GEOPHYSICS, VOL. 89, NO. 1 (JANUARY-FEBRUARY 2024); P. WA233–WA246, 15 FIGS., 3 TABLES.
10.1190/GEO2023-0146.1

An in-depth study of U-net for seismic data conditioning: Multiple removal by moveout discrimination

Ricard Durall[1], Ammar Ghanim[1], Norman Ettrich[1], and Janis Keuper[2]

ABSTRACT

Seismic processing often involves suppressing multiples that are an inherent component of collected seismic data. Elaborate multiple prediction and subtraction schemes such as surface-related multiple removal have become standard in industry workflows. In cases of limited spatial sampling, low signal-to-noise ratio, or conservative subtraction of the predicted multiples, the processed data frequently suffer from residual multiples. To tackle these artifacts in the postmigration domain, practitioners often rely on Radon transform-based algorithms. However, such traditional approaches are both time-consuming and parameter dependent, making them relatively complex. In this work, we present a deep learning-based alternative that provides competitive results, while reducing the complexity of its usage, and, hence simplifying its applicability. Our proposed model demonstrates excellent performance when applied to complex field data, despite it being exclusively trained on synthetic data. Furthermore, extensive experiments show that our method can preserve the inherent characteristics of the data, avoiding undesired oversmoothed results, while removing the multiples from seismic offset or angle gathers. Finally, we conduct an in-depth analysis of the model, where we pinpoint the effects of the main hyperparameters on real data inference, and we probabilistically assess its performance from a Bayesian perspective. In this study, we put particular emphasis on helping the user reveal the inner workings of the neural network and attempt to unbox the model.

INTRODUCTION

In seismic exploration, geophysicists interpret reflections of acoustic waves to extract information from the subsurface. These reflections can be classified as primaries or multiples. Primary reflections are those seismic events whose energy has been reflected once, and they are used to describe the subsurface interfaces. In contrast, multiples are events whose energy has been reflected more than once and appear when the signal has not taken a direct path from the source to the receiver. The presence of multiples in the recorded data set can trigger erroneous interpretations because they interfere not only with the analysis in the poststack domain, (e.g., stratigraphic interpretation) but also with the prestack analysis, (e.g., amplitude-variation-with-offset (AVO) inversion). For this reason, the demultiple process plays a crucial role in any seismic processing workflow. Multiple-attenuation methods can be classified as predictability- and separation-based. Predictability-based approaches exploit the repetitive nature of multiples and their inherent connection to primaries. In general, they consist of two steps: a multiple prediction step, in which a model of multiples is created, followed by adaptive subtraction (Verschuur et al., 1992; Abma et al., 2005) where the predicted multiples are adaptively matched and removed from the recorded wavefield. Some of the most widely used methods are wavefield extrapolation (Berryhill and Kim, 1986; Wiggins, 1988; Wang et al., 2011), surface-related multiple elimination (SRME) (Berkhout, 1985; Verschuur, 1991; Ma et al., 2019) and the inverse scattering series free-surface multiple elimination (Carvalho et al., 1991; Weglein et al., 1997, 2003; Ma et al., 2019). All of these approaches are recognized for their effectiveness

The first and second authors contributed equally to this paper. Manuscript received by the Editor 28 February 2023; revised manuscript received 25 August 2023; published ahead of production 13 September 2023; published online 8 December 2023.
[1]Fraunhofer ITWM, Kaiserslautern, Germany. E-mail: ricard.durall.lopez@itwm.fraunhofer.de (corresponding author); ammar.ghanim@itwm.fraunhofer.de; norman.ettrich@itwm.fraunhofer.de.
[2]Fraunhofer ITWM, Kaiserslautern, Germany and Offenburg University, IMLA, Offenburg, Germany. E-mail: janis.keuper@itwm.fraunhofer.de.

DOI:10.1190/geo2023-0146.1

in mitigating free-surface multiples. Nevertheless, they involve numerous steps, and their efficacy is highly influenced by factors such as acquisition setting and geometry, as well as signal-to-noise ratio (S/N) (Gisolf and Verschuur, 2010; Kostov et al., 2015; Ma et al., 2019). In addition, to not risk damaging weak primaries, the adaptive subtraction step is often applied conservatively, resulting in residual multiple energy in the final image (Wang et al., 2011; Zhang et al., 2021). Recently, closed-loop SRME (CL-SRME) (Lopez and Verschuur, 2015; Zhang and Verschuur, 2021) has been proposed to tackle the shortcomings of SRME in shallow-water settings, nonetheless, the high computational demand still poses a challenge.

However, separation-based methods translate seismic data into intermediate domains, where one can eliminate multiples based on different characteristics of multiples and primaries (Weglein et al., 2011). The concept here is to exploit the fact that, on average, multiples have encountered a lower velocity than the primaries, and thus multiples are expected to exhibit an increasing residual moveout (RMO) along the offset dimension. Although suffering from their own set of limitations, separation-based methods are of a computationally simpler nature and can be applied at various stages of the processing workflow. One of the most widespread approaches to making use of this feature is the parabolic Radon transform (PRT) (Hampson, 1986). It translates prestack gathers from a time-offset to a *tau-p* space, by mapping them by a set of parabolic events. By design, PRT works best in the case of multiples perfectly following parabolic paths and for unlimited offset axis, both of these aspects are, nevertheless, not realizable in practice (Hampson, 1986). As a consequence, PRT can potentially degrade parts of the primary signal. Another limitation appears when dealing with gathers that are coarsely sampled. In such cases, data sparsity can lead to false energy mapping to the *tau-p* space, which in turn leads to insufficient separation of primaries and multiples. This either creates residual multiple energy or removes primary energy. To address some of the aforementioned weaknesses, the high-resolution Radon multiple removal method has been introduced (Sacchi and Ulrych, 1995; Sacchi and Porsani, 1999; Trad et al., 2003). It is, however, an approach of higher complexity, requiring the interpreter to manually fine-tune numerous parameters. Moreover, another disadvantage arises from the necessary time-consuming step of picking an appropriate mute function in the *tau-p* space to separate primaries from potential multiples. Oftentimes, the nature of the data set requires a laterally varying mute function design, adding yet another level of complexity. When it comes to industry workflows, the usage of predictability-based methods in the premigration domain, e.g., SRME, and separation-based methods in the postmigration gather conditioning, e.g., PRT demultiple, are typically combined. In this fashion, interpreters can leverage the best from both methodologies and achieve more reliable outcomes.

