Optimising Deep-Learning First-Break Picking

Janis Heuel,[1]⋆ Arnaud Delsuc,[1,2]† Marie Weiel,[3,4] Daniel Coquelin,[3,4] Markus Götz,[3,4] and Andreas Rietbrock[1]

[1]*Geophysical Institute (GPI), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*

[2]*Observatoire de la Côte d'Azur, Université Côte d'Azur, CNRS, IRD, Géoazur, Nice, France*

[3]*Scientific Computing Center (SCC), Karlsruhe Institute of Technology (KIT), Eggenstein-Leopoldshafen, Germany*

[4]*Helmholtz AI, Germany*

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Accurate first-break picking is a key requirement for high-quality velocity model building, but manual picking remains time-intensive and error-prone. Existing single-station automatic approaches neglect spatial coherence across neighbouring traces and other deep-learning–based first-break picking approaches sacrifice temporal resolution through data downsampling, limiting their reliability. We present DeepFB, a U-Net–based neural network designed for robust automatic first-break picking in active-source seismic data. The model operates on overlapping chunks of multiple traces, preserving temporal resolution while exploiting spatial correlations. Automated hyperparameter optimisation using the evolutionary algorithm Populate yielded optimal model configurations without manual tuning. Among other hyperparameters, we tested whether model performance improves when training is performed with a reduced traveltime dataset or when noise augmentation techniques are used to improve first-break picking, particularly in noisy ocean-bottom seismometer records. Application to the HIPER2 experiment at the coastline of Ecuador demonstrates that DeepFB achieves picking accuracy comparable to manual picking, with residuals close to the manual picks from our test dataset. Tomographic inversions based on automatic and manual picks produce consistent velocity models, with only localised discrepancies near complex geological structures. DeepFB thus enables accurate, efficient, and scalable first-break picking, reducing manual workload while maintaining reliability.

Key words: first-break picking – active seismic – machine learning – semantic segmentation – attention layer – convolutional neural network

⋆ E-mail: janis-heuel@gmx.de
† E-mail: delsuc.arnaud@gmail.com

1 INTRODUCTION

In seismic exploration studies, such as continental crustal exploration programs (e.g. Zhao 2008; Almadani et al. 2015), large active-source seismic datasets are gathered. The first step in analysing these datasets is first-break picking, which is the process of identifying the earliest arrival of seismic energy at each receiver as accurately as possible (Sabbione & Velis 2010). The first break typically corresponds to the P-wave onset. These arrival times are fundamental for seismic refraction studies, where they are used, for example, in travel-time inversion to derive subsurface velocity models. Such models are essential for characterising tectonic settings, mapping structural features, and constraining the geologic framework of the survey area. First-breaks are conventionally picked automatically or semi-automatically using classical approaches such as the short-term average/long-term average algorithm (STA/LTA, Coppens 1985), modifications of STA/LTA (Sabbione & Velis 2010), or higher-order statistics (Yung & Ikelle 1997). These techniques still require a degree of manual intervention and can involve numerous parameters that must be carefully tuned by the analyst. This parameter selection process is time-consuming and can account for as much as 20–30% of the total data post-processing time (Sabbione & Velis 2010).

Nowadays, neural networks achieve performance comparable to analysts, while picking many more phases in less time and outperforming classical approaches (e.g. Münchmeyer et al. 2022). Picking seismic phases can be done, for example, by using PhaseNet (Zhu & Beroza 2018), which has learned from a large training dataset how to automatically label seismic phase arrivals. However, PhaseNet is only trained with single-station three-component seismic data and does not learn any features from the phase onsets at neighbouring stations, which is the case for active seismic experiments. Yin et al. (2023) also use single-station data to pick first onsets on a dataset with offsets up to 35 km. Recent studies (Yuan et al. 2018; Hu et al. 2019; Zhu et al. 2023; Mardan et al. 2024; Wen & Ma 2025) have shown that taking this spatial information into account leads to more accurate first-arrival picks. Studies like Hu et al. (2019) and Mardan et al. (2024) did not predict the first arrival at a distinct point in time; instead, they considered the problem of picking first arrivals as a semantic segmentation problem by segmenting a seismic section into times before and after the first arrival. However, one of the main problems of these published approaches is that the presented neural networks are always trained on the whole shot, i.e. one seismic section contains up to several hundreds traces. To keep the input data and the network architecture rather small, these seismic sections are downsampled, e.g. to images of the size 512×256 (Mardan et al. 2024).

In this study, we present a new approach to pick first-breaks with deep-learning that is a combination

of existing approaches. We refer to this method as DeepFB (Deep Neural Network for First-Breaks). In our approach, we take small chunks of the seismic section and predict the first arrivals within each data chunk. Since the chunks are overlapping, the neural network still learns the spatial relations and during the prediction phase we obtain several first-breaks for each trace. We optimised the model to identify the best set of hyperparameters by using the evolutionary optimisation algorithm Propulate (Taubert et al. 2023). In addition, we experimented with an automatic method to remove first-breaks with high uncertainties by computing the standard deviation from all predicted first-breaks of a single trace and discard the pick if the standard deviation exceeds a given threshold. To train and test DeepFB we used data from the "High resolution Imaging of the Pedernales Earthquake Rupture zone 2" (HIPER2) project at the coastline of Ecuador (Galve 2022). First-breaks have been manually picked for a first tomography study of the subducting plate at the coast of Ecuador (Delsuc et al. 2025). For this study, we trained DeepFB on the previously manually picked first-breaks and kept one part of the dataset for testing.

The paper is structured as follows: After this introduction, we show how the dataset at the coastline of Ecuador was gathered and how we created a training dataset to train DeepFB. Afterwards, we explain why PhaseNet (Zhu & Beroza 2018), one of the best-performing deep-learning frameworks for seismic phase picking (Münchmeyer et al. 2022), does not work for our seismic sections. Then we show how we set up DeepFB, how we find the best-performing model using Propulate and how to set up seismic tomography for our use case. In the results section, we show how the best optimised model works on our test dataset, comparing manual and automatic first arrivals and present the first tomographic results, including a comparison of manual and automatic approaches. Finally, we discuss the presented results and end with a short discussion and conclusion.

2 METHODOLOGY

2.1 The Dataset

During the HIPER2 project, a total of 43 Ocean-Bottom Seismometers (OBS) were deployed in Ecuador near the rupture area of the 2016 M_w 7.8 Pedernales earthquake (Galve 2022). The deployment began on land in January 2022 with the installation of 100 land stations and continued offshore in March 2022 aboard the French vessel R/V L'Atalante. The OBS network was arranged in a 2D rectangular grid with 10 km station spacing, covering the Ecuadorian forearc from the coastline to the trench between latitudes 0°10'S and 1°25'N (Fig. 1). The network recorded continuously for 25 days. For seismic data acquisition, a 4990 cu. in. airgun array was used to collect wide-angle

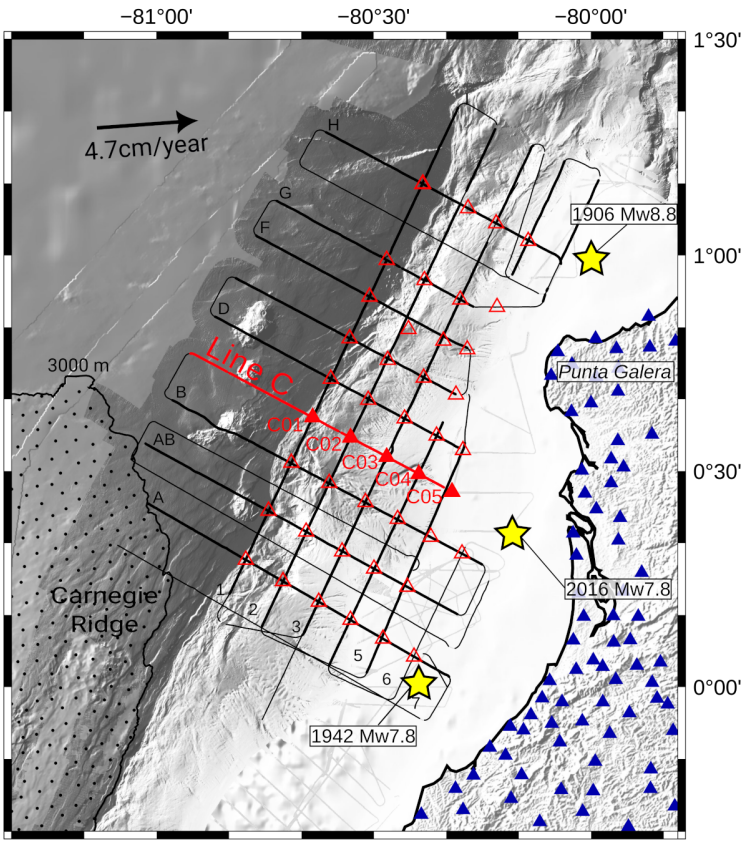

Figure 1. 3D grid acquisition of the HIPER2 marine campaign in Northern Ecuador and structural information from previous studies. Black lines correspond to the shots position for the wide-angle seismic (WAS) acquisition, recorded by the OBS (red triangles) and land stations (blue triangles). Bathymetry is from HIPER cruise (Galve 2020, 2022). Black dotted area is the Carnegie ridge limited by the 3000 m isobath. Yellow stars correspond to the main shock for the M_w 7.8 Pedernales earthquake in 2016 (Ye et al. 2016; Nocquet et al. 2017), the M_w 8.4-8.8 Colombia-Ecuador earthquake in 1906 (Kanamori & McNally 1982; Parra et al. 2016; Yoshimoto et al. 2017) and the M_w 7.8 1942 earthquake (Sennson & Beck 1996).

seismic (WAS) data, generating 14,000 shots spaced every 150 m and recording times of 60 s. OBS clocks were synchronised to absolute GPS time both before and after deployment to correct for clock drift, which typically averaged around 5 ms per day. Land stations relied directly on GPS timing throughout the deployment.

To create the training and test dataset for DeepFB, we used all available manually analysed hydrophone channels from the OBS sections (Delsuc et al. 2025). In total, we have 292 analysed sections with 229,065 traces and 106,960 manually labelled first-breaks (i.e., 122,105 traces with no first-break). The average signal-to-noise ratio (SNR) of all picks is 4.62 dB (Fig. 2a), and the distances between the shots and the receivers range from 0 km to 134 km (Fig. 2b). For the test dataset, all stations from line C (24 sections, Fig. 1) were excluded, as this line is reserved for evaluating the trained DeepFB models against manually labelled first-breaks and for comparing tomographic results from automatic versus manual picking (see section 3.3). Further, we randomly selected additional 34 sections for testing. In summary, we have 234 sections for training and 58 for testing. This results in 184,230 traces for training with 84,538 picks and 44,835 traces for

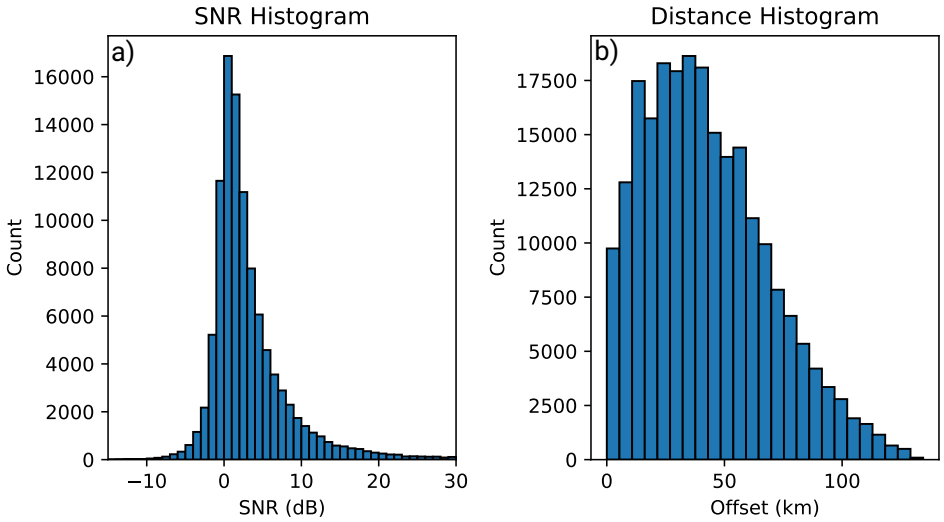

Figure 2. Histograms of signal-to-noise ratio (SNR) of all manually labelled picks and distances between shot and receivers (b) for the whole dataset.

