theta - i \sin \theta \cos 2\alpha \end{pmatrix} \begin{pmatrix} u_0^{(r)} \\ u_0^{(t)} \end{pmatrix} \quad (1)$$

where $\theta = \omega \delta t / 2$, $\alpha = \beta - \phi$ is the angular difference between back-azimuth and fast axis, and index 0 denotes waveforms before passing through the anisotropic layer. For XKS phases in a radially symmetric Earth, we can assume that $u_0^{(t)} = 0$ upon entering the (first) anisotropic layer on the receiver-side leg of the ray path, such that

$$u_1^{(r)} = (\cos \theta + i \sin \theta \cos 2\alpha) u_0^{(r)} \quad (2)$$

$$u_1^{(t)} = i \sin \theta \sin 2\alpha u_0^{(r)} \quad (3)$$

Note, that for relatively long periods, $T \gg \delta t$ (to first order in θ), this simplifies to

$$u_1^{(r)} \simeq (1 + i\omega \frac{\delta t}{2} \cos 2\alpha) u_0^{(r)} \quad (4)$$

$$u_1^{(t)} \simeq i\omega \frac{\delta t}{2} \sin 2\alpha u_0^{(r)} \quad (5)$$

where the factor $i\omega$ yields a derivative of the radial-component waveform and the amplitude is modulated by $\sin 2\alpha$. We will use this formulation in the development of our deconvolution approach, to be discussed in subsequent sections.

2.2 Deep Learning Analysis - Synthetic Data

We use synthetic data to train our model. The radial and transverse waveforms are generated with a sampling frequency of 50 Hz for back-azimuths between $0 - 360^\circ$ and fast axis ϕ ranging between $0 - 180^\circ$. Consequently α can vary between $0 - 180^\circ$, since ϕ and $\phi + 180^\circ$ represent the same fast axis orientation. We allow for possible values of δt between 0.2-2.0 seconds. Note that δt characterizes the anisotropy within the layer and is not equal to an “apparent” delay time which could be much larger (e.g. Silver and Savage, 1994). A total of 10^6 waveforms are used for the training process; this dataset is

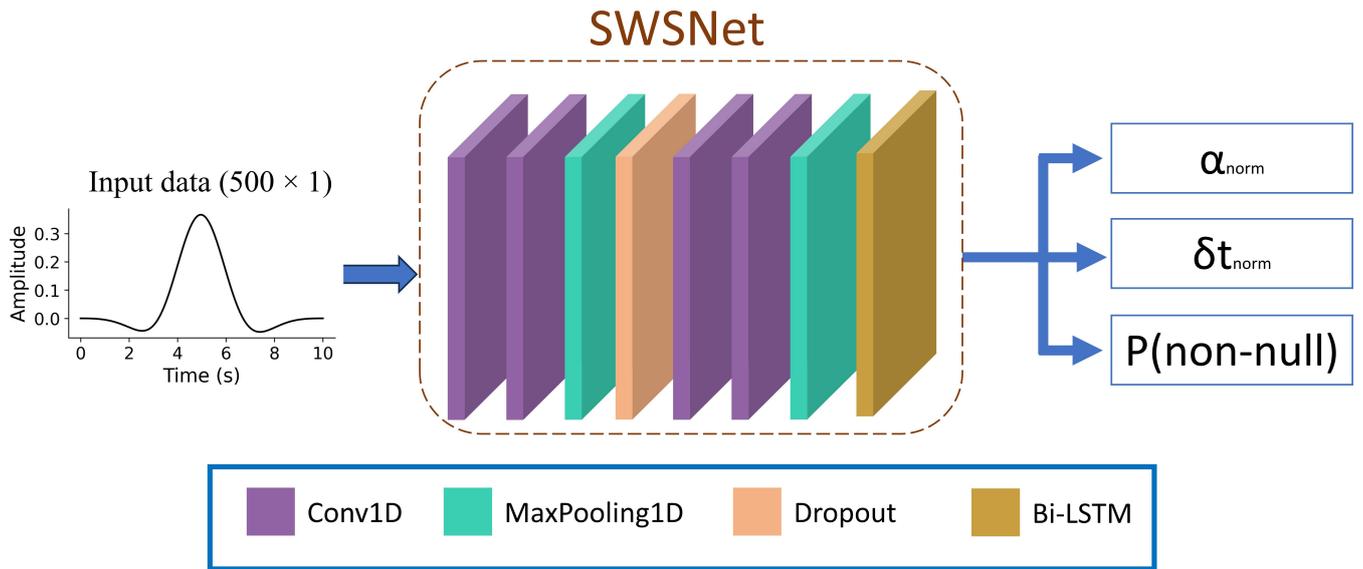


Figure 1 The architecture of SWSNet. The model takes as input the (deconvolved) transverse component and comprises of two blocks of 1D convolution and Maxpooling operations seperated by a Dropout layer with drop rate 30%, and followed by a bi-directional LSTM layers. The final outputs are the normalised values of α (α_{norm}) and δt (δt_{norm}) and the probability of the measurement being non-null.

split in a ratio of 80:20 for training and validation purposes.

Combinations of δt and ϕ are chosen from uniform random distributions for the ranges described above. We experiment with Convolutional layers (Kiranyaz et al., 2015), Bi-directional Long Short-Term Memory (Bi-LSTM) (Hochreiter and Schmidhuber, 1997) layers and a combination of both. Convolutional layers have been established to be effective at feature-extraction, while Bi-LSTMs are known for their ability to detect temporal dependence between these features. The model hyperparameters (such as the number of layers, the kernel size of filters in the convolutional layers, the dimensionality of the LSTM layers, the activation functions to be used etc.) are chosen by experimenting to maximise the model performance on validation data. Each 1D convolutional layer used has a Rectified Linear Unit (ReLU) activation function (Agarap, 2018). The model outputs three values corresponding to the probability of the measurement being non-null and the normalised predictions for δt and ϕ . The normalization of the target variables ensures that the mean squared error loss calculated for them are of the same order; this helps in the convergence of the loss function during backpropagation. Here, any measurement with $\alpha < 2$, $88 < \alpha < 92$, and $\alpha > 178$ is considered a null measurement. Since it is impossible for the model to discriminate between $\alpha = 0^\circ$, $\alpha = 90^\circ$ and $\alpha = 180^\circ$, the transverse component energy for all these cases being zero, we find that defining a non-null class helps the model learn to estimate α . A rectified linear-unit (ReLU) activation function (Agarap, 2018) is used for layers predicting α and δt while a sigmoid function is used to output the probability corresponding the measurement being non-null. A schematic example of such an architecture is shown in Figure 1 (note that Figure 1 shows the final architecture of SWSNet described in section 2.4); a more

detailed diagram is provided in the Supplementary information (Figure S7).