With the introduction of deep learning, a new vein of methods has emerged (Breuer et al., 2020; Bugge et al., 2021; Qu et al., 2021; Wang et al., 2022). These approaches are based on artificial neural network architectures, which are universal approximators, i.e., they can, in theory, model any continuous function. Breuer et al. (2020) present a deep learning-based method to trim statics and remove multiples on postmigration common-depth point (CDP) gathers using a moveout discriminator approach trained on synthetic data. Subsequently, Bugge et al. (2021) propose a similar approach that simultaneously tackled both demultiple and denoising on prestack gathers. Qu et al. (2021) present a hybrid workflow combining a deep neural network trained on synthetic data for shallow reflection reconstruction and PRT for deeper event reconstruction followed by CL-SRME. Finally, Wang et al. (2022) introduce a solution that exploits noise and data augmentation applied to training data generated using SRME or PRT for the free-surface multiple removal. Unfortunately, although the aforementioned methods have contributed to improving state-of-the-art results on multiple removal, they still suffer from generalization problems. To deal with this issue, Qu et al. (2021) require the generation of synthetic training data for each field of interest. Similarly, the approach by Wang et al. (2022) necessitates the synthetization of labeled data per survey using conventional multiple elimination methods for real data applications. Note, however, that these are proxy solutions, as they do not attempt to solve the survey-data set dependency of the model, but rather bypass it.

In this paper, we introduce and perform a detailed analysis of a separation-based automated end-to-end deep-learning approach, which can be applied on moveout-corrected post-migration CDP gathers to remove events that follow parabolic-like patterns while preserving the primary energy at cross-points. As already pointed out by Qu et al. (2021), training the model on data sets preprocessed using traditional methods introduces the limitations of such methods into the model as a side effect. To decouple the model from such limitations, we follow the workflow introduced in Breuer et al. (2020) and train a convolutional neural network (CNN) with synthetic pairs of multiple-contaminated and multiple-free gathers. The network is trained on feature-rich synthetic CDP gathers designed to enable the trained network to identify multiples in the prestack domain based on the reflection moveout paths rather than periodicity, thus making the model highly generalizable and independent of acquisition design. Furthermore, our approach works in a parameter-free manner, relieving the user from any manual task. In addition, we conduct an in-depth hyperparameter search, where we study the role played by the different components and their impact on the outcome. To that end, we visualize the inner workings of our neural network, to pinpoint the effect of the main hyperparameters on physical events. Finally, extensive in-field evaluations show that our model is able to preserve the inherent characteristics of the data in different scenarios, and thus, to generalize well. As a result, our approach can be seen as an alternative to traditional moveout separation-based approaches in the postmigration stage, such as PRT, in existing processing workflows.

U-NET VISUALIZATION

U-net (Ronneberger et al., 2015) is a CNN topology, which was initially designed for semantic segmentation tasks in the medical domain. However, due to its generalization capacity, it has been widely adapted to various other domains. The architecture of U-net is divided into two paths: the contraction path, known as the encoder, designed to capture the image's context, and the expanding path, referred to as the decoder, responsible for facilitating accurate localization. Both paths are symmetric and made of blocks of convolutional layers followed either by a down-sampling operation (encoder) or by an up-sampling operation (decoder). In addition to the encoder-decoder scheme, U-net has long skip connections that bypass some layers and connect different blocks from the encoder to their counterparts from the decoder. These shortcuts provide alternative paths for the gradient during back-propagation that help the model to incorporate fine-grained details in the predictions. Figure 1 shows the architecture of U-net for the demultiple scenario.

CNN architectures are successfully used in a large variety of applications, ranging from computer vision to natural language processing. They are made up of neurons that have learnable parameters arranged in filter-shape structures. Each of these neurons receives some inputs, performs a dot product, and finally, applies a nonlinear activation function (e.g., sigmoid or rectified linear unit [ReLU]) (Nair and Hinton, 2010). The output of the activation for a given filter is called a feature map or an activation map. Although the learning mechanism (back-propagation) is well understood, the intrinsic details, such as the reason why a specific decision or prediction is made, are not. As a result, neural networks are typically treated as black box models. To better understand the internal workings, we visualize different components of the network. In particular, we investigate the filters and the feature maps to try to conceptually unravel the learning of the model when dealing with demultiple problems.

On the left side of Figure 2, we can see some filters that the network has learned. Seemingly, they do not display any human-recognizable pattern from which one can draw conclusions. The statistics are, however, more informative. The filters' weights appear to always follow a Gaussian distribution, independent of the layer. Similar observations by Gavrikov and Keuper (2022) suggest that convolution filters do not have distribution shifts along different axes of meta-parameters, such as data type, task, architecture, or layer depth. Nonetheless, we notice that the first block might break these empirical deductions, meaning that depth could indeed play a certain role in shallow layers. On the right side of Figure 2, we can observe some feature maps from different blocks. These intermediate representations display how the network modifies the input image and help us understand how multiples are identified and suppressed. On the one hand, as expected, we can visually assess a gradual loss of resolution (high-frequency components) in the first blocks, due to their down-sampling operations from the contraction path. The opposite effect is seen in the last blocks, caused by the up-sampling operations from the expanding path. However, contrary to what might be intuitive, the network is not learning to suppress multiples directly from the beginning. In fact, they are present in all the blocks, and almost in all feature maps. What the network seems to learn, instead, is to identify the multiples in each block to have a full understanding of the event. In this manner, in the very last layer, the model combines the feature maps in such a way that the undesirable events (multiples) are canceled out.