testing with 22,422 manually labelled picks. For the training and neural network design, we created chunks of traces from each section. Each chunk contains 32 traces with at least 28 manually labelled picks, resulting in 1529 chunks. The number of traces per chunk is kept constant in order to keep the neural network relatively small (see section 2.2.2) and to obtain enough chunks with many manually labelled phase onsets. The data are bandpass filtered between 2.5 Hz and 16 Hz, since manually picking was done in this frequency range to suppress seismic noise. Most of the data were recorded with a sampling rate of 250 Hz and each trace has a length of 60 s, i.e. each trace has 15,000 samples. In order to reduce the number of trainable parameters we downsampled the 60 s data so that each trace only contains 2048 samples. This results in a sampling rate of 34.13 Hz. To create chunks of data, neighbouring windows overlap by 95%. Each trace has zero mean and is min-max-normalised. Furthermore, we set up a second training and test dataset where we reduced the traveltime of each trace by

$$t_{\text{red.}} = t - \frac{x}{v_{\text{red.}}}\,, \tag{1}$$

where $t_{\text{red.}}$ denotes the reduced traveltime, t is the original time, x is the distance between source and receiver (Fig. 2b) and $v_{\text{red.}}$ denotes the reduction velocity, which is 7000 m/s for this study. Figure 3 shows examples of a downsampled data chunk (Fig. 3a) and one with reduced traveltime (Fig. 3b), including the colour-coded semantic segmentation for times before and after the first-break (blueish area before and yellowish area after the first-break in Fig. 3). Usually, the arrivals lie on a curved line (moveout curve, Fig. 4a), however, when working with reduced traveltime data, the arrivals will lie on a horizontal line (Fig. 4b) which can improve the quality of data interpretation. Data from the reduced traveltime dataset were downsampled to a frequency of 75 Hz, even if the

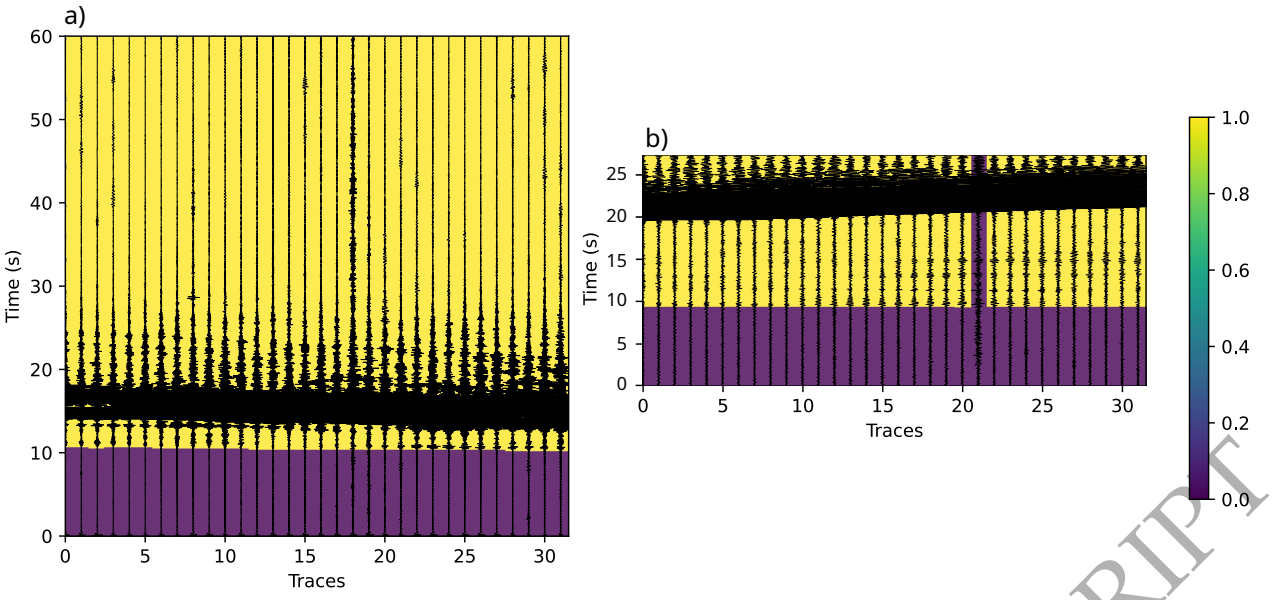

Figure 3. Input traces (only hydrophone channel from OBS) and output map for downsampled (a) and reduced data (b). Blue area indicates segment before first-break and yellow area is the segment after the first-break, i.e., the segments before the first-break are set to 0 and the segment after the first-break to 1 (represented by the colour bar). If manually picking of first arrival was not possible, e.g. due to too noisy conditions at trace 5 in (b), the trace is labelled with no first-break. The data are bandpass filtered between 2.5 Hz and 16 Hz.

data were also bandpass filtered. Each trace also has a length of 2048 samples to use the same neural network as for the other dataset (see section 2.2.2), i.e., each trace of the reduced dataset has a length of 27.31 s. Beside these described methods, we additionally created a noisy training dataset by adding Gaussian noise. Therefore, we computed the average signal-to-noise ratio (SNR) of each chunk by averaging over all SNRs of each trace in the chunk. The SNR of a single trace is computed by

$$\mathrm{SNR} = 20 \cdot \log_{10} \frac{S}{N}\,, \tag{2}$$

where S and N are the root-mean-square of the demeaned signal and noise window, respectively. The signal is determined from 1 s after the first-arrival and noise from a 1 s window before the first-arrival. If the SNR is above a threshold of 4 dB, we deteriorated the data by adding Gaussian noise. We selected the threshold of 4 dB since the mean of the SNR is 4.6 dB (Fig. 2). The standard deviation of the Gaussian noise was controlled by a random scale factor uniformly sampled from the interval $[0, 0.4]$. We end up with additional 362 noisy chunks. In summary, we have two different training datasets, where one contains downsampled data for reduced traveltime and a second one contains downsampled data without traveltime reduction. Both datasets have a total size of 1891 chunks with 32 traces and are bandpass filtered between 2.5 Hz to 16 Hz, with at least 28 picks in each chunk of data.

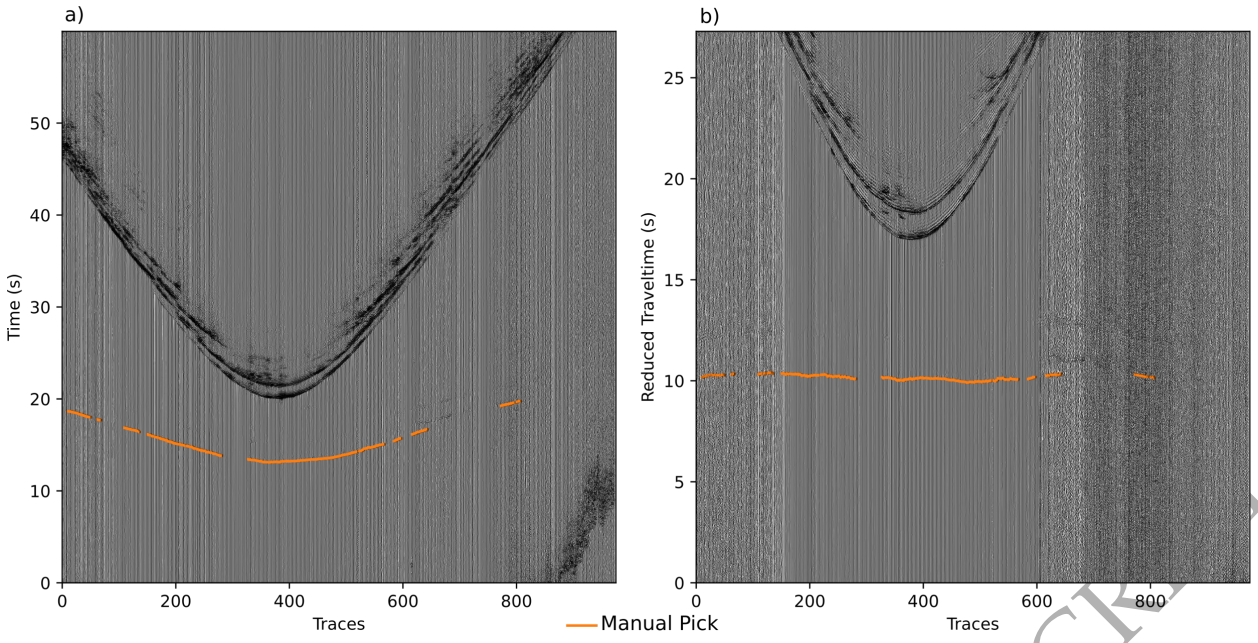

Figure 4. Plot of the recorded (a) and reduced (b) section C03_1 and manually picked arrivals (orange line). Data in (a) are downsampled and in (b) reduced traveltime is applied. The data are bandpass filtered between 2.5 Hz and 16 Hz.

2.2 Deep-Learning for Seismic Phase Picking

2.2.1 PhaseNet

Over the last century, deep-learning approaches have shown that they outperform classical methods in seismic data processing (Mousavi & Beroza 2022). Zhu & Beroza (2018) developed a deep-learning phase picking algorithm to pick seismic phase onsets at single stations. As shown by Münchmeyer et al. (2022), PhaseNet is one of the best-performing deep-learning phase picking approaches in seismology. It is a modified U-Net (Ronneberger et al. 2015), consisting of a down-sampling branch using 1D convolutions and an up-sampling branch employing 1D deconvolutions. The input of the deep neural network takes three-component seismograms with a length of 3001 samples, so the input layer has a dimension of 3×3001 samples. PhaseNet returns three probability distributions with the same shape as the output, where the distributions represent the probability for a P-wave, S-wave and noise (i.e. neither P nor S). The probability distribution is a Gaussian window around the manual label that is one at the label. Here, we used the implementation of PhaseNet in SeisBench (Woollam et al. 2022) but only used the hydrophone channel of the OBS. The two remaining channels were set to zero. In addition, PhaseNet was only trained on P-waves, i.e., it only predicts P-phase onsets. To train PhaseNet, we took all manually labelled first-breaks from our dataset and trained PhaseNet for 100 epochs with a constant learning rate of 0.001, a batch size of 512 and the original sampling rate of 250 Hz. The standard deviation of the Gaussian

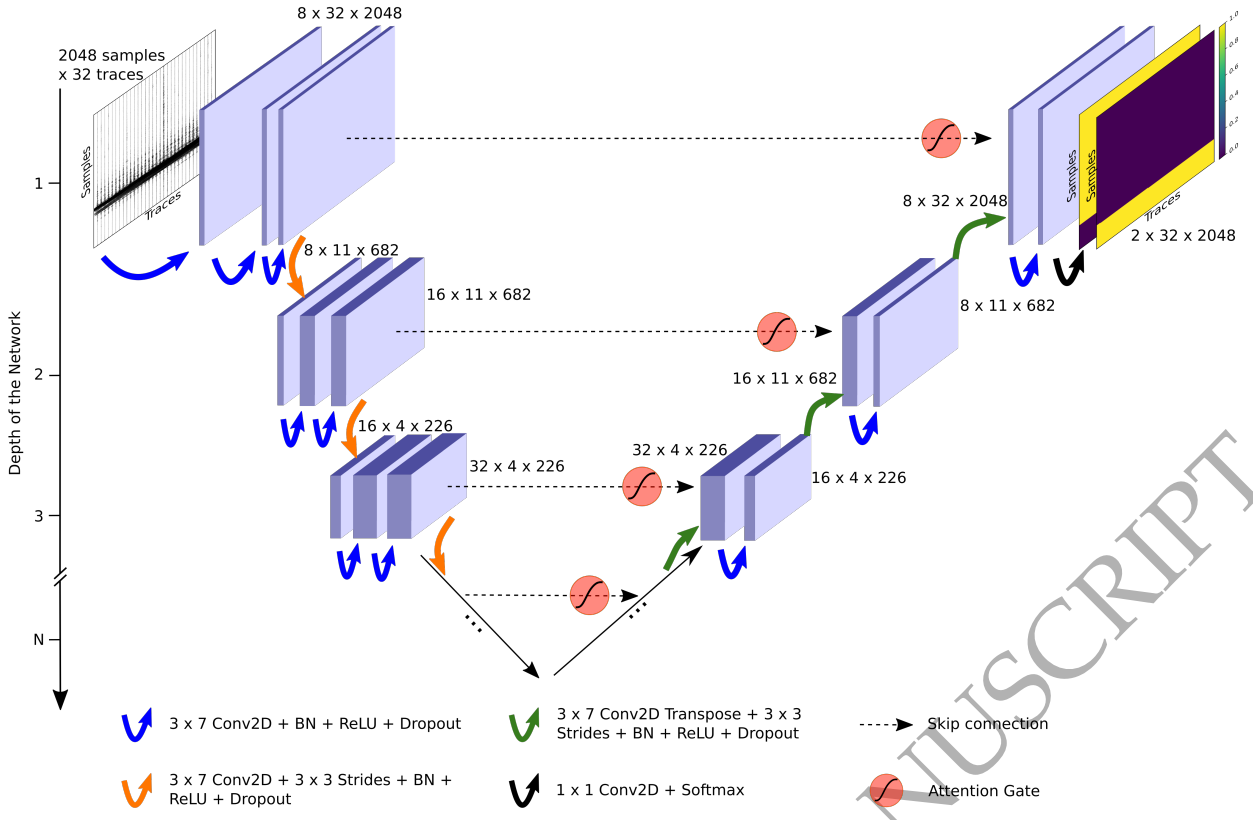

Figure 5. General neural network architecture of DeepFB. The input layer consists of 32 traces which 2048 samples for each trace. The output are two mask functions with the same shape as the input layer, with a boundary between the time before and after the first-break. Kernel size and strides are kept constant for the downsampling as well as for the upsampling path, however, depth, filter root (here 8), and usage of skip connections and attention gates is optimised using the evolutionary algorithm Propulate.

distribution around a manual labelled pick was set to 20 samples (i.e. 0.08 s at a sampling rate of 250 Hz). The training datasets was split into 80% for training and 20% for validation. Testing the dataset on unseen data from line C (Fig. 1) was done after finishing the training.