The model is trained using the Adam Optimiser (Kingma and Ba, 2014). We use a batch size of 256. Mean squared error and binary cross-entropy are used as loss functions for regression and classification respectively. Apart from using Maxpooling (Nagi et al., 2011) and Dropout (Srivastava et al., 2014) layers in the model architecture, early stopping (Prechelt, 2012) is used to further prevent overfitting, whereby training stops if validation loss does not decrease for 8 consecutive epochs. With this condition the model trains for 35 epochs.

2.2.1 Results - Synthetic Data

We train the Neural Network on a dataset with noise applied independently to the fast and slow components. Two types of noise are experimented with— random and Gaussian. The noise level is chosen from a random normal distribution with mean 30% and standard deviation 10%. Some examples for these datasets can be found in Figures S2 and S3 in the Supplementary Materials. Figure 2 (a) and (b) show the results when the model trained on synthetic data with random noise is tested on an independent test dataset also with random noise. As can be seen from Figure 2, the deep learning model has RMSE 5.9° and 0.12 s in the predictions of α and δt respectively. The corresponding figure (S4) for data with Gaussian noise is provided as Supplementary information.

2.3 Application to real data from USArray

We apply our model to pre-selected waveforms from the USArray dataset and compare our results with Liu et al. (2014) and those calculated by the automatic Splitracer toolbox proposed by Link et al. (2022). To make sure that only meaningful results are used in the calculation

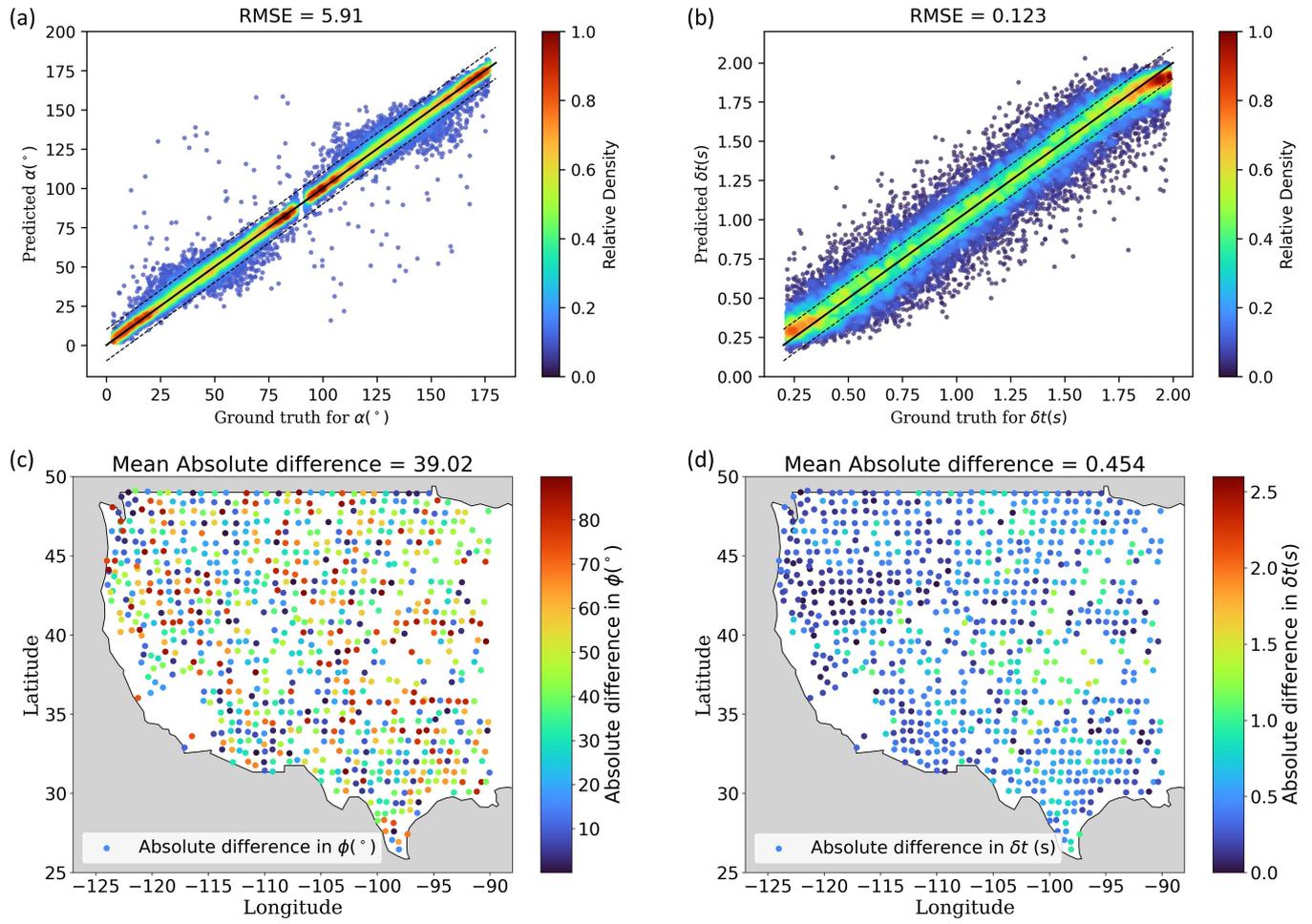


Figure 2 The relation between ground truth and predictions for (a) α and (b) δt when the model trained on synthetic training data contaminated with random noise is tested on synthetic test data contaminated by random noise; comparison between station-wise averages of (c) α and (d) δt calculated using the deep learning model and those given by Liu et al. (2014). (The corresponding figure for data contaminated with Gaussian noise can be found in the Supplementary Materials.)

of station averages, we perform a quality check on the estimations made by the neural network on given waveforms. We perform splitting inversion using the splitting parameters predicted by the neural network and check the percentage reduction in the transverse component energy (sum of squared amplitudes) as proposed by Silver and Chan (1991). An experimentally chosen threshold of 60% reduction in transverse component of energy is used to select the waveforms to be used for calculating station-wise averages for splitting parameters.

2.3.1 Direct application of the Model

When the model trained on the synthetic data is directly applied to the real data (radial and transverse components), the station-wise averages obtained for the splitting parameters differ significantly from those presented by Liu et al. (2014), as shown in Figure 2(c) and (d) (and also, S4 (c) and (d)). This happens as real waveforms look significantly different from the synthetic data. Thus a direct application of the trained model to the real waveforms renders unusable results. This necessitates an intermediate step to bridge the gap between the synthetic and real waveforms.

2.4 Deconvolution approach

Observed real waveforms are not only affected by anisotropic layering but may vary significantly due to different source mechanisms (and path effects). This poses a challenge to the training of the deep learning model, as it is not computationally feasible to include all waveform variations that may arise from different source mechanisms and complexities of the medium. Here, we choose a deconvolution approach to mitigate source effects and “homogenize” the waveforms. This method is similar to the one utilized in receiver-function processing, for instance Langston (1979); Owens et al. (1984); Ammon (1991).