TRAINING DATA SET

When interpreting real seismic data, we do not have the ground truth (GT) (annotated data). Unfortunately, these labeled data are some of the cornerstones of any supervised deep-learning model. Manual interpretation is an effective way to acquire GT, but it is an expensive and time-consuming process. Furthermore, its outcomes rarely contain all the events that would define the characteristics of the subsurface. To address this issue, in the demultiple scenario, one could create real labeled data, by using a traditional approach, for example, the PRT (Wang et al., 2022). Nevertheless, the network would be biased and limited by the performance of the traditional approach (Qu et al., 2021).

In this work, we introduce a network that is able to suppress multiples regardless of the domain and nature of the seismic gathers, i.e., offset or angle domain and time or depth domain. To achieve this, we systematically generate a substantial data set comprising 40,000 synthetic pairs of multiple-contaminated and multiple-free

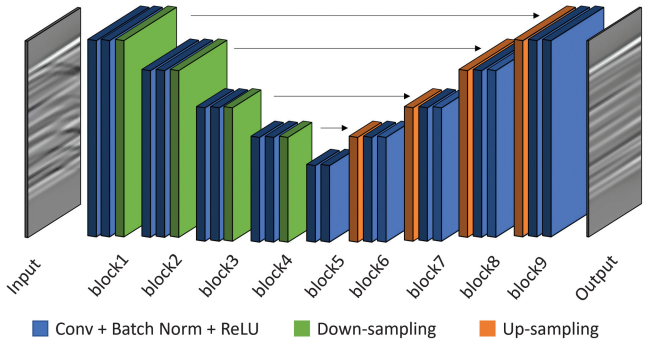


Figure 1. U-net architecture for multiple attenuation. The task of this model is to learn to remove multiples while keeping the rest of the image unmodified, i.e., primaries and data characteristics.

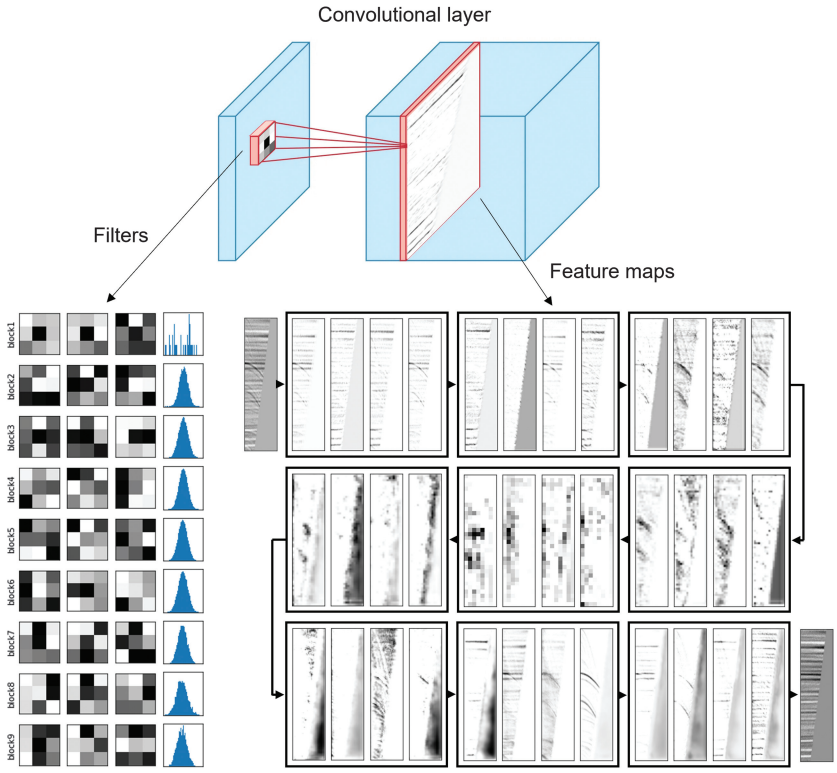


Figure 2. Visualization of the U-net inner structure after each block. Left: From top to bottom, each row shows three randomly selected filters and the histogram of the first moment (mean) of all the filters from each block, where the x-axis is the weight values and the y-axis is the frequency of appearance. Right: From upper left to bottom right, following a "Z" shape, the transformations that the input image undergoes before the multiples are removed. Each group shows four random feature maps and belongs to one block of the U-net structure (see Figure 1).

DOI:10.1190/geo2023-0146.1

gathers. Exercising precise control over the features of the synthetically generated data set via an extensive parameter space empowers us to create a training data set that significantly enhances the model's capacity to perform well on a wide range of real-world scenarios. Crucially, it is worth noting that our network's proficiency does not stem from its ability to identify multiples based on their periodic relationships to specific primary signals. Instead, our goal is to exploit geometric differences in the RMO and cross-points between multiples and even barely visible primaries in the prestack gather. This parameter space consists of (1) variations of the density of multiples and primaries, and their position along the vertical axis; (2) variations of the strength of the RMO effect controlling minimum multiple moveout; (3) variations of the spectral components of the source wavelet together with a central frequency decay along the vertical axis; and (4) variations of amplitude change with offset/angle.