2.2.2 *DeepFB*

Since PhaseNet (Zhu & Beroza 2018) only uses data from single stations and also picks the phase onsets on single stations, it is not able to learn any spatial relations from neighbouring stations, for example a moveout in phase onsets. Existing deep neural networks, such as the one presented by Mardan et al. (2024), take all available traces from a seismic section and resample these images to a certain size (512×256 samples). However, the number of traces for different shots might change and thus, resampling these shots to a certain input size for the neural network ends up in a loss of information. Therefore, we developed a new deep neural network that takes a certain number of seismic traces as input and outputs the first-breaks for these traces (Fig. 5). Our neural network is called DeepFB (Deep Neural Network for First-Breaks). The neural network is also a modified

U-Net (Ronneberger et al. 2015), that takes as input 32 traces where each trace has a length of 2048 samples (i.e. shape of the input is 2048×32, Fig. 3). In the first step, a 2D convolution with a kernel size of 3×7 is applied on the input data, which is repeated two times in the first layer. Each convolution is followed by batch normalisation (Ioffe & Szegedy 2015), ReLU as the activation function, and dropout. In each layer of DeepFB, this 2D convolution is applied on the resulting feature maps. As shown in Figure 5, the number of filters (i.e. filter root) is initiated with 8 and increases with the depth by $2^{\text{depth}} \cdot$ filter root. To decrease the size of the feature maps, also 2D convolutions are applied with the same kernel size and a stride of size 3×3. The upsampling branch uses 2D transpose convolutions with a kernel size of 3×7 and a stride of 3×3, followed by batch normalisation, ReLU and dropout, and by a 2D convolution with the same dimensions as in the downsampling branch. To create the output maps which represent samples before and after the first-break, a 2D convolution with a kernel size of 1×1 is applied on the last layer and are passed through a softmax activation function to produce the output mask functions. Skip connections at each depth concatenate layers from the left to the right side in order to improve the convergence during the training process (Li et al. 2017). Additionally, attention gates (Oktay et al. 2018) can be added to the skip connections to focus on certain areas. To find the best working model on our test dataset, we optimised the hyperparameters using the evolutionary algorithm Propulate (Taubert et al. 2023).

2.3 Optimisation of DeepFB

Propulate (Taubert et al. 2023) implements a massively parallel, asynchronous optimisation algorithm tailored for global optimisation tasks, especially hyperparameter tuning and neural architecture search in machine learning. It builds on the concept of evolutionary algorithms (EAs), a class of optimisation techniques inspired by natural selection and biological evolution. These algorithms operate by maintaining a population of candidate solutions to an optimisation problem, which are iteratively refined through biologically inspired modifications such as selection, mutation, and crossover. Over successive generations, the population evolves toward better solutions by favouring individuals with higher performance, as measured by an objective or fitness function. Evolutionary algorithms are especially effective for complex and non-convex optimisation problems without explicit closed-form definition. Propulate enhances classical parallel evolutionary algorithms by introducing an asynchronous population update scheme, making it particularly well-suited for large-scale parallel and high-performance computing environments. Unlike conventional EAs that

Table 1. Search space to find best working DeepFB model using Propulate. Different loss functions: MSE: Mean squared error loss; BCE: Binary cross entropy loss; Dice loss; (Huang et al. 2018); Focal loss (Lin et al. 2017). The value std threshold is only used in the prediction mode. More details about the std threshold is given in section 2.4.

Hyperparameter	Search space
Depth	$4, \ldots, 8$
Filter root	$2, 4, 8, 16$
Drop rate	$0, \ldots, 0.5$
Batch size	$16, 32, 64, 128, 256$
Learning rate	$0.1, 0.01, 0.001$
Loss function	MSE, BCE, Dice, Focal
Skip connections	True, False
Attention gate	True, False
Reduced traveltime	True, False
Noise dataset	True, False
std threshold	$0, \ldots, 100$ samples

require fixed synchronisation points between generations, causing inefficiencies when objective function evaluation times vary strongly, Propulate enables continuous evolution without waiting for all individuals to be evaluated. This highly scalable design cannot only deliver results up to three orders of magnitude faster than other hyperparameter optimisation frameworks (Taubert et al. 2023), but may also yield better hyperparameters or neural architectures in fewer evolutionary steps due to shorter information pathways. Here, we optimised DeepFB on the hyperparameters presented in Table 1. We used the true positive rate (TPR) of the trained model as our objective function to find the best-performing model. Each DeepFB model was trained for a maximum of 500 epochs with a constant learning rate using the Adam optimizer (Kingma & Ba 2014). The weights of each neural network were initialised using the Kaiming initialisation (He et al. 2015). Different loss functions (Tab. 1) were tested to handle class imbalances, since some traces in each chunk do not contain manually labelled phase onsets (i.e., each chunk contains at least 28 manually labelled phase onsets). The training was stopped when the validation loss did not decrease further over 20 consecutive epochs (i.e. early stopping). After finishing the training, the model was tested on our test dataset from line C, and we evaluated the model on its true positive pick rate (TPR). Additionally, model performance was assessed using the residual times on the test dataset, as well as the root-mean-square (RMS) deviation between manually picked and DeepFB-predicted arrival times. A pick was counted as a positive pick when the residual of the predicted pick was below a certain threshold with respect to the manual pick. We optimised DeepFB on two different residuals (0.1 s and 0.05 s).

2.4 Detecting first-breaks

Since DeepFB is only trained on chunks of 32 seismic traces, we developed an algorithm to predict the first-breaks on a single section (Fig. 6). Therefore, we load the data of a whole section

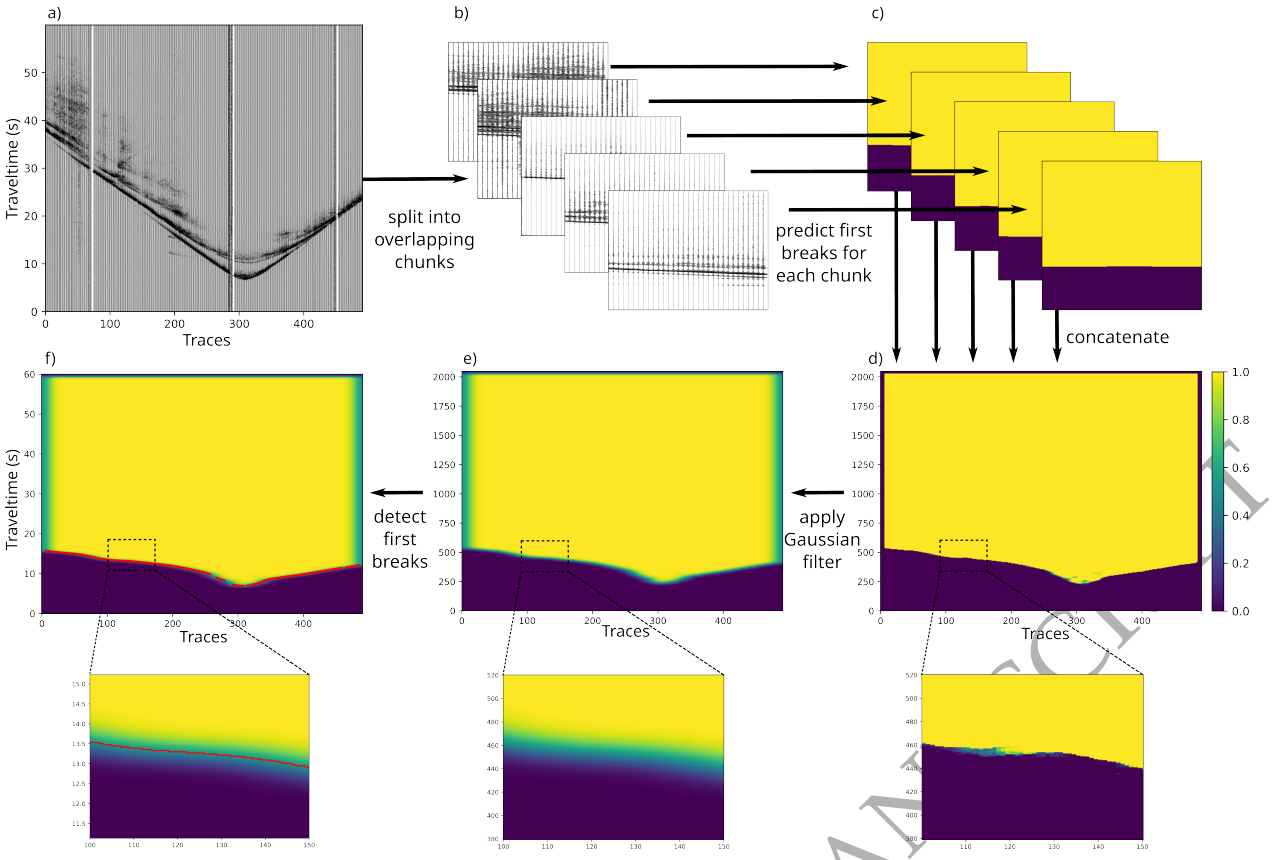

Figure 6. Workflow how to detect first-breaks from a seismic section: First, the whole section (a) is divided into overlapping chunks, where each chunk has a size of 32 traces and 2048 samples (b). Second, the previously trained DeepFB model is applied to each chunk of data (c). Afterwards, the predicted output is concatenated to obtain a whole map containing the first-breaks for the whole section (d). Since each single seismic trace can have several predictions due to the overlapping, all predicted data are merged together either by taking the average or the maximum value from all predictions. In the next step, the predicted output for the whole section is convolved with a 2D Gaussian filter to remove jumps and smoothing first-breaks (e). In the last step, the first-break is detected for each trace by finding a certain threshold value for the predicted output map (f). Due to the overlap, we can calculate the first-breaks for each trace several times during the prediction. With this approach, it is possible to calculate the standard deviation of all predicted first-breaks for each trace and thus remove picks if the standard deviation exceeds a certain threshold (std threshold in Tab. 1).