We deconvolve both the radial and transverse components by the radial component. In the frequency-domain, in view of eq. (5), the procedure applied to real data can be described as follows:

$$u_*^{(r)} = u_1^{(r)} / u_1^{(r)} = 1 \quad (6)$$

$$\begin{aligned}
u_*^{(t)} &= i\omega \frac{\delta t}{2} \sin 2\alpha u_0^{(r)} / u_1^{(r)} \\
&\simeq i\omega \frac{\delta t}{2} \sin 2\alpha
\end{aligned}
\tag{7}$$

Note that we assumed $u_0^{(r)} / u_1^{(r)} \simeq 1$ in the derivation of eq. (7). This implies that the radial-component waveform is a sufficient representation of the incoming waveform (before it enters the anisotropic layer), which further limits the applicability to waveforms with periods much longer than δt ($T \gg \delta t$). The value of 1 for the radial component in the frequency domain corresponds to a δ -function in the time domain. For the transverse component, the factor $i\omega$ causes a time-domain derivative (of the unsplit waveform) with amplitude modulated by $\sin 2\alpha$. In a second step, the deconvolved components can now be convolved with a reference waveform, such as the normalised derivative of an exponential function (Figure S5, also shown in the radial component of Figure 6 described in Section 2.5), to yield a uniform radial component, and standard transverse component that depends on the two splitting parameters. Figure S6 shows the appearance of the transverse component for different α and δt pairs.

For real data, first the waveforms within the selected time windows are resampled at 50 Hz and then the mean is removed. For both the real and synthetic data the following steps are applied:

- A Hanning window is applied to smoothen the transition to zero amplitude at the boundaries of the time window.
- The data is zero-padded to have a uniform total of 2000 time samples corresponding to a 40 s time window.
- A butterworth lowpass filter with corner frequency of 1 Hz is applied to suppress higher-frequency noise.
- The radial component is deconvolved from both the radial and transverse components as per equations 6 and 7.
- The clean waveform shown in Figure S6 (also shown in the radial component of Figure 6 described in Section 2.5) is convolved with both the deconvolved waveforms (radial and transverse components).
- A Hanning window is applied to reduce the effect of possible sinusoidal “ringing” on the transverse component of the reconvolved data.
- The waveform is cropped to the central 10 seconds.
- Another Hanning window is applied followed by the normalisation of the data such that the absolute maximum amplitude in the transverse component is 1.

Figure 3 demonstrates the effectiveness of this method in uniforming the waveforms: while the two

waveforms with very close splitting parameters look significantly different due to different source mechanism and path effects, upon applying the deconvolution and reconvolution method described above, they look almost the same.

With this approach it is only the transverse component that carries meaningful information about the splitting parameters. Therefore we retrain our model on the transverse component of the de/reconvolved synthetic waveforms. Once again we experiment with different model architectures and hyperparameters; we find the best performing model to be the one shown in Figure 1. This model will henceforth be called the SWSNet (shear-wave splitting network). A detailed description of the hyperparameters used can be seen in Figure S7. As the input data structure is relatively simple a deeper network does not improve the results and a simple network is sufficient. Please note that the labels corresponding to α and δt are always scaled to be in the range 0-1 as this is known to benefit learning. A training data size of 10^6 waveforms is experimentally found to be optimum (Figure S8).

Once again, we experiment with both random and Gaussian noise. The performance of SWSNet on the synthetic dataset can be seen in Figure 4, and the corresponding figure for the Gaussian noise case is shown in Supplementary Materials (Figure S9). It can be noted here that the performance on the synthetic data worsens in comparison to Figure 2. This is because a major difference in the deconvolution approach, as compared to the method discussed in Section 2.2, is that when we train the model on the deconvolved data, only the transverse component carries the relevant information. Hence, the model is trained only on this component, as opposed to the previous method where both the radial and transverse components were used. Using two components might help the model learn the noise characteristics in the data, resulting in a smaller spread in the predicted parameters. However, despite this deterioration in performance on the synthetic test data, the use of the deconvolution method leads to much better generalizability when applied to real-world data, as will be seen in the subsequent discussion and in Figure 7.

2.5 Application to USArray

We apply our final SWSNet to the real data from USArray. The method used to find the station-wise averages is the same as described in Section 2.3; we experiment with the threshold for energy reduction once again, to choose the optimum threshold for our calculations. A threshold of 60% is determined to be optimum based on our observation in Figure 5 as it results in relatively lower mean absolute differences in the station-wise averages of both splitting parameters, while still retaining a good number of stations.

This leaves us with 8699 acceptable waveforms out of a total of 106323 ($\simeq 8.2\%$). This number is very similar to the 7.6% waveforms marked as ‘good’ category by Link et al. (2022). Some examples of SWSNet’s performance on individual waveforms can be seen in Figure 6 and the corresponding splitting parameters are summarised in