The synthetic gathers for training are created by generating a prestack reflectivity series $r_p(t_0) = r_p(t, h = 0)$ at zero offset $h = 0$, first, for the primary reflections p. For the expansion to nonzero-offset $r_p(t, h > 0)$, linear interval velocity functions are defined, converted to RMS v_p velocity, and applied in the hyperbolic normal moveout (NMO) formula to calculate the event time $t_p(t_0, h, v_p(t_0))$. For the amplitude part, the Shuey approximation (Shuey, 1985) is used with $r_p(t_0)$ as the intercept of the amplitude variation with angle equation, to which we add a gradient term. A preliminary version of the primary-only gather is generated by convolving $r_p(t, h)$ with a synthetic source wavelet of Ricker-type whose degrees of freedom are central frequency, bandwidth, and phase shift. Furthermore, we generate custom wavelets through the superposition of two individual wavelets, which are weighted and mutually shifted. Analogously, we also generate a nonzero-offset reflectivity series $r_m(t, h)$ for the multiples, followed by convolution with the same wavelets as used for the primaries. The main difference from the primary reflectivity is the lower velocity v_m used to calculate $t_m(t_0, h, v_m(t_0))$. This gather of the multiples is added to the primary-only gather to generate a gather that contains both, the primaries and the multiples. Subsequent NMO correction of the gathers with perturbed RMS velocity, obtained by time-dependent perturbations of the interval primary velocity model, approximates gathers after prestack migration. The primaries appear almost flat (not necessarily perfectly flat) and multiples show stronger positive moveout than the primaries, thus they are seen in the gathers as events that intersect primaries and have a larger vertical extent. The NMO-corrected primary-only gathers are the GT in the process of training the network; the NMO-corrected gathers of primaries and multiples are the input gathers. The values of all parameters are obtained by Monte Carlo sampling of the parameter space within the bounds defined by the user of the modeling routine. Setting such bounds follows the guidelines of the variability of the corresponding parameters in field data acquisition and data preprocessing. It seems reasonable to let, e.g., the central frequency of the wavelet vary between 10 and 150 Hz to account (for deep seismic data) for the range of frequency content of various typical sources and the decay of frequency toward large depth. For the case of synthetic training data for shallow applications with typically much higher frequency content, one could stay within the same frequency limits and the same vertical resolution, because the network does not acknowledge physical units and, thus, makes no difference between realizations of N times higher frequency data on an N times higher resolved vertical grid. Some parameter bounds, however, have a steering effect on the functionality of the trained network. For example, defining a minimum-allowed moveout for the removed multiples teaches the network not to suppress potentially nonflattened primaries.

ANALYSIS OF U-NET PARAMETERIZATION

Hyperparameters are values that control the learning process of neural networks. They define different aspects of the model, such as the learning rate, optimizer, depth, activation function, and loss function, just to mention a few. In general, neural networks are notorious for being very sensitive to the choice of hyperparameters, resulting in relatively different outcomes when the parameters are slightly modified.

In this section, we identify and describe the empirical effects that some hyperparameters have on our multiple-attenuation network. In particular, we focus on the impact of the optimizer, sampling technique, kernel size, loss function, and depth. To that end, we average validation results of 25 independent runs to guarantee reproducibility. We evaluate these results on four different metrics: mean-square error (MSE), S/N, structural similarity, and peak correlation. Furthermore, we validate the outcome on synthetic and real data sets. In this manner, we ensure certain generalizability and neutrality in our observations.

Optimization functions

Within a neural network, the optimizer is an algorithm that modifies the weights of the network to minimize the loss function. They are built upon the idea of gradient descent, i.e., the greedy approach of iteratively decreasing the loss function by following the gradient. There are two main groups of optimizers: adaptive and nonadaptive methods. Hardt et al. (2016) argue that nonadaptive methods, such as stochastic gradient descent (SGD), are conceptually more stable for convex and continuous optimization, having smaller generalization errors. They also prove that, under certain conditions, the results can be carried over to nonconvex loss functions. Follow-up work by Wilson et al. (2017) finds empirical evidence of the poor generalization performance of adaptive optimization methods, such as adaptive moment estimation (Adam) (Kingma and Ba, 2014). Even when adaptive methods achieve a better training loss than nonadaptive methods, the test performance is worse. Finally, Choi et al. (2019) claim that the hyperparameter of the optimizer could be the reason that adaptive optimization algorithms failed to generalize.

In our experiments, we evaluate the impact of SGD with momentum and Adam optimizers for the demultiple task. Figure 3a shows the validation metrics in synthetic data for the two selected optimizers. In these plots, we can observe how the Adam optimization converges faster than the nonadaptive one (SGD) and also ends up in lower local minima, i.e., all the metrics reach better values. Nonetheless, although the gap between both optimizers seems to be significant when inspecting synthetic results, the differences are negligible (see Figure 3b). Furthermore, surprisingly, the demultiple outcomes on the real data set suggest that the model trained with the Adam optimizer tends to fail to generalize more often, and its results are not always consistent, varying among different runs. In Figure 3c, we display some results on real data, where we see how the Adam approach occasionally suppresses the primary energy, as it does for the reflection marked by the red rectangle from the second gather, and leaves some residual multiples in the far stack, as it does for the

reflection marked by the red rectangle from the sixth and seventh gathers. Despite the fact that our model is trained using synthetic data, the system is meant to be applied to real data. Therefore, we prefer to use the SGD optimizer.

Sampling technique and kernel size

The CNN-based models gradually down-sample their inputs so that the receptive fields of the deeper filters can reach most of the image at a certain depth. By doing that, the pixel dependencies, which lie far away from each other in an image, can be captured. This is an important aspect for any neural network that needs to interact with content that is spread on the input image, such as in fault detection or multiple removal. In our study, we conduct a twofold analysis related to the sampling, we evaluate the effect of different sampling techniques, and we analyze the impact of the kernel size.