(i.e. loading all traces of the section), and resampled the section that each trace has a length of 2048 samples or reduced traveltime as presented in eq. 1. Afterwards, we split the section into overlapping chunks, where each chunk contains the required number of 32 traces. The model is then applied to each chunk of data and the predicted output maps are concatenated to form the first-break map for the whole section. For each chunk, we set the prediction of the first and last six traces to 'nan' (not a number) to avoid any effects from the edges. Since the chunks overlap, we must take the overlap into account when merging the predictions. Therefore, we compute either the average or the maximum from all overlapping segments of a single trace. In a second post-processing step, we apply a 2D Gaussian filter on the first-break map of the whole section in order to remove jumps and to smooth the first-breaks. The 2D Gaussian filter is defined as

$$G(x, y) = \frac{1}{2\pi\sigma^2} \cdot \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) , \tag{3}$$

where σ denotes the standard deviation of the filter and x and y are the grid coordinates. The filter $G(x, y)$ is then convolved (i.e., 2D convolution) with the predicted images. In this study, we set $\sigma = 10$ samples. Comparisons of how σ affects the first-breaks are presented in Fig. A1. In the last step, the first-breaks are detected by searching for a certain threshold value in each trace. Here, we used 0.65 as a threshold value, i.e., when this value is exceeded, the first-break is detected. Both values are selected in such a way as to maximise the number of correctly labelled first-breaks by our proposed algorithm. We also implemented a method to remove unreliable picks, for example on dead or very noisy traces: Each trace receives multiple first-break predictions because of the overlapping input windows. This allows us to compute the standard deviation of the predicted arrival times from all overlapping traces and to discard picks whose standard deviation exceeds a given threshold. In this way, traces can be automatically excluded. In this study, we used an overlap of 95%, resulting in up to 16 independent predictions per trace. Experiments with smaller overlap fractions (30%, 40%, 50%, 60%, 70%, 80%, 90%) result in lower true positive rates for our test dataset (77.7%, 88.7%, 89.3%, 88.8%, 89%, 89.5%, 89.6%). Furthermore, the use of larger overlap fractions did not lead to a significant increase in computing time. In order to test whether the proposed standard deviation thresholding method is necessary, we added the standard deviation threshold as a hyperparameter to the optimisation process (Tab. 1).

2.5 Bayesian approach of Tomo2D

After detecting the first arrivals through manual or automatic picking, we used the Bayesian approach of Tomo2D (Korenaga & Sager 2012) to obtain our 2D P-wave velocity models of line C. This approach is an evolution of the joint refraction and reflection tomography method from Korenaga et al. (2000). For a complete description of this method, please refer to Korenaga & Sager (2012). We used a velocity model mesh with a horizontal node spacing of 1 km and a vertical node spacing of 0.2 km in the upper 2 km, and 0.5 km from 2 to 40 km. For the effective parameters we chose the same values as for the dip lines A and E presented in Delsuc et al. (2025).

3 RESULTS

3.1 PhaseNet

To pick the first-breaks, we started to train PhaseNet (Zhu & Beroza 2018) with our dataset. To predict the phase onsets, the SeisBench (Woollam et al. 2022) implementation of PhaseNet takes as input the data from a single station and outputs a probability distribution for the phase onsets. If this

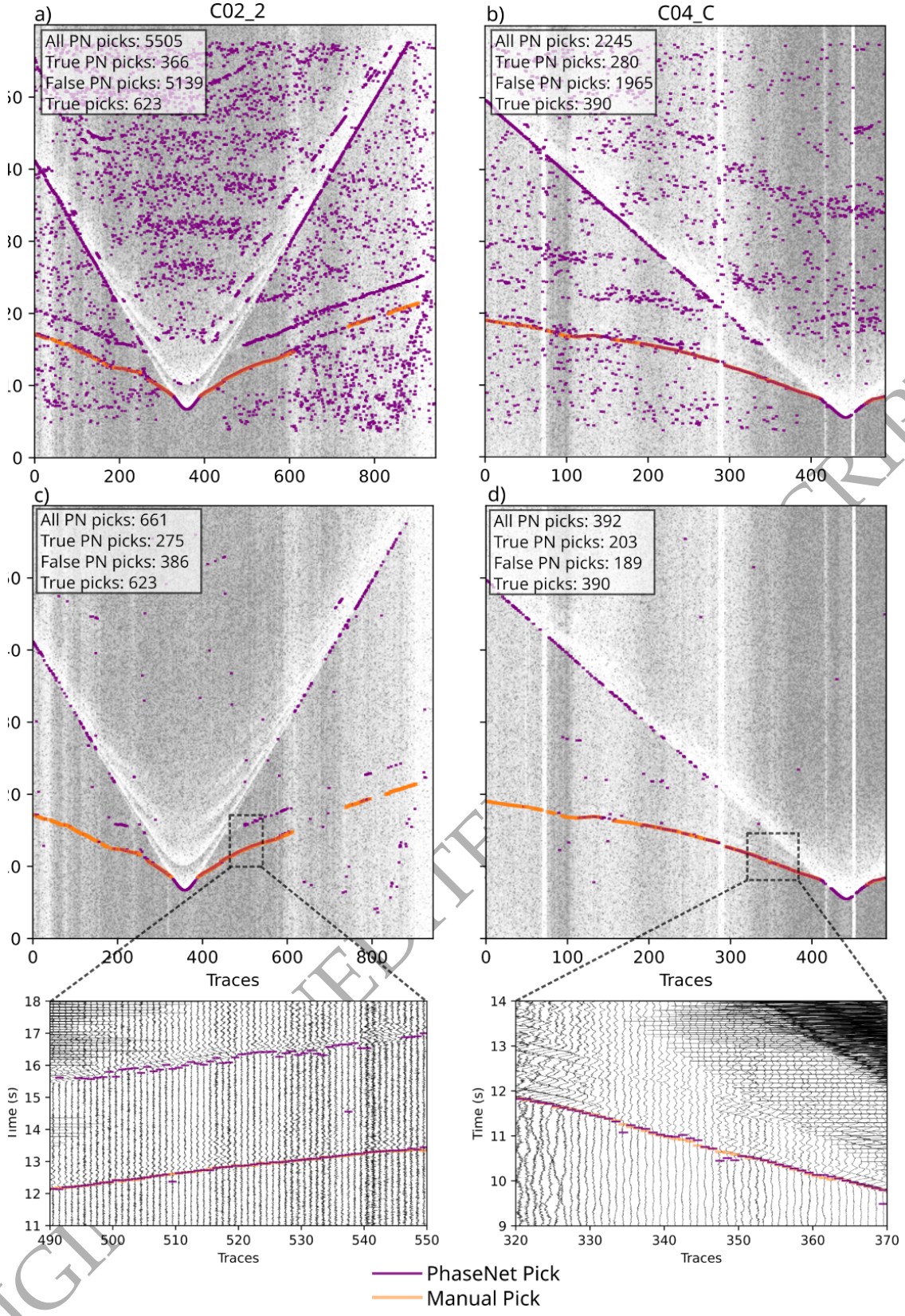

Figure 7. Applying the trained PhaseNet model to two sections from line C. Purple dots are the detected first-breaks using the trained PhaseNet model and orange dots represent the manual picks. A first-break was picked when the output probability of PhaseNet exceeds a threshold of 0.2 (a, b) or 0.8 (c, d). Numbers in the boxes represent the numbers of detected picks in comparison to the numbers of detected true and false picks using PhaseNet (PN), and the number of manually labelled picks (true picks) of each section.

Table 2. Hyperparameters for best performing models tested with residuals of 0.1 s and 0.05 s. Additionally, two models were tested and optimised on the standard deviation threshold in samples (std_threshold), otherwise std_threshold was not used (None) during the prediction phase. The last line shows the true positive rate (TPR) of each model, tested on our test dataset from line C.

residual (s)	0.05	0.1	0.05	0.1
Depth	5	6	5	5
Filter root	8	8	8	4
Drop rate	0.432	0.267	0.451	0.182
Skip connections	True	True	True	True
Attention gate	True	False	False	False
Noise dataset	False	False	False	True
Reduced traveltime	True	True	True	True
Batch size	32	32	64	128
Learning rate	0.001	0.001	0.001	0.001
Loss function	Focal	Focal	MSE	Focal
std_threshold	96	None	None	96
TPR	89.1%	89.8%	89.6%	89.6%

distribution exceeded our defined threshold value of 0.2 (Fig. 7a, b) or 0.8 (Fig. 7c, d), a phase onset was found. Figure 7 shows the detected phases from two different sections and for comparison the manually labelled first-breaks (orange line in Fig. 7). PhaseNet detects 59% of manually labelled picks in Figure 7a and 72% in Figure 7b, however, it also picks different phases or other signals (5139 and 1965). Also for a higher detection threshold of 0.8 (Fig. 7c, d), the trained PhaseNet model is not able to correctly pick the first-breaks. Thus, PhaseNet has a high false positive rate for low thresholds and misses many manually labelled first-breaks for a high threshold. In both cases, manual verification of the automatically detected picks would be necessary to remove the incorrect picks or select the correct first-breaks, which is very time-consuming and therefore, not faster than manual picking.

3.2 Finding best DeepFB model

To find the best working DeepFB model, Propulate (Taubert et al. 2023) took as input the range of hyperparameters given in Table 1. Each model has been trained on a different set of hyperparameters and tested on our test datasets from line C. We used two different residuals (0.1 s and 0.05 s) for testing each model during the optimisation process and evaluated each model on its true positive rate (TPR, Fig. 8, Tab. 2). The higher TPR, the better performed the model on our test dataset. In total, we trained 240 models for each residual in parallel, which took around 20–30 hours for each run on two Nvidia A100 GPUs with 40 GB memory. Figure 8 shows the residuals of the four best-performing models for a residual of 0.1 s. The different models show only minor differences in TPR, and the best-performing model does not require the standard deviation threshold to discard first-breaks. Figure 9 shows two examples of our best performing model (Tab. 2) on two sections from line C. Orange lines in Figure 9a & b represent the manual picked first-breaks and red lines

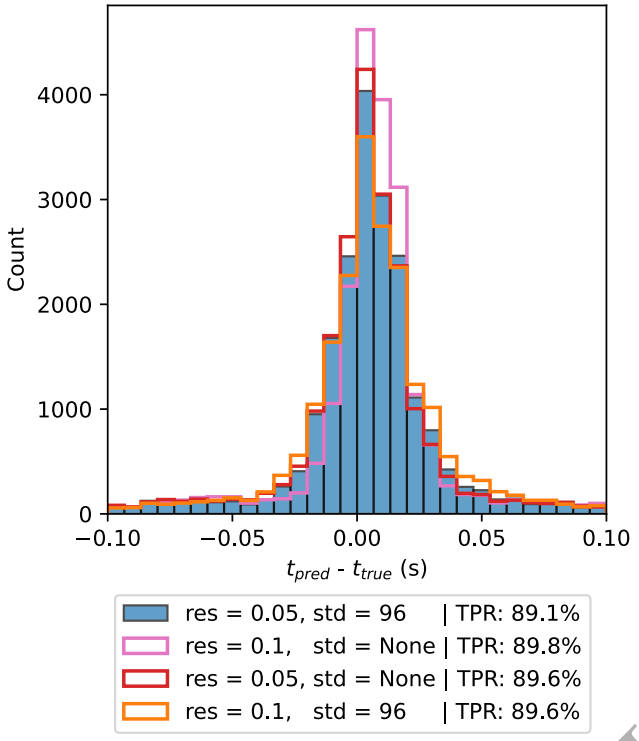

Figure 8. Histogram of four different optimised DeepFB models tested on our test dataset from line C with a maximum uncertainty of 0.1 s. During the optimisation process, the models were optimised for the true positive rate (TPR) and for two different maximum permissible residuals (res = 0.05 s and res = 0.1 s). Additionally, two models were optimised on the standard deviation threshold (std) to test if there is a best threshold value to remove predictions with high uncertainties and to automatically clean the predicted first-breaks.

are the first-breaks detected by DeepFB. Figure 9c & d shows the histogram of the pick residual and TPR of this model for a residual of 0.1 s.

3.3 Comparison between manual and DeepFB picks

To assess the efficiency of DeepFB on line C we have picked manually the first arrival on every OBS of this line. This results in 1921 picks out of 2455 available traces (78% of possible picks, blue picks in Fig. 10). For manual picking, three different filters were used to maximise the SNR: A bandpass filter between 0.05 Hz and 80 Hz to erase very high and very low frequency noise and two more restrictive bandpass filters between 2.5 Hz and 16 Hz, and between 2.5 Hz and 8 Hz. Depending on the used filter and the precision of the picking, different uncertainties were assigned varying from 50 ms to 150 ms. For the comparison between manual and automatic picks of DeepFB, we applied the best model from the optimisation process (DeepFB model with highest TPR in Tab. 2) on line C.