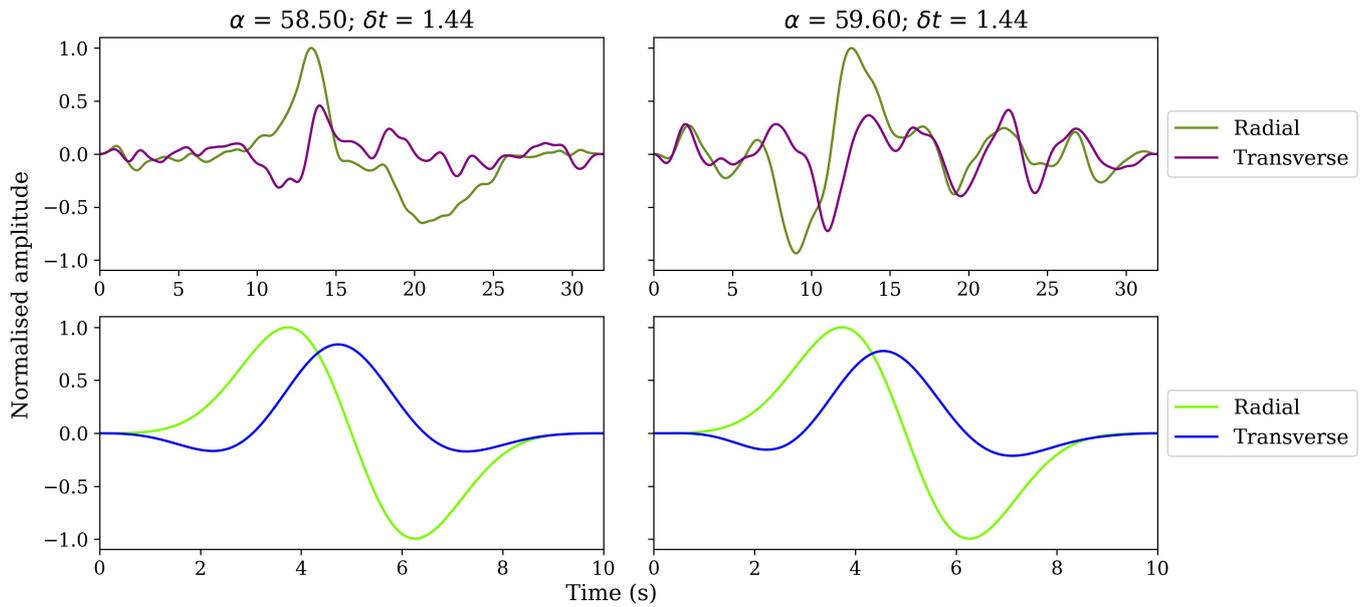


Figure 3 Two example waveforms with splitting parameters calculated by Link et al. (2022) very close to each other (top panels). The bottom panels show the corresponding waveforms after undergoing the deconvolution and reconvolution process described in Section 2.4. While the waveforms in their original form look significantly different due to their respective source mechanism, deconvolution makes them look almost the same, thereby eliminating the source and path effects.

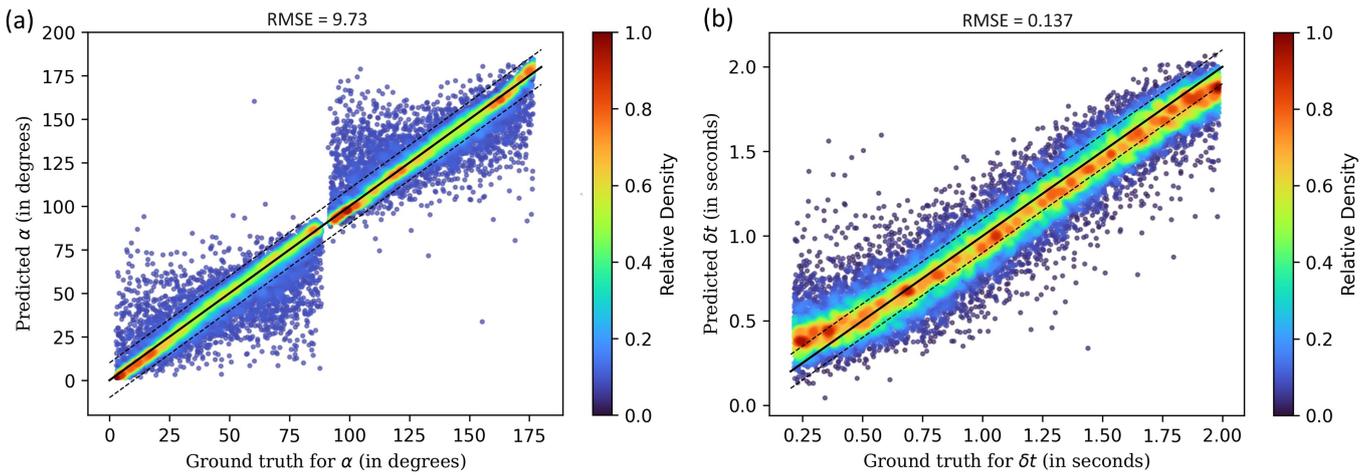


Figure 4 The performance of SWSNet on the synthetic test dataset when including the deconvolution approach. Both the training and test datasets are contaminated by random noise, with noise level chosen from a random normal distribution with mean 30% and standard deviation 10%.

Table 1. One can see the similarity between parameters calculated by SplitRacer, used in Link et al. (2022), and those calculated by SWSNet. A more detailed comparison with grid search results is included in table S1.

Figure 7 shows a visual representation of the station-averages of the splitting parameters calculated by SWSNet and Liu et al. (2014). Unlike Link et al. (2022), Liu et al. (2014) does not employ a joint splitting analysis, allowing for a more direct comparison with our approach, as it is also based on averaging results from individual split phases at a given station. Please note that this model is trained on data with random noise. The results for a model trained on Gaussian noise can be seen in Figure S10. While the performance of the models trained on random and Gaussian noise have comparable performance on the corresponding synthetic test

dataset, we observe throughout our experiments that the models trained on data with random noise fit the real data better. We suspect that this is because it is easier for the model to overfit the data with Gaussian noise during training as compared to when the noise is completely random. We also show the comparison between SWSNet and Link et al. (2022) in Figure S11.

3 Discussion

We apply the transverse energy reduction thresholds to SWSNet calculations when calculating station averages. This results in different sets of waveforms being used by this study and by Liu et al. (2014) for these calculations. However, in cases of multi-layer anisotropy, there is a strong dependence of splitting parameters on back-

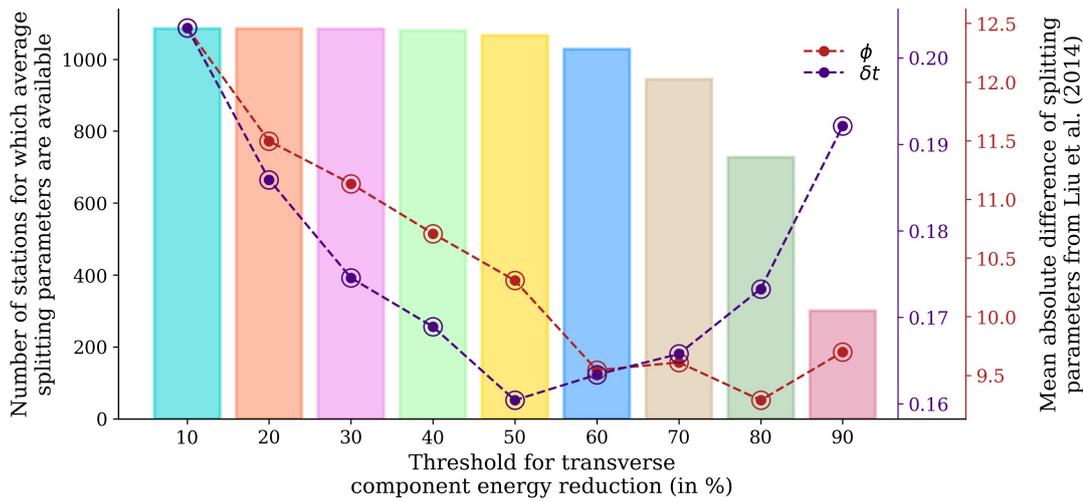


Figure 5 The effect of using different thresholds for energy reduction in the transverse component energy on the final calculation of the station averages. Based on these observations, a threshold of 60% is determined to be optimum. It results in relatively lower mean absolute differences in the station-wise averages of both splitting parameters, while still retaining a good number of stations.