Sampling techniques refer to those methods that decrease or increase the size of an input. In the contraction path of U-net, there are two down-sampling approaches: the pooling operation and the convolution operation. Although typically the pooling operation does not have learnable parameters (less computationally demanding), the convolutional operation does have such parameters. As a consequence, the latter can capture additional information, whereas the pooling will always imply a loss of information. In the expanding path, the decoder recombines the features sequentially until it recovers the original input size. To that end, this path requires up-sampling operations. Similarly to the contraction path, there are two main approaches: interpolation operation and transposed convolution operation. The first type of operation is parameter-free and lossy, and the second is the opposite. To evaluate the impact of the sampling methods, both down- and up-sampling, we check the different combinations. For the sake of simplicity, we restrict our analysis to the default configurations, which are max-pooling as a nonlearnable down-sampling technique and bilinear as a nonlearnable up-sampling technique.

Based on Figure 4a, experiments with transposed convolutions have less stable runs, nonetheless, all the sampling techniques have similar performance. Therefore, the extra computational cost of the learnable operations is not justified. Furthermore, the combination of max-pooling and bilinear, which are both nonlearnable sampling methods, provides the most stable results. Testing with synthetic and real data shows no difference among the configurations.

In addition to the sampling techniques, the kernel size might also contribute to the final outcomes. This hyperparameter determines to what degree the sampling operation down- and up-samples the corresponding input. Given that we work with elongated events, we empirically analyze the impact of kernels with square and nonsquare shapes and assess the impact of more aggressive sampling, i.e., the down- and up-sampling factors. Table 1 and Figure 5

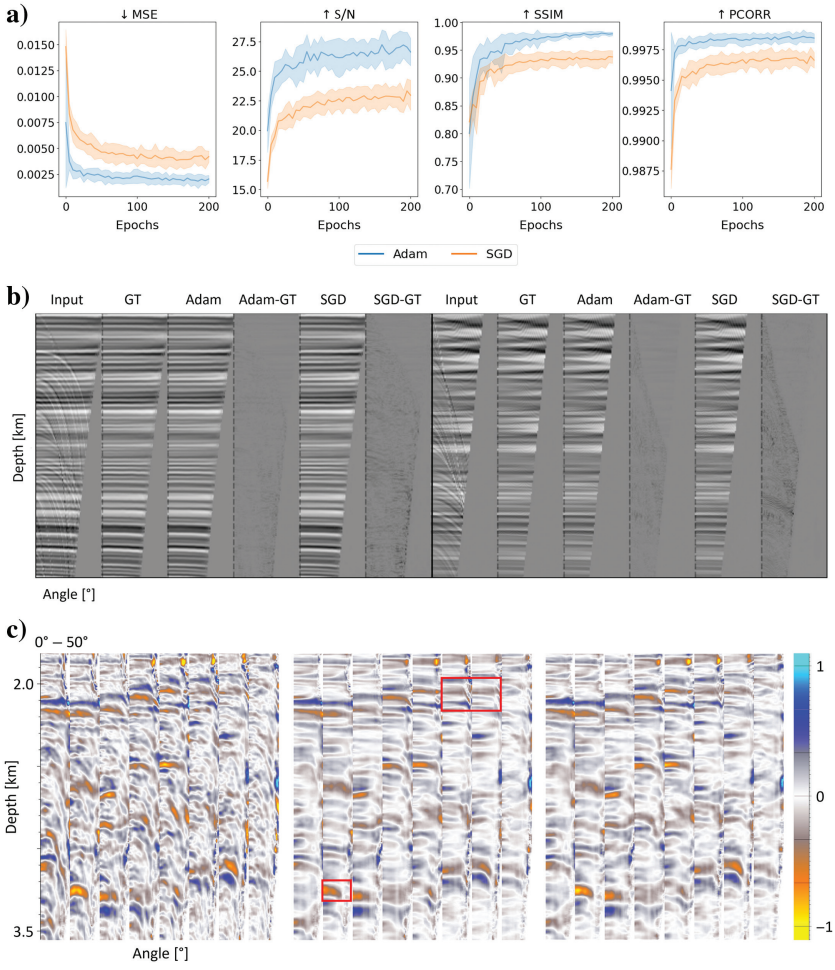


Figure 3. The optimizers used in the neural network might converge to different local minima. These figures show how these optimizers behave under synthetic and real scenarios. (a) Optimizer assessment based on different quality metrics. (b) Two random gathers from our validation synthetic data set. We infer results on a pretrained model with Adam and a pretrained model with SGD. Both outcomes are multiple-free, and their differences with the GT are negligible. (c) A collection of eight gathers from real data. From left to right, the input data (with multiples), the output from a pretrained model with Adam, and the output from a pretrained model with SGD. The red rectangles on the middle panel of (c) mark the undesirable effects of the pretrained model with Adam.

Table 1. Each case belongs to a particular arrangement of the kernels.

	Kernel shape	Aggressiveness	Configuration
Case A	Square	Low	1 × 1, 2 × 2, 2 × 2, 2 × 2
Case B	Nonsquare	High	1 × 2, 2 × 4, 2 × 4, 2 × 4
Case C	Square	Low	2 × 2, 2 × 2, 2 × 2, 2 × 2
Case D	NonSquare	High	2 × 4, 2 × 4, 2 × 4, 2 × 4

Note: For example, in Case B (1 × 2, 2 × 4, 2 × 4, 2 × 4): the kernel of block 1 is defined as 1 × 2, and the following three blocks as 2 × 4. This means that the first block will down-sample its input only along the y dimension by a factor of two, and the x dimension will remain unmodified. Then, the second block will down-sample its input along the x dimension (incidence angle or offsets) by a factor of two, and by a factor of four in its y dimension (time or depth). Note that all of these operations will be reversed in the expanding part.

describe the scenarios of our examples. Although the validation metrics seem to report the same behavior for all of the kernels, we observe a consistent improvement after quality control when using a 1×1, 2×2, 2×2, 2×2 kernel sequence (see Figure 4b). Models trained with the larger max-pooling kernels appear to remove multiples more aggressively, i.e., oversmoothing results and suppressing far stack energy of events that exhibit small moveout, marked by the rectangles in Figure 4c. According to Araujo et al. (2019), the effective maximum receptive field of the model trained with the chosen kernel sequence is 112 samples, meaning that a single pixel in the output is influenced by a square of 112×112 pixels from the input, as shown in Figure 6. This appears to be sufficient to observe multiples and their localized interactions with primaries, and hence we conclude that such a localized view is more important than the global view of the gather for this task. Moreover, the models trained with larger kernels seem to be more sensitive to the initial weights than their counterparts trained with smaller max-pooling kernels, as confirmed in the probabilistic study (see the following section).