DeepFB generates a pick for every trace in the OBS section. However, in some cases, first arrivals are not visible and can be masked by the water wave or by excessive noise (Fig. 11a for traces between 250-270 and Fig. 11a, b, c, d for traces between 285-295, respectively). In areas

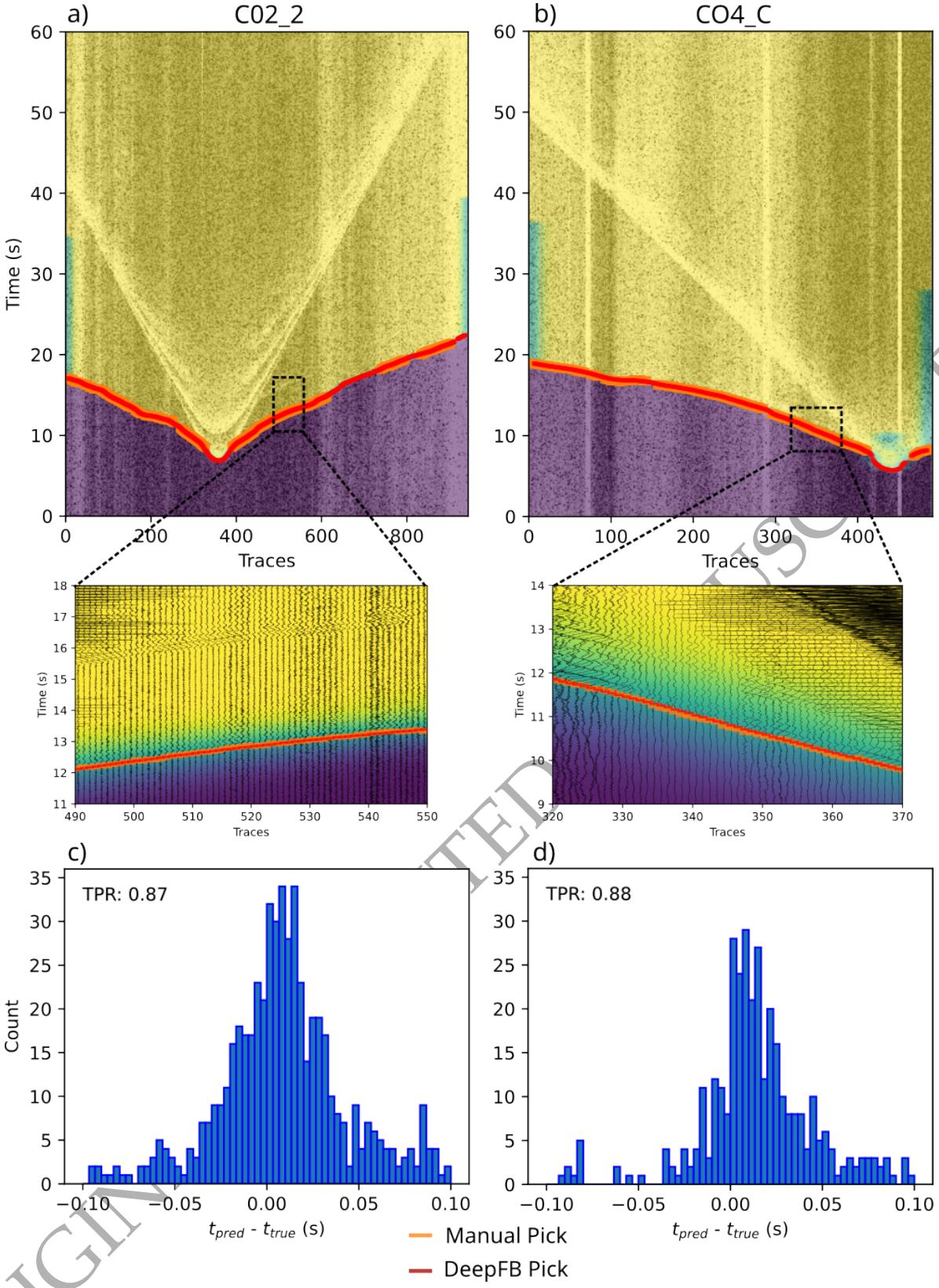

Figure 9. Our best performing model (residual 0.1 s and no standard deviation threshold, Tab. 2) applied and tested on two sections from line C (a, b) and histograms for pick residuals (c, d) including the true positive rate (TPR). To avoid any edge effect of the neural network, we set the first and last six traces of each chunk to 0, which causes the blueish areas at the edge of both sections (a, b). All other traces have several predictions, due to the overlapping of 95%.

with low SNR, such as between traces 320-340 in Figure 11d, DeepFB misidentifies the arrivals, producing picks that deviate significantly from the expected position (blue picks in Fig. 11). Such inconsistencies are easily detected by inspecting the OBS cross-section and are subsequently removed through manual cleaning, producing gaps in the pick distribution, visible in Figure 12. A second challenge was identified at the position of an Atacames seamount seaward of OBSs (Fig. 1) causing a gap in picks distribution on every OBS section (respectively at -20 km, -33 km, -40 km, -50 km, and -60 km from corresponding OBS, Fig. 12). Additionally, Figure 12c highlights a larger difference between manual and DeepFB picks around this area with values > 50 ms. The discrepancy between manual and DeepFB picks in this area are likely coming from the impact of seamount's bathymetry leading to strong changes of apparent velocities on the seismic section. Due to these changes, the spatial coherence advantage of DeepFB does not apply here and leads to a difference of 0.05 to 0.15 s between manual and DeepFB picks. After the cleaning process, we retained 1,722 reliable picks, representing about 70% of the possible arrivals (red picks in Fig. 10). The root-mean-square of the difference between manual and DeepFB picks is 21 ms which is less than half of our smallest picking uncertainty (50 ms).

3.4 Impact on tomographic velocity models from manual and DeepFB picks

To validate the DeepFB first-break picks using Bayesian Tomo2D inversion, it was first necessary to assign an uncertainty to each pick. Uncertainties of 50 ms, 70 ms, and 150 ms were assigned based on the distance from the OBS. Each pick from the training dataset was grouped into 1 km offset bins ranging from 0 km to 120 km, and the percentage of picks associated with each uncertainty value was calculated within each bin. For each bin, the smallest uncertainty that applied to more than 40% of the picks was selected. Using these estimated uncertainties, Bayesian Tomo2D inversion was applied to the DeepFB picks, and the resulting velocity model was compared with the model obtained from manual picks. This procedure resulted in an uncertainty of 50 ms for offsets between 0 km and 9 km from the OBS, 70 ms for offsets between 9 km and 45 km, and 150 ms for offsets greater than 45 km. While assigning uncertainties by hand would be preferable, this approach offers an automatic and data-driven alternative.

The velocity models obtained from manual and DeepFB picks are presented in Figures 13a and 13b, with their difference shown in Figure 13c. Uncertainties for the two velocity models are estimated using the standard deviation between the different inversions performed with the Bayesian approach of Tomo2D (Korenaga & Sager 2012) and are presented in Figure C1. The

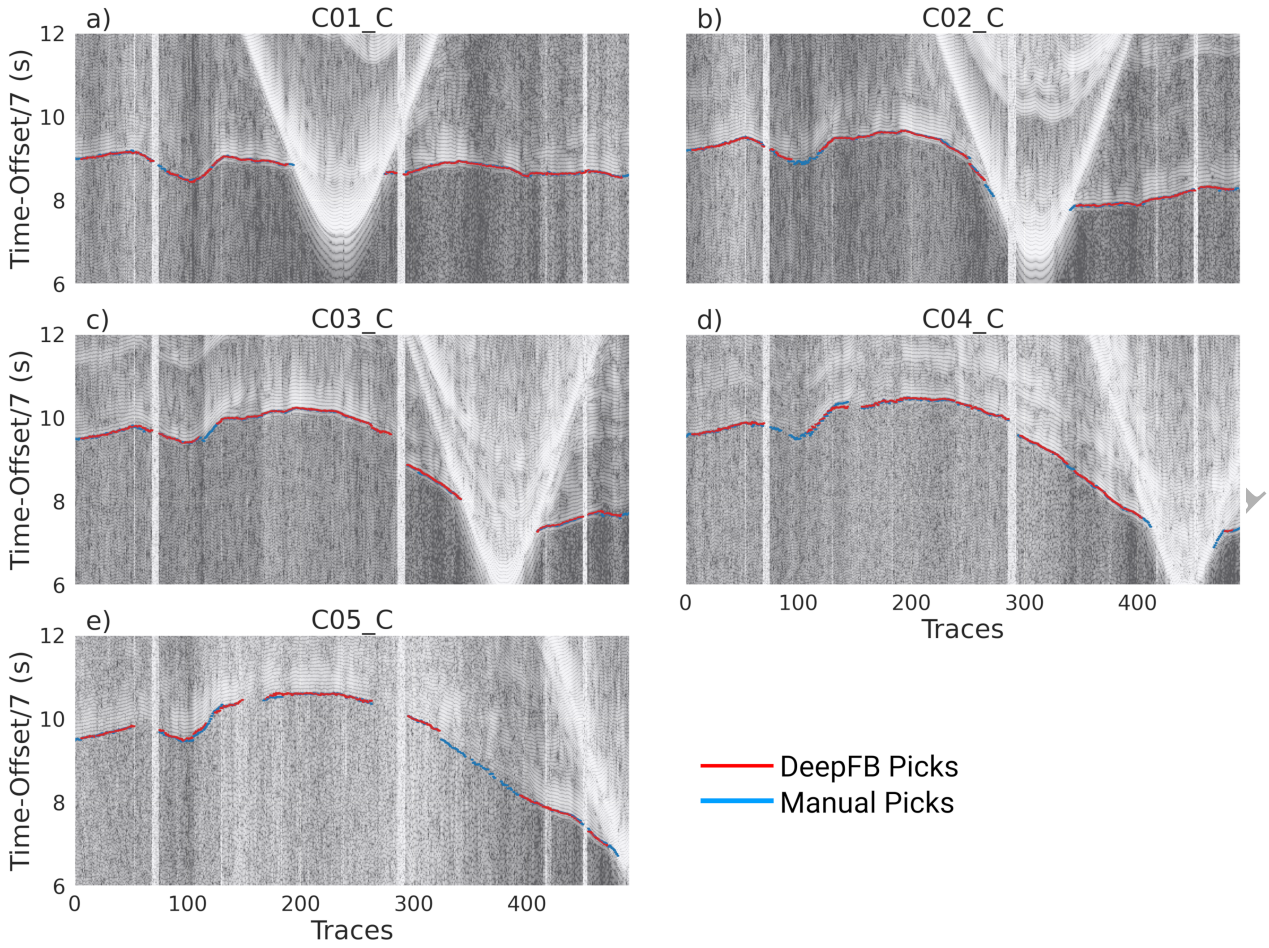

Figure 10. OBS sections from Line C. In blue, manual picks and in red, picks from DeepFB manually cleaned. A bandpass filter between 2.5 Hz and 8 Hz and a traveltime reduction of 7 km/s (see section 2.1) have been applied on these sections.

spatial resolution of the tomography model along profile C is shown using checkerboard tests presented in Figure B1. Overall, the two models are highly consistent, with discrepancies of less than 0.2 km/s across most of the section. The main exception occurs beneath the seamount, around −17 km from the trench, where the difference between the two models is reaching −0.4 km/s. This localised misfit is consistent with the lack of DeepFB picks near the seamount and the arrival time differences between DeepFB and manual picks (Fig. 12). It also corresponds to an area with uncertainties > 0.4 km/s (Figure C1). In the following subsection, we describe the differences between the two velocity models along the main velocity structures of the Ecuadorian margin. Velocities obtained further than 5 km seaward of the trench are not discussed in this article because of the lack of resolution in this area due to the absence of OBS in the oceanic domain (Figure B1). We therefore choose to focus on velocity structure near the trench and in the forearc domain. 'Manual model' will refer to the velocity model obtained through the inversion of the manual picks