Table 1 A comparison between splitting parameters for individual waveforms shown in Figure 6, calculated by Link et al. (2022) and SWSNet. A detailed comparison between grid search results, results from Link et al. (2022) and SWSNet can be found in table S1 in the Supplementary Information.

Event ID	ϕ (°)	ϕ (°)	δt (s)	δt (s)
	(Link et al., 2022)	(SWSNet)	(Link et al., 2022)	(SWSNet)
Y13A2008-05-09T22:15:04SKS	45	49.1	1.33	1.11
P59A2014-08-18T02:55:43SKS	86	88.4	0.82	0.80
121A2018-07-13T10:10:08SKS	9	12.7	1.02	1.03
D25K2017-07-15T12:35:42SKKS	66	71.5	1.44	1.50

azimuth, observed at many locations in the Western/Central U.S, which could significantly affect the splitting analysis if the events included are not identical.. As such, efforts were made to keep our method free of any requirements of prior knowledge; hence, the threshold was applied for the selection of waveforms irrespective of whether they were used in the calculations by Liu et al. (2014). To understand what the comparison would look like when using the same waveforms in both calculations, we examined a subset of waveforms included in both the station-average calculations by Liu et al. (2014) and in the data used for SWSNet calculations. We recalculated the station averages using just this data and conducted a comparison similar to Figure 5. This figure has been added to the supplementary materials as Figure S12. As expected, we found a closer alignment of the station averages in this case. Furthermore, we compared the splitting parameters calculated by SWSNet and those published by Liu et al. (2014) for individual waveforms, finding that the mean absolute difference for ϕ and δt are 11.08° and 0.239 s respectively.

As a further step to evaluate SWSNet’s performance in a multi-layer anisotropy case, we tested it on synthetic waveform data generated by considering two layers of anisotropy with the following two sets of splitting parameters:

- $\phi_1 = 20^\circ, \delta t_1 = 1.0 \text{ s}$ and $\phi_2 = 70^\circ, \delta t_2 = 1.0 \text{ s}$
- $\phi_1 = 20^\circ, \delta t_1 = 1.5 \text{ s}$ and $\phi_2 = 110^\circ, \delta t_2 = 0.5 \text{ s}$

where ϕ_1 and δt_1 represent the fast-axis orientation and time-delay in the first (lower) anisotropic layer, respectively, and ϕ_2 and δt_2 represent the fast-axis orientation and time-delay in the second (upper) anisotropic layer, respectively. Note that in the second case, the fast axes are perpendicular, such that the model effectively corresponds to a model with a single anisotropic layer. The resulting effective delay time is given by the difference between the delay times in each layer. We compare the variation of the splitting parameters with backazimuth to the theoretical curves calculated as per Silver and Savage (1994) (Figure 8). We find a good agreement between the expected apparent splitting parameters and those predicted by SWSNet, except when the resulting transverse components are very small. These small components correspond to null measurements (indicated by the gray patches in Figure 8) and are attributed by SWSNet to small (case 1) or variable (case 2) delay times. It is further interesting to note that the largest delay times predicted by SWSNet (2 s) agree with maximum delay times used in the training data for a single layer.

We further explore the different factors that affect the station-averaged results, and find the predominant factor to be the number of acceptable measurements for a given station, whereby the difference between the station averaged splitting parameters calculated by SWSNet and those from Liu et al. (2014) diminishes with an increased number of acceptable measurements corre-

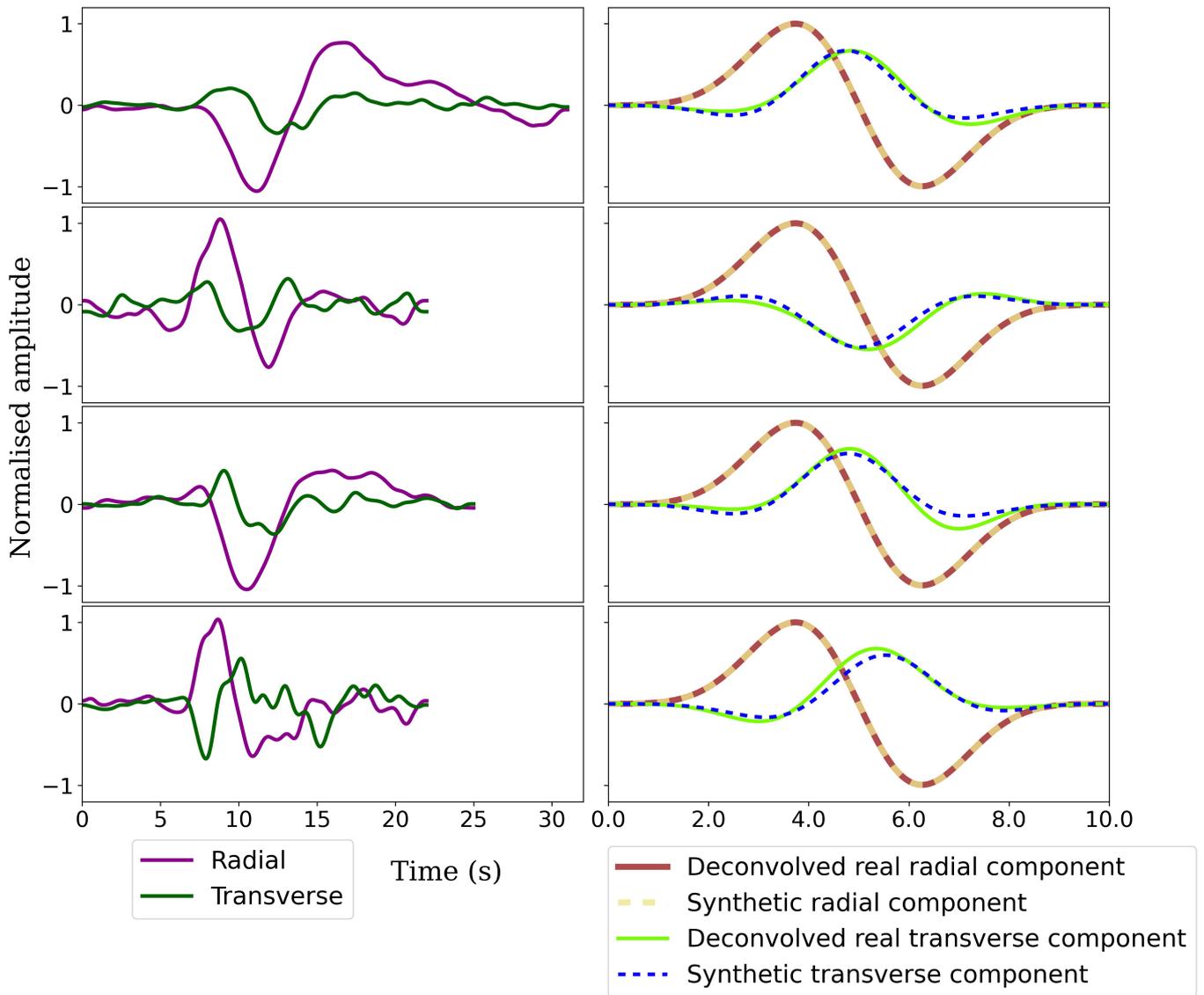


Figure 6 Examples of applying SWSNet to deconvolved real waveforms from the USArray Dataset. The left panel displays the original radial and transverse waveforms. The right panel shows a comparison between the deconvolved real waveforms and the synthetic counterparts, which are generated using the splitting parameters as predicted by SWSNet. The comparison reveals that the radial components are identical, as expected, while the transverse components exhibit a high degree of similarity. The corresponding splitting parameters can be found in Table 1.

sponding to a station (Figure S13).