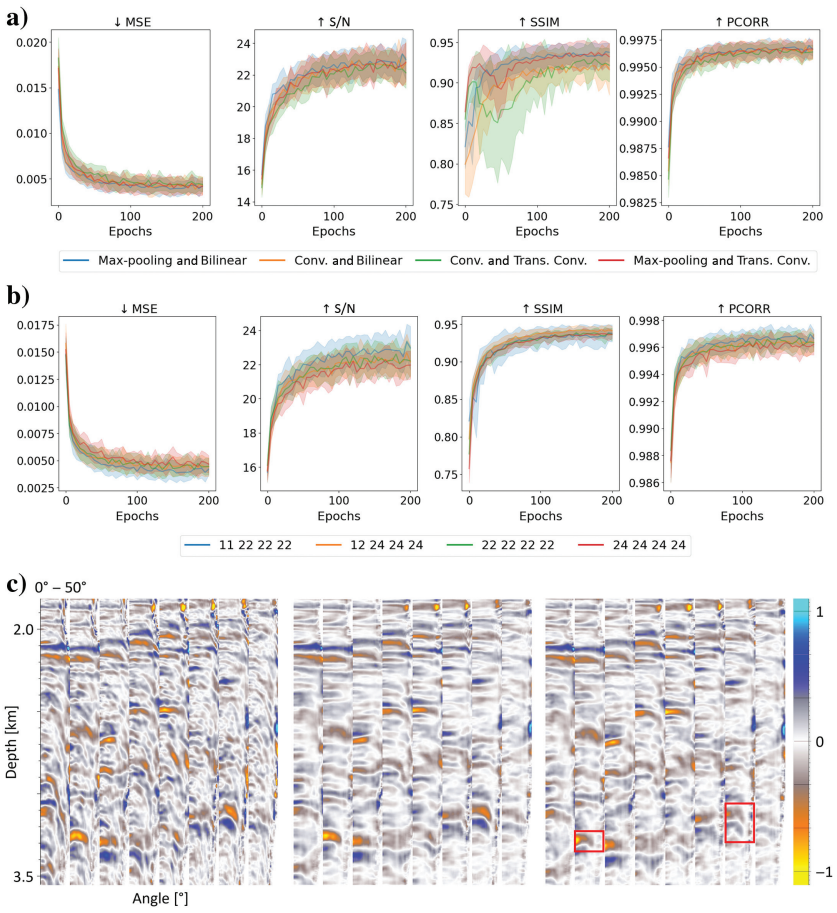


Figure 4. The sampling technique and kernel size determine how the system modifies the scale of the input images. Depending on the task, they can lead to undesired artifacts. (a) Assessment of down- and up-sampling based on different quality metrics, (b) assessment of max-pooling kernel configurations based on different quality metrics, and (c) collection of eight gathers from real data. From left to right, the input data (with multiples), the output from Case A, and the output from Case D. The red rectangles on the right figure mark the undesirable effects of Case D (see Table 1).

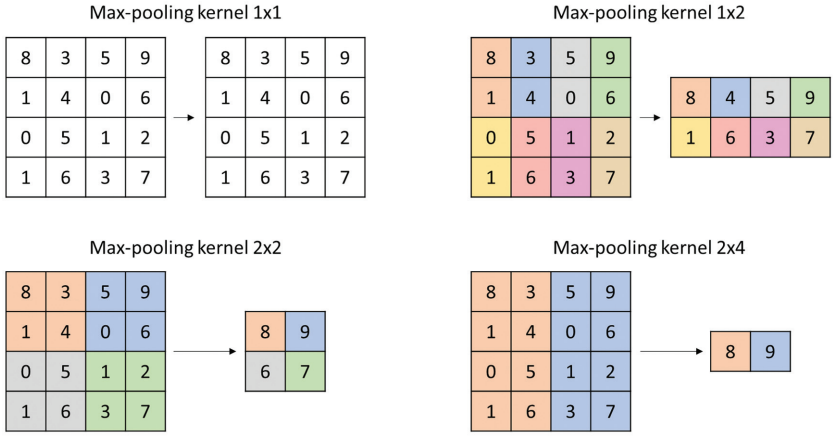


Figure 5. Visualization of the different max-pooling kernels assessed in this study.

Loss function

The selection of a loss function is a challenging task that has a direct impact on the model's behavior. For this reason, it is important to choose a function that captures the relevant information that needs to be propagated through the network. In this work, we advocate for the use of MSE for its simplicity and capacity to deal with outliers. This loss calculates the difference between the model's predictions $\hat{y}$ and the ground truth y, squares and averages it, across the entire data set (N samples). Mathematically, it can be formulated as

$$\mathrm{MSE} = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2. \qquad (1)$$

Besides formulating the loss function, it is crucial to define the primary objective. This entails clearly outlining the specific task that the network is designed to achieve. To elaborate further, we propose two distinct objectives: direct and inverse. Given an input image x, the direct proposal tackles the demultiple problem by optimizing the prediction $\hat{y}$, which is a multiple-free image. The inverse approach, however, formulates the solution from another perspective. It defines the objective task as an optimization problem, where the prediction $\hat{y}$ should contain only the multiples of the input image, i.e., $x - y$ (see Figure 7a). In this way, the network should focus exclusively on identifying the multiples, omitting the rest. Once the model is able to do that, we can subtract the prediction from the input image to obtain a multiple-free image. In Figure 7b, we plot the metrics using different objective functions. Interestingly, the

results from both scenarios are similar. We hypothesize that the network learns to cancel out the same features, in the direct and inverse formulation, and consequently, the outcomes seem equivalent. Nonetheless, more advanced loss functions could potentially improve the results.