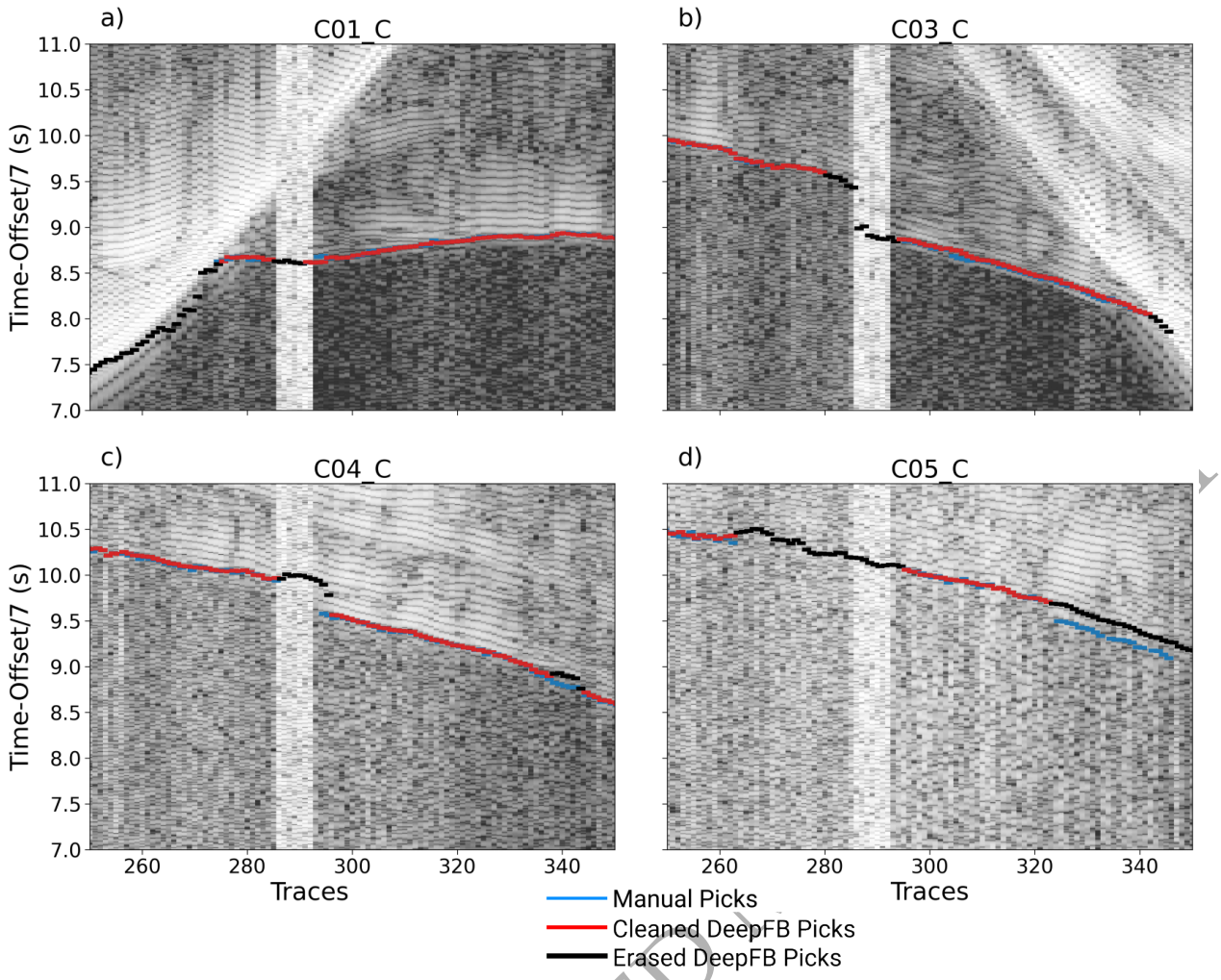

Figure 11. OBS sections from Line C. In blue, manual picks, in red, picks from DeepFB manually cleaned and in black, DeepFB picks erased during the manual cleaning. A band pass filter between 2.5 Hz and 16 Hz and a traveltime reduction of 7 km/s (see section 2.1) have been applied on these sections.

(Fig. 13a) and 'DeepFB model' will refer to the velocity model obtained through the inversion of the picks coming from DeepFB (Fig. 13b).

3.4.1 Oceanic domain

A comparison of the 1D v_P models in this region is shown in Figure 13c with the top of the subducting Nazca plate provided by Font et al. (2013) (dashed black line in Figure 13). Both DeepFB and Manual models yield a velocity of 4.1 ± 0.1 km/s at the oceanic basement (5.1 km depth) 5 km NE from the trench. At greater depths, velocities increase similarly in both models, despite the larger associated uncertainties, from ~ 5.6 ± 0.3 km/s at 7 km depth to ~ 6.5 ± 0.3 km/s at 10 km depth.

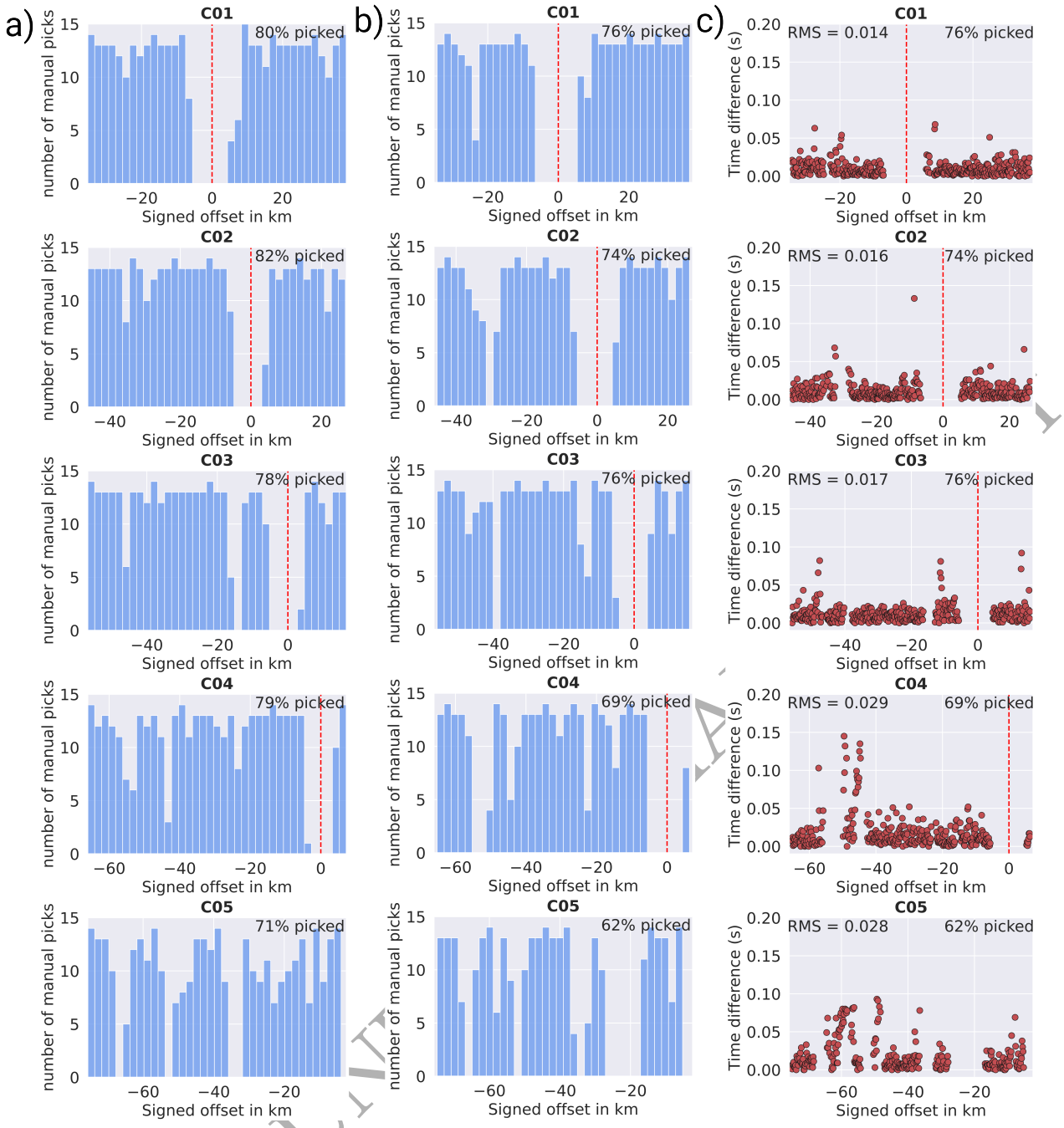

Figure 12. a) Manual picks distribution, b) DeepFB picks distribution. c) Difference between manual picks and DeepFB picks. Red vertical dashed line corresponds to the position of the OBS.

3.4.2 Forearc domain

In the forearc domain, Gailler et al. (2007) used the 4 km/s iso-velocity contour to track the basement depth. This contour will then be used in this section to mark the forearc basement. From both models (Figure 13a and b), we extracted three 1D v_P models at 11 km, 23 km, and 36 km from the trench. DeepFB and manual 1D v_P models at 11 km and 36 km yield the same forearc basement depth of $\sim$ 2.6 km below the seafloor, with uncertainties around 0.1 km/s (Fig. 13c). However, at

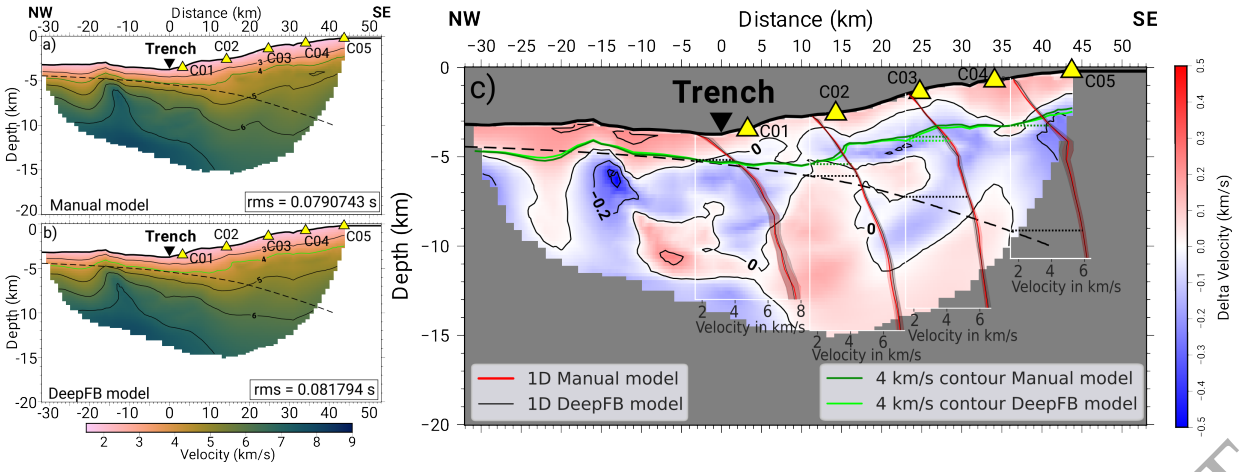

Figure 13. Velocity model from the manual picks (a) and DeepFB picks (b), and the difference of both models (c). Dashed black line corresponds to the position of the top of the subducting crust from Font et al. (2013).Dark green line corresponds to the 4 km/s contour from the manual model (a). Dotted dark green line corresponds to its projection on the 1D velocity model. Light green line corresponds to the 4 km/s contour from the DeepFB picking velocity model (b). Dotted dark green line corresponds to its projection on the 1D velocity model.

23 km from the trench, the two 1D v_P models differ by nearly 0.2 km/s at 4.3 km depth, resulting in a 0.25 km shift of the forearc basement depth between DeepFB and manual models. This offset remains smaller than the vertical grid spacing in this region (0.5 km below 2 km depth below the seafloor), indicating that the forearc basement topography does not show significant differences between the two velocity models.

We use the interplate topography from Font et al. (2013) to compare velocities in this area in both DeepFB and manual models. Along the interplate zone, v_P increases landwards as the forearc crust thickens with similar values for both models ranging from 4.4 km/s at 11 km from the trench to 5.2 km/s at 23 km, and 6.0 km/s at 36 km. Estimated uncertainties are also similar, with very low values around the interplate zone with ±0.1 km/s at 11 km and 23 km from the trench, and approximately ±0.15 km/s at 36 km from the trench. These results demonstrate the ability of the DeepFB automatic picking to recover the main velocity structures of the Ecuadorian margin, despite minor differences with the manual picks.

4 DISCUSSION AND CONCLUSION

First-break picking remains one of the most fundamental and time-consuming steps in the processing of active-source seismic data. Manual picking provides high-quality results but is a time intensive task, while existing automatic methods such as PhaseNet (Zhu & Beroza 2018) or classical STA/LTA approaches (e.g. Coppens 1985) have notable limitations. PhaseNet, for example, operates only on single-station data and cannot exploit the spatial coherence across neighbouring

traces (Fig. 7). As a result, it struggles to resolve moveout patterns and often produces either excessive false picks at low thresholds or significant numbers of missed arrivals at high thresholds (Fig. 7). While adjustments to the training procedure, such as larger training datasets, different loss functions, or additional regularisation, could marginally improve its performance, these measures cannot fundamentally overcome this limitation. The core issue lies in the model architecture, which processes each trace independently and therefore lacks access to the spatial information required to identify coherent moveout patterns across a seismic gather. Other recent neural-network-based methods that process entire seismic sections (e.g. Mardan et al. 2024; Wen & Ma 2025) typically downsample input data to reduce neural network model size. This compression inevitably sacrifices temporal resolution, which is critical for accurate first-break detection.