We also compare our method against a simple grid search algorithm that, like previous studies, finds the splitting parameters for which (upon waveform inversion) the energy in the transverse component is the lowest. The grid search is done between 0.2-2 seconds for δt and 0-180° for α , with a grid spacing of 0.1 second and 1°, respectively. We plot the energy distributions for different combinations of α and δt for five randomly chosen events from five different stations, and find the parameters calculated by SWSNet to be quite close to those found by grid search and what is calculated by Link et al. (2022) (Figure S14). We further observe that grid search on average takes 3-6 times the amount of time taken by SWSNet to calculate splitting parameters for a single waveform.

4 Conclusion

In this study we introduce a baseline deep learning model SWSNet that has the potential to replace grid search methods used by previous studies to find splitting parameters for a waveform. Due to the dearth of labelled real data we train the model on synthetic data. We demonstrate that a direct application of model trained on the synthetic waveforms to real waveforms does not work well, the real waveform being affected by source mechanisms and path effects. This is resolved by using a deconvolution approach to minimise the difference between real and synthetic data. We retrain the model on transverse components of deconvolved synthetic waveforms contaminated by random noise, and show that the model learns to perform reasonably well in identifying the splitting parameters for such waveforms. We then apply our model to pre-selected wave-

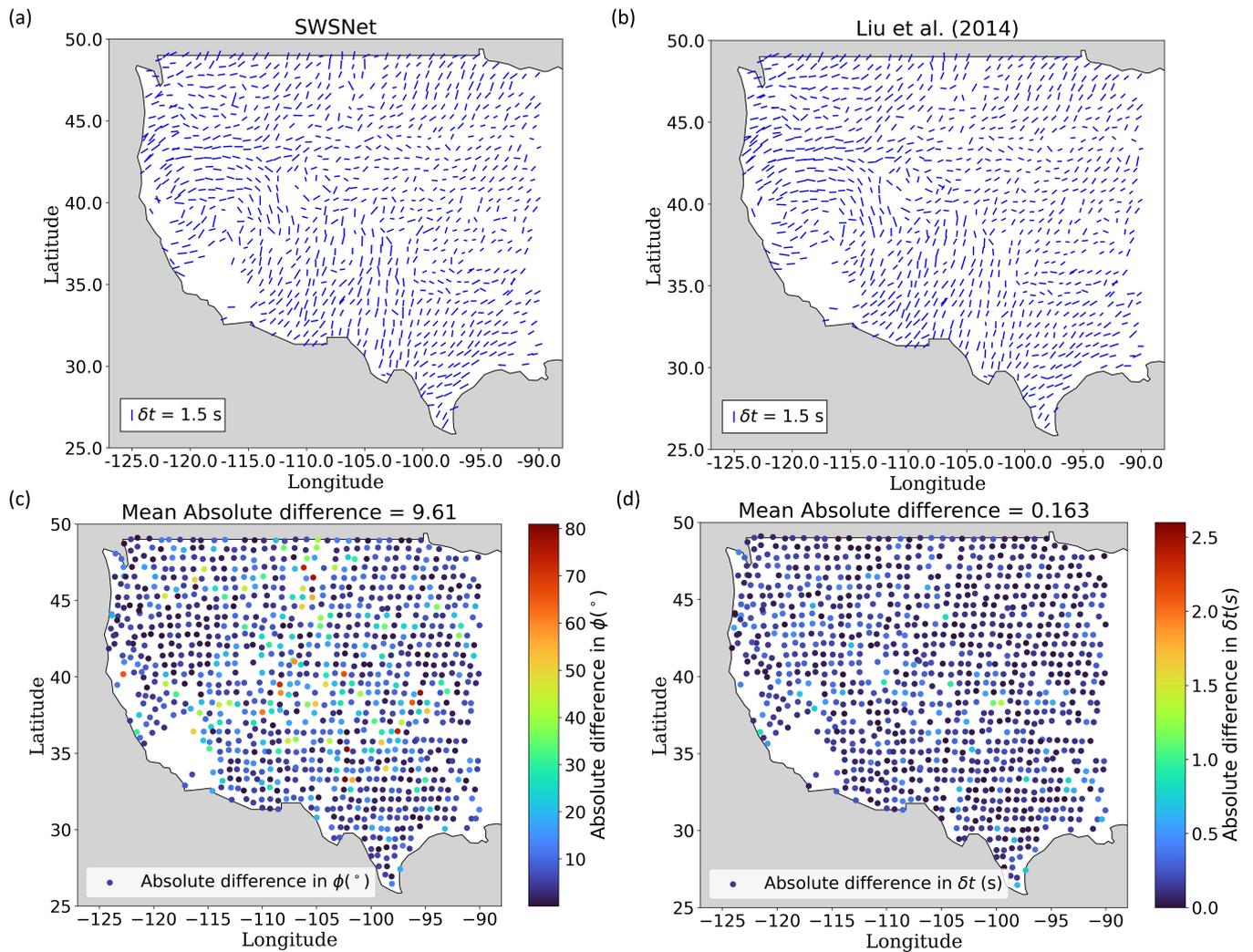


Figure 7 (a) Splitting parameters calculated by SWSNet (b) Splitting parameters calculated by Liu et al. (2014). The orientation of the straight lines is representative of the fast axis orientation while the length represents delay time. A similar general pattern is observed in both cases. (c) Station-wise comparison between ϕ calculated by SWSNet and Liu et al. (2014) (d) Station-wise comparison between δt calculated by SWSNet and Liu et al. (2014)

forms from the USArray dataset and show that the station averages calculated using SWSNet follow the same general trends as previous studies. We observe that the robustness of the proposed method increases with increased number of measurements for a given station.