Depth of the network

The goal of our neural network is to model a function F that maps the raw input data x to a multiple-free output. To that end, we create F by concatenating n nonlinear functions f, i.e., $F(x) = f_1(f_2(..f_n(x)))$. Notice that adding more layers provides higher capacity to the network, which leads to deeper networks. In our experiments, we investigate the effect of three levels of depth. We take as a baseline the standard model shown in Figure 1, which consists of nine blocks. Then, we remove two down-sampling and two up-sampling layers to create a smaller version, called "small U-net." Finally, we repeat the procedure, but this time adding two down-sampling and two up-sampling layers into the baseline. We call this last model big U-net. Table 2 shows the details of each topology and their inference times.

Figure 8a and 8b shows the depth analysis from which we derive the following statements. (1) The small U-net is too shallow and does not have sufficient capacity to suppress the multiples and occasionally oversmoothes the gathers. As a result, metrics and real data underperform when compared with the standard model. (2) The big U-net model is overparametrized, and therefore, the extra layers do not offer any further improvement. Note, however, that this analysis involves a training data set of a constant size and thus, training the big U-net model on a larger data set could yield different results. In summary, our standard model has the optimal trade-off between quality and size.

Alternative topologies

The attention U-net architecture, proposed by Oktay et al. (2018), enhances the standard U-net model by incorporating self-attention mechanisms (Jetley et al., 2018). These mechanisms, such as channel and spatial attention, allow the model to adaptively emphasize relevant features during both the encoding and decoding stages. By selectively highlighting informative regions and suppressing noise or irrelevant details, the attention U-net improves its overall performance. In contrast, the MultiResUNet architecture introduced by Ibtehaz and Rahman (2020) introduces the concept of multiresolution residual blocks within the U-net structure. The main idea is that the incorporation of multiple resolution paths will help the architecture to effectively capture local and global contextual information. The fusion of information from different resolution levels enables MultiResUNet to learn intricate details and capture a broader context, enhancing its segmentation capabilities. In terms of architecture details, attention U-net and MultiResUNet consist of nine layers with max-pooling operations at resolutions of 2×2, 2×2, 2×2, 2×2. Attention U-net has a parameter count of 34.9 million and uses a combination of max-pooling and bilinear interpolation for down-sampling and up-sampling. MultiResUNet has 7.2 million parameters and uses max-pooling for down-sampling and transposed convolution for

Table 2. The number of parameters and inference time in dependence on network complexity defined by the number of block layers.

	# of the block layer	# of parameters (M)	Inference time (s)
Standard U-net	9	17.2	22.91 ± 0.02
Big U-net	13	276.8	42.70 ± 0.01
Small U-net	5	1.0	13.67 ± 0.04

Note: The last column shows the inference time for each model when testing on 6000 images 64×256 pixels.

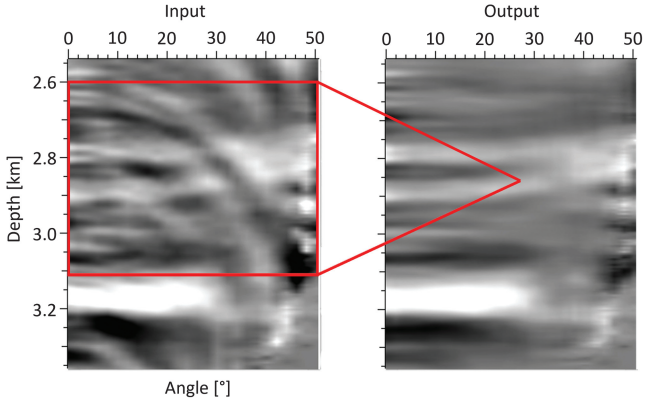


Figure 6. Visualization of the size of the receptive field of one pixel using the configuration Case A.

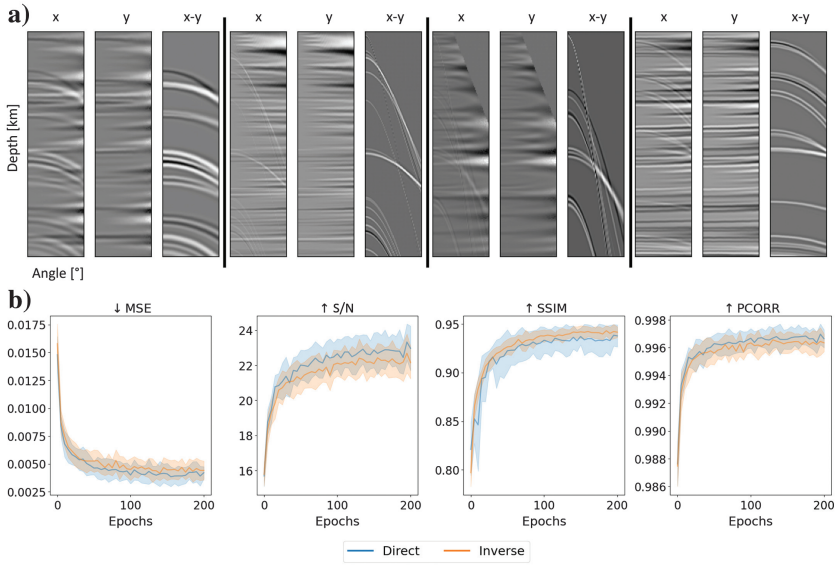


Figure 7. Evaluation of the impact of two different objective functions. (a) Four training examples. Given an input image with multiples x, our goal is to build a network that can eliminate them. To achieve this, either we design a model that removes the multiples directly, i.e., targeting y, or we design a model that only keeps the multiples, i.e., targeting $x - y$, and then we subtract this result from the input x. (b) Loss function assessment based on different quality metrics.