In this study, we introduced DeepFB, a U-Net-based model designed to address these shortcomings. By operating on overlapping chunks of 32 traces, DeepFB balances resolution with tractable model complexity. Overlapping predictions provide multiple independent estimates per trace, which can be used for uncertainty quantification via the standard deviation of multiple predictions. Combined with Gaussian smoothing (Fig. 6), this approach can remove spurious picks while preserving sharp first-breaks. The smoothing parameter and detection threshold were empirically chosen, but could be optimised in future work via validation-based tuning. The strategy of training with reduced traveltime data further enhanced model stability, as arrivals align along near-horizontal lines (Fig. 4), making learning more efficient. Attention gates and noise augmentation during training offer additional means to improve DeepFB generalisation, particularly in low-SNR OBS recordings where conventional pickers often fail.

A key aspect of our work is the automated hyperparameter optimization using the Propulate evolutionary framework (Taubert et al. 2023). This eliminated the need for manual trial-and-error tuning and resulted in models with higher true positive rates than hand-designed configurations. Although kernel sizes and strides were kept fixed in this first study, future extensions could include optimisation of these parameters. Further, we only optimised DeepFB on a single seed which controls the initialisation of random numbers. Using different seeds in future studies could also lead to small improvements in model performance and is a common technique to verify that a deep-learning model has not merely learned a favourable configuration of weights due to a particular random initialisation. Training multiple models with different random seeds and reporting the mean and variance of their performance would provide a more robust estimate of the model's generalisation capability. Future studies could also use additional augmentation techniques to increase the dataset size, for example by shifting a whole chunk in time. In addition,

parameters, such as the standard deviation σ of the Gaussian filter or the detection threshold are selected and optimised manually throughout this work. Here, automatic selection and fine-tuning of these parameters might be necessary to make automatic first-break picking with DeepFB more robust. Hyperparameter optimisation of DeepFB using our training and testing dataset from the HIPER2 experiment at the coastline of Ecuador has shown that the best-performing model does not require attention gates, noise augmentation and our proposed strategy of standard deviation thresholding to discard uncertain picks (Tab. 2). The best evaluated model was trained with the Focal loss function (Tab. 2) due to the different segmentation sizes of the areas before and after the first-break (Mardan et al. 2024). The evaluation against manual picks on the HIPER2 dataset (Fig. 13) demonstrates that DeepFB achieves residuals lower than the smallest uncertainty assigned during human picking (Fig. 9c and d). Tomographic inversions based on automatic and manual picks yield very similar velocity models, with differences below 0.2 km/s across most of the domain. The largest discrepancies occur near the Atacames seamount, where both SNR is low and complex geology complicates picking. This highlights that while DeepFB substantially reduces the manual workload, human oversight may remain necessary in challenging environments (Fig. 11). Nonetheless, the reduction in processing time is considerable: DeepFB enables efficient large-scale first-break picking with only minor manual post-processing required to refine results in problematic areas. Future studies need to determine whether DeepFB generalises well enough to be directly applied to other active-seismic datasets. In this context, transfer learning or fine-tuning strategies could be explored, where the pretrained DeepFB model is adapted to new datasets with a limited number of labelled examples, potentially reducing the need for full retraining and accelerating deployment on diverse seismic settings.

Beyond the immediate application to 2D tomography, the approach has broader implications. The methodology can be extended to 3D velocity model building for the entire HIPER2 dataset (Fig. 1) in future, which will provide new insights into the Ecuadorian subduction zone. In this context, the term 3D refers to a simultaneous 3D recording to constrain a 3D velocity model. The dataset would still need to be processed line by line for offlines and inlines, making the computational cost comparable to repeated 2D applications of the method and thus scalable to larger datasets. While some retraining or fine-tuning with additional data from other lines may improve performance, the trained DeepFB model is expected to generalise well and provide reliable first-arrival picks for previously unseen profiles. Moreover, the chunk-based architecture and uncertainty-aware picking strategy could be adapted for other seismic datasets, including distributed acoustic sensing (DAS, Zhu et al. 2023), microseismic monitoring, and induced seismicity studies (e.g. Lim et al. 2024;

Heuel et al. 2025). In this study, we only used the hydrophone channel of the OBS, but future improvements could include additional data from the vertical and horizontal components of the OBS.

In summary, DeepFB and automatic hyperparameter optimisation through evolutionary algorithms represents a step forward in automatic first-break picking by combining chunk-based learning, spatial context, and noise robustness. It achieves accuracy comparable to manual picking while significantly accelerating processing workflows. While some limitations remain in areas of low SNR and complex geology, the method has strong potential for broader application and scalability to 3D studies.

ACKNOWLEDGMENTS

J. Heuel was funded by the German Federal Ministry of Research, Technology and Space under the name ”AI-based monitoring of geothermal Seismicity” with the grant number 01|S23030A. A. Delsuc has been supported by the French government, through the UCAJEDI Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-15-IDEX-01. The authors thank the handling editors Alice Power and Anandaroop Ray, Vaibhav Vijay Ingale, and an anonymous reviewer for reviewing our manuscript and providing constructive feedback to improve our manuscript.

DATA AND CODE AVAILABILITY

The code of DeepFB including trained models is available at GitHub (https://github.com/JanisHe/FirstBreakPicking) including examples for prediction and a few examples from the training dataset. Bugs and other problems can be reported through GitHub. Propulate is also available at GitHub (https://github.com/Helmholtz-AI-Energy/propulate). The SEGY dataset of the HIPER2 experiment and associated picks are available at Delsuc et al. (2025). Data will be released after the acceptance of the paper and can be accessed at https://radar.kit.edu/radar/en/dataset/edwh9kz9t586hw2k?token=yhfdhtENFvGwEpIrcJua.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Almadani, S., Ibrahim, E., Abdelrahman, K., Al-Bassam, A., & Al-Shmrani, A., 2015. Magnetic and seismic refraction survey for site investigation of an urban expansion site in abha district, southwest saudi arabia, *Arabian Journal of Geosciences*, **8**(4), 2299–2312.

Coppens, F., 1985. First arrival picking on common-offset trace collections for automatic estimation of static corrections, *Geophysical prospecting*, **33**(8), 1212–1231.

Delsuc, A., Galve, A., Laigle, M., Rietbrock, A., Schenini, L., Agurto-Detzel, H., Skrubej, A., Marcaillou, B., Vaca, S., Maier, A., & Paulatto, M., 2025. Characterizing the forearc basement and subducting structures of the northern Ecuadorian margin [Dataset], *radar kit*.

Font, Y., Segovia, M., Vaca, S., & Theunissen, T., 2013. Seismicity patterns along the Ecuadorian subduction zone: new constraints from earthquake location in a 3-D a priori velocity model, *Geophysical Journal International*, **193**(1), 263–286.

Gailler, A., Charvis, P., & Flueh, E. R., 2007. Segmentation of the Nazca and South American plates along the Ecuador subduction zone from wide angle seismic profiles, *Earth and Planetary Science Letters*, **260**(3-4), 444–464.

Galve, A., 2020. Hiper cruise, *RV L'Atalante*.

Galve, A., 2022. Hiper (suite) cruise, *RV L'Atalante*.

He, K., Zhang, X., Ren, S., & Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034.

Heuel, J., Maurer, V., Frietsch, M., & Rietbrock, A., 2025. Picking Induced Seismicity with Deep Learning (piSDL), *Seismica*, **4**(*2*).

Hu, L., Zheng, X., Duan, Y., Yan, X., Hu, Y., & Zhang, X., 2019. First-arrival picking with a u-net convolutional network, *Geophysics*, **84**(6), U45–U57.

Huang, Q., Sun, J., Ding, H., Wang, X., & Wang, G., 2018. Robust liver vessel extraction using 3d u-net with variant dice loss function, *Computers in biology and medicine*, **101**, 153–162.

Ioffe, S. & Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37 of **Proceedings of Machine Learning Research**, pp. 448–456, PMLR, Lille, France.

Kanamori, H. & McNally, K. C., 1982. Variable rupture mode of the subduction zone along the Ecuador-Colombia coast, *Bulletin of the Seismological Society of America*, **72**(4), 1241–1253, Publisher: The Seismological Society of America.

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

Korenaga, J. & Sager, W. W., 2012. Seismic tomography of Shatsky Rise by adaptive importance sampling, *Journal of Geophysical Research: Solid Earth*, **117**(B8), 2012JB009248.

Korenaga, J., Holbrook, W. S., Kent, G. M., Kelemen, P. B., Detrick, R. S., Larsen, H., Hopper, J. R., & Dahl-Jensen, T., 2000. Crustal structure of the southeast Greenland margin from joint refraction and reflection seismic tomography, *Journal of Geophysical Research: Solid Earth*, **105**(B9), 21591–21614.

Li, H., Xu, Z., Taylor, G., Studer, C., & Goldstein, T., 2017. Visualizing the loss landscape of neural nets, *arXiv preprint arXiv:1712.09913*.

Lim, C. S. Y., Lapins, S., Segou, M., & Werner, M. J., 2024. Deep learning phase pickers: how well can existing models detect hydraulic-fracturing induced microseismicity from a borehole array?, *Geophysical Journal International*, **240**(1), 535–549.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P., 2017. Focal loss for dense object detection, in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988.

Mardan, A., Blouin, M., Fabien-Ouellet, G., Giroux, B., Vergniault, C., & Gendreau, J., 2024. A fine-tuning workflow for automatic first-break picking with deep learning, *Near Surface Geophysics*, **22**(5), 539–552.

Mousavi, S. M. & Beroza, G. C., 2022. Deep-learning seismology, *Science*, **377**(6607), eabm4470.

Münchmeyer, J., Woollam, J., Rietbrock, A., Tilmann, F., Lange, D., Bornstein, T., Diehl, T., Giunchi, C., Haslinger, F., Jozinović, D., et al., 2022. Which picker fits my data? a quantitative evaluation of deep learning based seismic pickers, *Journal of Geophysical Research: Solid Earth*, **127**(1), e2021JB023499.

Nocquet, J.-M., Jarrin, P., Vallée, M., Mothes, P. A., Grandin, R., Rolandone, F., Delouis, B., Yepes, H., Font, Y., Fuentes, D., Régnier, M., Laurendeau, A., Cisneros, D., Hernandez, S., Sladen, A., Singaucho, J.-C., Mora, H., Gomez, J., Montes, L., & Charvis, P., 2017. Supercycle at the Ecuadorian subduction zone revealed after the 2016 Pedernales earthquake, *Nature Geoscience*, **10**(2), 145–149.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., et al., 2018. Attention u-net: Learning where to look for the pancreas, *arXiv preprint arXiv:1804.03999*.

Parra, H., Benito, M. B., & Gaspar-Escribano, J. M., 2016. Seismic hazard assessment in continental Ecuador, *Bulletin of Earthquake Engineering*, **14**(8), 2129–2159.

Ronneberger, O., Fischer, P., & Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Springer International Publishing, Cham.

Sabbione, J. I. & Velis, D., 2010. Automatic first-breaks picking: New strategies and algorithms, *Geophysics*, **75**(4), V67–V76.

Sennson, J. L. & Beck, S. L., 1996. Historical 1942 Ecuador and 1942 Peru subduction earthquakes and earthquake cycles along Colombia-Ecuador and Peru subduction segments, *Pure and applied geophysics*, **146**, 67–101.

Taubert, O., Weiel, M., Coquelin, D., Farshian, A., Debus, C., Schug, A., Streit, A., & Götz, M., 2023. Massively parallel genetic optimization through asynchronous propagation of populations, in *International Conference on High Performance Computing*, pp. 106–124, Springer.

Wen, Z. & Ma, J., 2025. Effective first-break picking of seismic data using geometric learning methods, *Remote Sensing*, **17**(2).

Woollam, J., Münchmeyer, J., Tilmann, F., Rietbrock, A., Lange, D., Bornstein, T., Diehl, T., Giunchi, C., Haslinger, F., Jozinović, D., et al., 2022. Seisbench—a toolbox for machine learning in seismology, *Seismological Society of America*, **93**(3), 1695–1709.

Ye, L., Kanamori, H., Avouac, J.-P., Li, L., Cheung, K. F., & Lay, T., 2016. The 16 april 2016, mw 7.8 (ms 7.5) ecuador earthquake: A quasi-repeat of the 1942 ms 7.5 earthquake and partial re-rupture of the 1906 ms 8.6 colombia–ecuador earthquake, *Earth and Planetary Science Letters*, **454**, 248–258.