The current version of the model is trained entirely on synthetic data, but in future versions real data can be added to the training set for improved representation. We would like to reiterate that the approach presented in this work is a baseline method to establish deep learning as a potential candidate for shear wave splitting studies. There are several avenues to further improve the results that would be explored in the future such as using a deeper model or using more complex data, for example, by considering multiple anisotropic layers instead of one. One major drawback of basic neural networks is their inability to provide uncertainty estimates Gawlikowski et al. (2023); therefore, providing uncertainty estimates would be another important avenue to explore in the future.

Acknowledgements

This research is supported by the “KI-Nachwuchswissenschaftlerinnen” - grant SAI 01IS20059 by the Bundesministerium für Bildung und Forschung - BMBF. GR gratefully acknowledges the support provided by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through grants RU 886/15 and RU 886/16. Calculations were performed at the Frankfurt Institute for Advanced Studies’ new GPU cluster, funded by BMBF for the project Seismologie und Artificielle Intelligenz (SAI). The authors are also thankful to Darius Fenner, Jonas Köhler and Kai Zhou for their kind suggestions and to Dr. Tom Hudson and an anonymous reviewer, whose valuable feedback and suggestions have greatly enriched the paper.

Data and code availability

The raw seismic waveforms used in this study are open for download from the IRIS Data Management Center under the network code TA (IRIS Transportable Array,

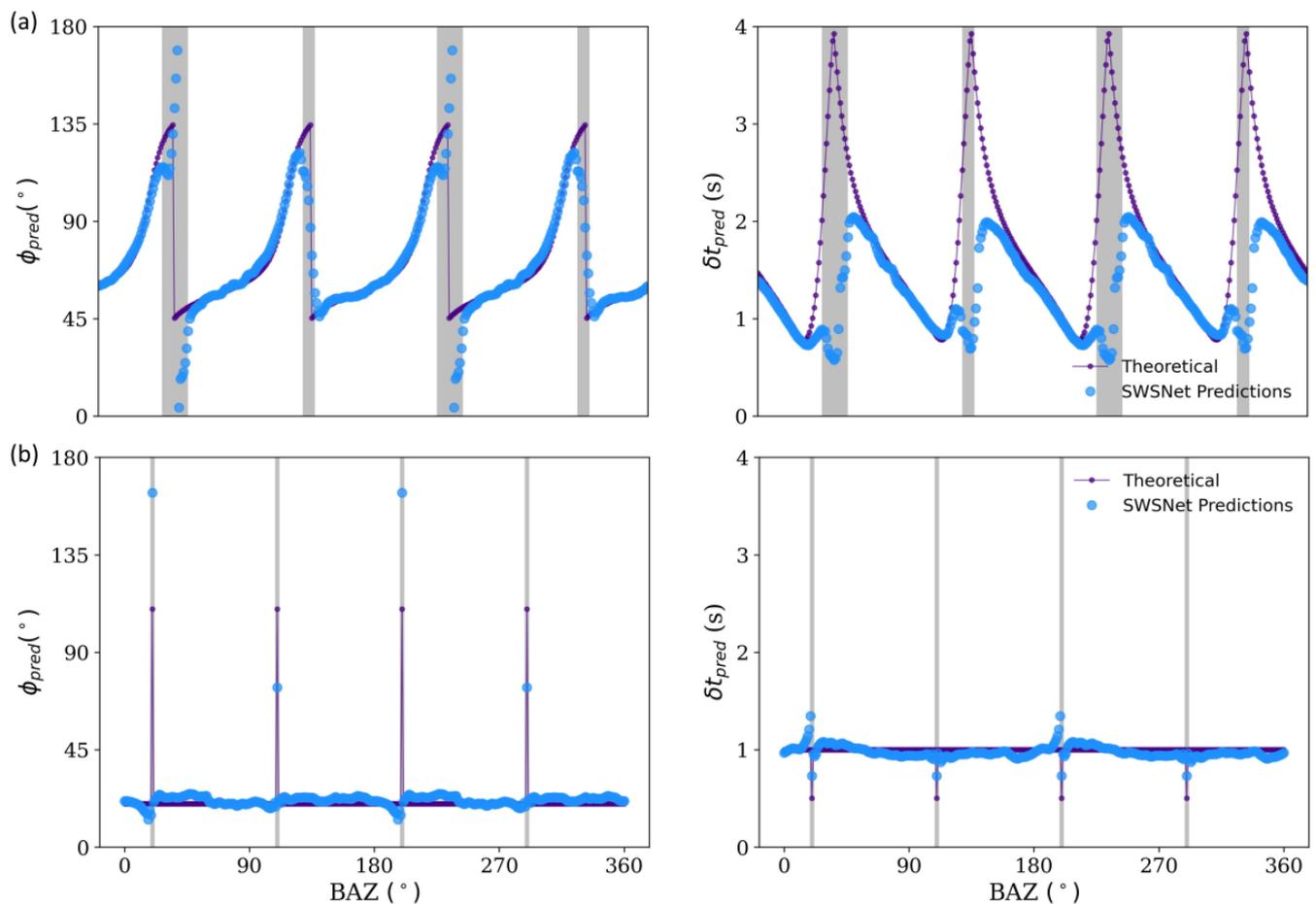


Figure 8 Comparison between apparent splitting parameters calculated according to Silver and Savage (1994) and those predicted by SWSNet for cases of 2-layer anisotropy with the following sets of splitting parameters: (a) $\phi_1 = 20^\circ$; $\phi_2 = 70^\circ$; $\delta t_1 = 1.0s$; $\delta t_2 = 1.0s$ and (b) $\phi_1 = 20^\circ$; $\phi_2 = 110^\circ$; $\delta t_1 = 1.5s$; $\delta t_2 = 0.5s$

2003). The event selection and corresponding labels used for training of SWSNet are available in the supplementary data alongside Link et al. (2022). The codes used for this study can be found at <https://github.com/srivastavaresearchgroup/SWSNet>.

Competing interests

The authors declare no competing interests.

References

- Agarap, A. F. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.
- Ammon, C. J. The isolation of receiver effects from teleseismic P waveforms. *Bulletin of the Seismological Society of America*, 81(6):2504–2510, 1991. doi: 10.1785/BSSA0810062504.
- Barruol, G., Wuestefeld, A., and Bokelmann, G. SKS-Splitting-database. *Université de Montpellier, Laboratoire Géosciences*, 2009. doi: 10.18715/sks_splitting_database.
- Gawlikowski, J., Tassi, C. R. N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., Shahzad, M., Yang, W., Bamler, R., and Zhu, X. X. A survey of uncertainty in deep neural networks. *Artificial Intelligence Reviews*, 56 (Suppl 1):1513–1589, 2023. doi: <https://doi.org/10.1007/s10462-023-10562-9>.
- Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.
- Holtzman, B. K. and Kendall, J.-M. Organized melt, seismic anisotropy, and plate boundary lubrication. *Geochemistry, Geophysics, Geosystems*, 11(12), 2010. doi: <https://doi.org/10.1029/2010GC003296>.
- Hudson, T. S., Asplet, J., and Walker, A. M. Automated shear-wave splitting analysis for single- and multi-layer anisotropic media. *Seismica*, 2023. doi: <https://doi.org/10.26443/seismica.v2i2.1031>.
- IRIS Transportable Array. USArray Transportable Array, 2003. doi: 10.7914/SN/TA.
- Jia, Y., Liu, K. H., Kong, F., Liu, L., and Gao, S. S. A systematic investigation of piercing-point-dependent seismic azimuthal anisotropy. *Geophysical Journal International*, 227(3): 1496–1511, 07 2021. doi: 10.1093/gji/ggab285.
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Kiranyaz, S., Ince, T., Hamila, R., and Gabbouj, M. Convolutional Neural Networks for patient-specific ECG classification. pages 2608–2611, 2015. doi: 10.1109/EMBC.2015.7318926.
- Langston, C. A. Structure under Mount Rainier, Washington, inferred from teleseismic body waves. *Journal of Geophysical Research: Solid Earth*, 84(B9):4749–4762, 1979. doi: <https://doi.org/10.1029/JB084iB09p04749>.

- Link, F., Reiss, M. C., and Rumpker, G. An automatized XKS-splitting procedure for large data sets: Extension package for SplitRacer and application to the USArray. *Computers & Geosciences*, 158:104961, 2022. doi: <https://doi.org/10.1016/j.cageo.2021.104961>.
- Liu, K. H. and Gao, S. S. Making Reliable Shear-Wave Splitting Measurements. *Bulletin of the Seismological Society of America*, 103(5):2680–2693, 10 2013. doi: [10.1785/0120120355](https://doi.org/10.1785/0120120355).
- Liu, K. H., Elsheikh, A., Lemnifi, A., Purevsuren, U., Ray, M., Refayee, H., Yang, B. B., Yu, Y., and Gao, S. S. A uniform database of teleseismic shear wave splitting measurements for the western and central United States. *Geochemistry, Geophysics, Geosystems*, 15(5):2075–2085, 2014. doi: <https://doi.org/10.1002/2014GC005267>.
- Long, M. D. and Silver, P. G. Shear Wave Splitting and Mantle Anisotropy: Measurements, Interpretations, and New Directions. *Surveys in Geophysics*, 30:407–461, 2009. doi: <https://doi.org/10.1007/s10712-009-9075-1>.
- Nagi, J., Ducatelle, F., Di Caro, G. A., Cireşan, D., Meier, U., Giusti, A., Nagi, F., Schmidhuber, J., and Gambardella, L. M. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 342–347, 2011. doi: [10.1109/ICSIPA.2011.6144164](https://doi.org/10.1109/ICSIPA.2011.6144164).
- Owens, T. J., Zandt, G., and Taylor, S. R. Seismic evidence for an ancient rift beneath the Cumberland plateau, Tennessee: A detailed analysis of broadband teleseismic P waveforms. *J. Geophys. Res.; (United States)*, 9 1984. doi: [10.1029/JB089iB09p07783](https://doi.org/10.1029/JB089iB09p07783).
- Prechelt, L. *Early Stopping — But When?*, pages 53–67. Springer Berlin Heidelberg, 2012. doi: [10.1007/978-3-642-35289-8_5](https://doi.org/10.1007/978-3-642-35289-8_5).
- Reiss, M. C. and Rumpker, G. SplitRacer: MATLAB Code and GUI for Semiautomated Analysis and Interpretation of Teleseismic Shear-Wave Splitting. *Seismological Research Letters*, 88(2A): 392–409, 01 2017. doi: [10.1785/0220160191](https://doi.org/10.1785/0220160191).
- Rumpker, G., Kaviani, A., Link, F., Reiss, M., and Komeazi, A. Testing observables for teleseismic shear-wave splitting inversions: ambiguities of intensities, parameters, and waveforms. *Ann. Geophys.*, 66, 2023. doi: <https://doi.org/10.4401/ag-8870>.
- Savage, M. K. Seismic anisotropy and mantle deformation: What have we learned from shear wave splitting? *Reviews of Geophysics*, 37(1):65–106, 1999. doi: <https://doi.org/10.1029/98RG02075>.
- Savage, M. K., Wessel, A., Teanby, N. A., and Hurst, A. W. Automatic measurement of shear wave splitting and applications to time varying anisotropy at Mount Ruapehu volcano, New Zealand. *Journal of Geophysical Research: Solid Earth*, 115(B12), 2010. doi: <https://doi.org/10.1029/2010JB007722>.
- Silver, P. G. and Chan, W. W. Shear wave splitting and subcontinental mantle deformation. *Journal of Geophysical Research: Solid Earth*, 96(B10):16429–16454, 1991. doi: <https://doi.org/10.1029/91JB00899>.
- Silver, P. G. and Savage, M. K. The Interpretation of Shear-Wave Splitting Parameters In the Presence of Two Anisotropic Layers. *Geophysical Journal International*, 119(3):949–963, 12 1994. doi: [10.1111/j.1365-246X.1994.tb04027.x](https://doi.org/10.1111/j.1365-246X.1994.tb04027.x).
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. <http://jmlr.org/papers/v15/srivastava14a.html>.
- Teanby, N. A., Kendall, J.-M., and van der Baan, M. Automation of Shear-Wave Splitting Measurements using Cluster Analysis. *Bulletin of the Seismological Society of America*, 94(2):453–463, 04 2004. doi: [10.1785/0120030123](https://doi.org/10.1785/0120030123).
- Wuestefeld, A., Al-Harrasi, O., Verdon, J. P., Wookey, J., and Kendall, J. M. A strategy for automated analysis of passive microseismic data to image seismic anisotropy and fracture characteristics. *Geophysical Prospecting*, 58(5):755–773, 2010. doi: <https://doi.org/10.1111/j.1365-2478.2010.00891.x>.
- Wüstefeld, A., Bokelmann, G., Zaroli, C., and Barruol, G. SplitLab: A shear-wave splitting environment in Matlab. *Computers & Geosciences*, 34(5):515–528, 2008. doi: <https://doi.org/10.1016/j.cageo.2007.08.002>.
- Zhang, Y. and Gao, S. S. Classification of Teleseismic Shear Wave Splitting Measurements: A Convolutional Neural Network Approach. *Geophysical Research Letters*, 49(12):e2021GL097101, 2022. doi: <https://doi.org/10.1029/2021GL097101>.

The article *Feasibility of Deep Learning in Shear Wave Splitting analysis using Synthetic-Data Training and Waveform Deconvolution* © 2024 by Megha Chakraborty is licensed under CC BY 4.0.