Yin, Y., Han, L., Zhang, P., Lu, Z., & Shang, X., 2023. First-break picking of large-offset seismic data based on cnns with weighted data, *Remote Sensing*, **15**(2).

Yoshimoto, M., Kumagai, H., Acero, W., Ponce, G., Vásconez, F., Arrais, S., Ruiz, M., Alvarado, A., Pedraza García, P., Dionicio, V., et al., 2017. Depth-dependent rupture mode along the ecuador-colombia subduction zone, *Geophysical Research Letters*, **44**(5), 2203–2210.

Yuan, S., Liu, J., Wang, S., Wang, T., & Shi, P., 2018. Seismic waveform classification and first-break picking using convolution neural networks, *IEEE Geoscience and Remote Sensing Letters*, **15**(2), 272–276.

Yung, S. & Ikelle, L. T., 1997. An example of seismic time picking by third-order bicoherence, *Geophysics*, **62**(6), 1947–1952.

Zhao, B.-L., 2008. Application of multi-component seismic exploration in the exploration and production of lithologic gas reservoirs, *Petroleum Exploration and Development*, **35**(4), 397–412.

Zhu, W. & Beroza, G. C., 2018. Phasenet: A deep-neural-network-based seismic arrival-time picking method, *Geophysical Journal International*, **216**(1), 261–273.

Zhu, W., Biondi, E., Li, J., Yin, J., Ross, Z. E., & Zhan, Z., 2023. Seismic arrival-time picking on distributed acoustic sensing data using semi-supervised learning, *Nature Communications*, **14**(1), 8192.

APPENDIX A: INFLUENCE OF GAUSSIAN FILTER ON FIRST-BREAKS

The standard deviation σ of the Gaussian filter (eq. 3) controls the sharpness of the transition between the pre–first-arrival and post–first-arrival regions (blueish and yellow areas in Fig. A1, respectively). Without Gaussian smoothing, the transition is abrupt and the resulting first-break picks exhibit staircase-like artifacts along the moveout curve (Fig. A1a). For moderate values of σ, the transition zone is smoothed, yielding a laterally coherent and continuous first-arrival moveout (Fig. A1b). For large σ values (Fig. A1c), the transition zone becomes overly diffuse, and the picked first-arrival times become increasingly sensitive to the chosen picking threshold. Consequently, although larger σ values improve moveout smoothness, excessive smoothing may degrade pick

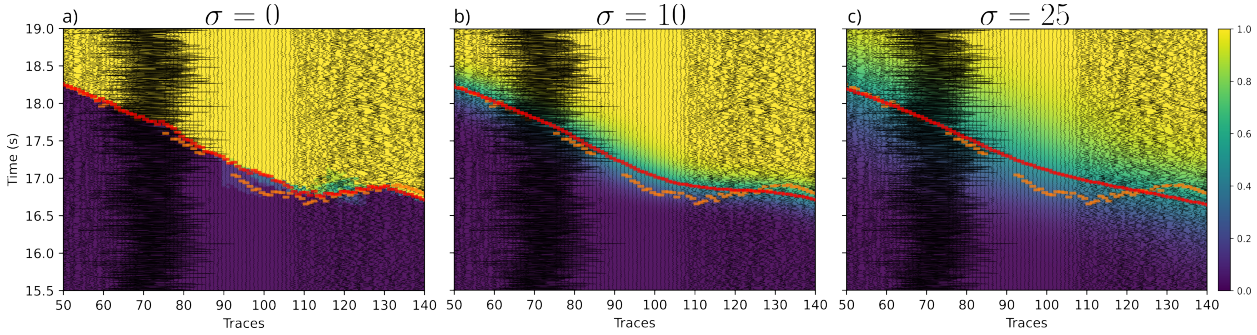

Figure A1. Effect of Gaussian filter standard deviation σ on first-break picking: (a) no smoothing, (b) moderate smoothing, and (c) strong smoothing. Increasing σ improves moveout continuity but broadens the transition zone and reduces pick resolution.

resolution and lead to information loss that can adversely affect subsequent applications, such as traveltime tomography.

APPENDIX B: CHECKERBOARD TESTS

Checkerboard tests are used to determine the spatial resolution of our 2D velocity models. We use anomalies of three different sizes to understand which feature sizes are resolved in our model and where.

APPENDIX C: STANDARD DEVIATION

The standard deviation between all output model generated during the Bayesian inversion of Tomo2D is used to assess the uncertainties in the model (Korenaga & Sager 2012). Furthermore, a mask has been applied on both model in the area without any ray path using the derivative weighted sum.

This paper has been typeset from a TEX/LATEX file prepared by the author.

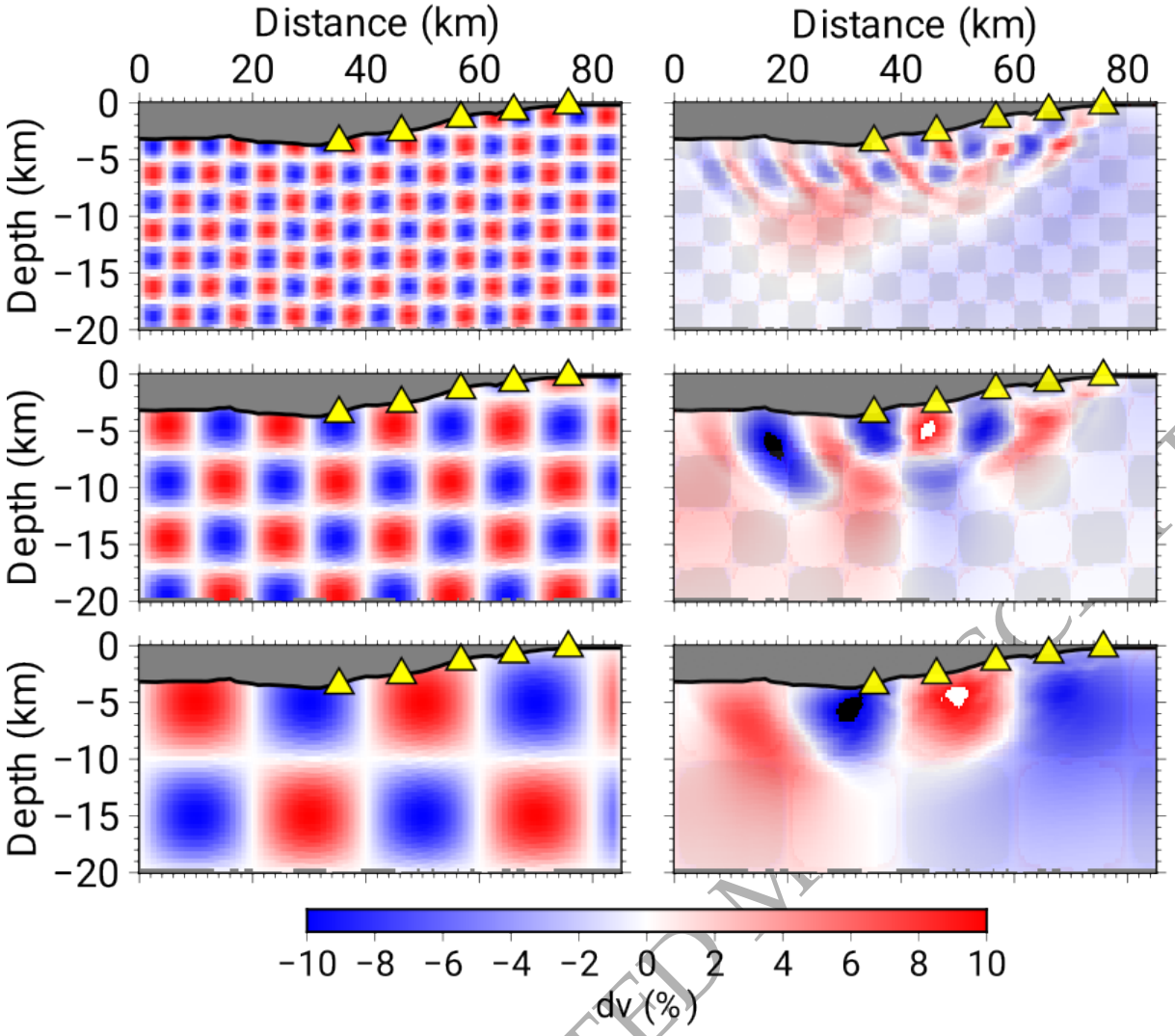

Figure B1. Checkerboard tests were performed on line C for anomalies of different sizes. From top to bottom, we used anomalies of 5x2.5 km, 10x5 km and 20x10 km, respectively. The input models, which have alternating positive and negative anomalies of 10% of the initial starting velocity, are visible in the left-hand column and as transparent black-and-white layers on top of the recovered models.

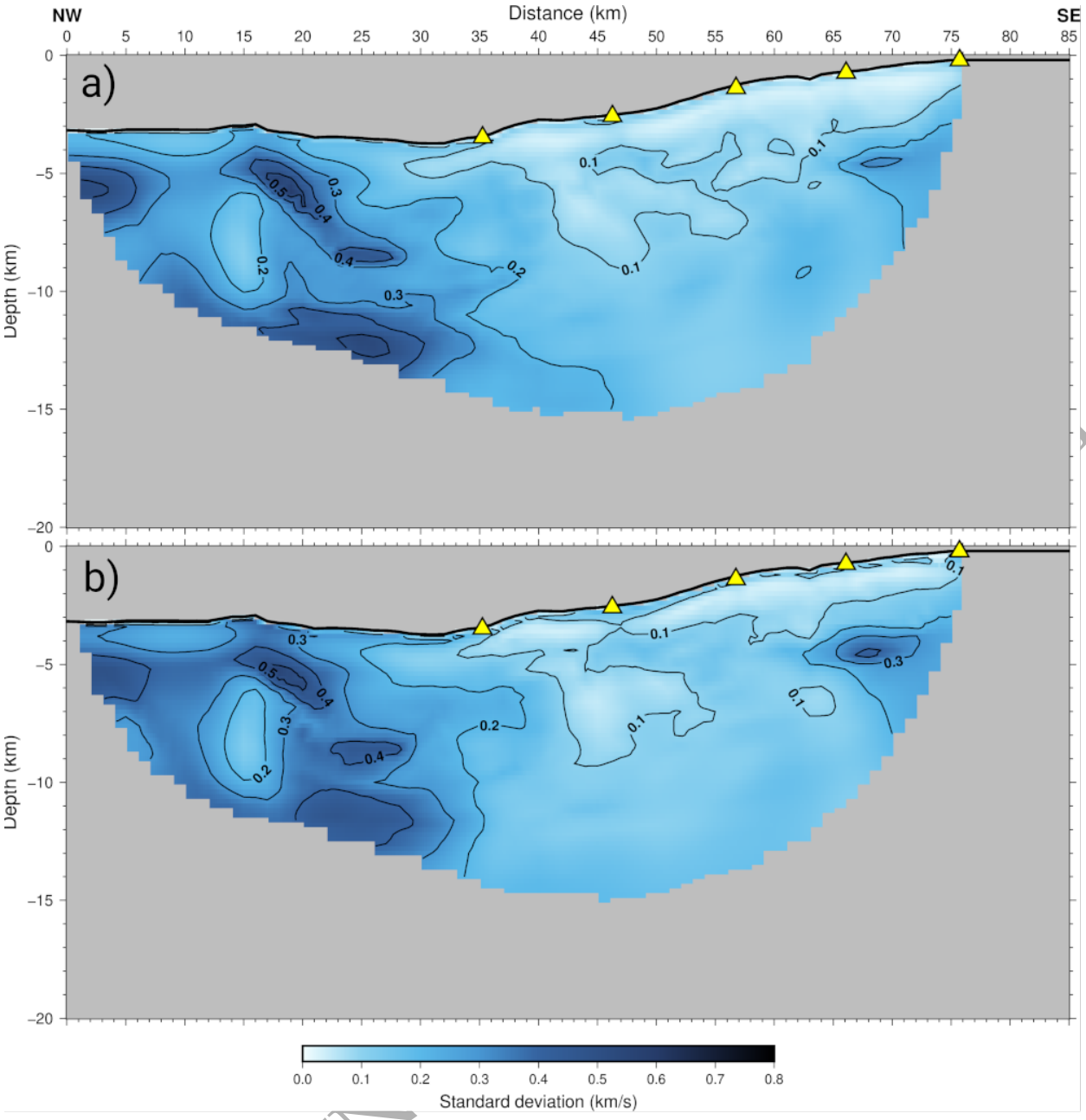

Figure C1. Standard deviation for the velocity model obtained through manual picking (a) and through DeepFB picks (b)