up-sampling. In Figure 9a, we show the evaluation scores from topology analysis. Twenty-five models have been trained with each topology and tested on synthetic testing data. Based on the depicted curves, the performance of the attention U-net is comparable with that of the proposed U-net architecture, whereas the MultiResUNet demonstrates noticeably inferior results. Figure 9c shows the results of U-net, MultiResUNet, and attention U-net and amplitudes extracted along two selected reflectors, which are plotted above the gathers. Based on these plots, it becomes evident that the MultiResUNet affects the absolute amplitudes of primaries, whereas the U-net and attention U-net output primaries with an overall similar amplitude intercept and gradient. Moreover, the MultiResUNet has not successfully suppressed the multiple crossing of the red reflector. Figure 9b shows another comparison of these three topologies, this time, however, on numerous gathers from the Norwegian Sea. A comparable observation can be made based on this figure.

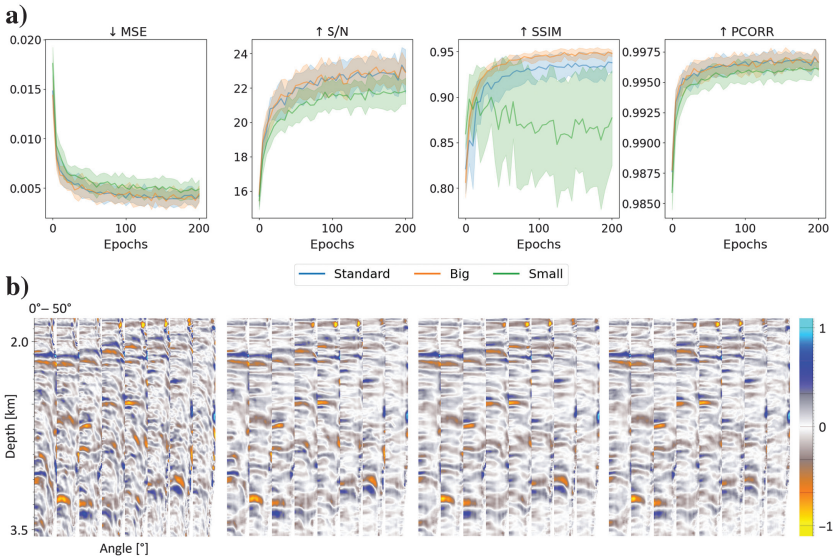


Figure 8. The capacity of a network plays an important role in any learning task. Too shallow topologies cannot capture the complexity of the data; too deep can overfit, not improving the final results. (a) Depth assessment based on different quality metrics and (b) a collection of eight gathers from real data. From left to right, the input data (with multiples), the output from the small U-net, the output from the standard U-net, and the output from the big U-net.

In summary, both the attention U-net and the MultiResUNet introduce modifications to the standard U-net architecture to address specific limitations and potentially enhance performance. For our use case, the MultiResUNet demonstrates a tendency to diminish the absolute amplitude across the gathers rather than solely addressing the presence of multiples. In contrast, the U-net and attention U-net exhibit similar outcomes, although the attention U-net occasionally exhibits too severe of an effect. It is important to note that the attention U-net, as opposed to standard U-net, is not fully convolutional and thus is input-shape dependent. Despite its competitive performance, the constraint of this topology to a specific input shape is too limiting for our use case.

BAYESIAN INVESTIGATION

Once we have analyzed the role of the different parameters, it is important to quantify the uncertainty of the model, i.e., epistemic uncertainty. In this manner, one can determine how reliable the actual predictions are, avoiding miscalibrated models. To that end, we need to move from a deterministic approach, where we solely rely on a point estimator, to a probabilistic approach, where we leverage Bayesian probabilities via Bayesian neural networks (BNNs). Although traditionally BNNs have been computationally expensive and difficult to train, recent approximations, such as deep ensembles (Lakshminarayanan et al., 2017), concrete dropout (Gal et al., 2017), and stochastic weight averaging Gaussian (Maddox et al., 2019), have eased these constraints.

In this work, we have implemented deep-ensemble learning, which can be considered a special case of BNNs (Wilson et al., 2022). The idea behind ensemble learning comes from the observation that aggregating the predictions of a large set of average-performing but independent predictors can lead to better predictions than a single well-performing expert predictor (Breiman, 1996). In our case, however, we prefer to use such a method to obtain the uncertainty associated with the underlying processes. This is achieved by normalizing and then computing the standard deviation of the predictions of numerous sampled

Table 3. Summary of all models used for the Bayesian analysis.

	# of block layers	Optimizer	Configuration
Model A (Subsection optimization)	9	SGD	1 × 1, 2 × 2, 2 × 2, 2 × 2
Model B (Subsection optimization)	9	ADAM	1 × 1, 2 × 2, 2 × 2, 2 × 2
Model C (Subsection sampling — Case B)	9	SGD	1 × 2, 2 × 4, 2 × 4, 2 × 4
Model D (Subsection sampling — Case C)	9	SGD	2 × 2, 2 × 2, 2 × 2, 2 × 2
Model E (Subsection Sampling — Case D)	9	SGD	2 × 4, 2 × 4, 2 × 4, 2 × 4
Model F (Subsection depth — big U-net)	13	SGD	1 × 1, 2 × 2, 2 × 2, 2 × 2
Model G (Subsection depth — small U-net)	5	SGD	1 × 1, 2 × 2, 2 × 2, 2 × 2
Model H (Subsection topology — MultiResUNet)	9	SGD	2 × 2, 2 × 2, 2 × 2, 2 × 2
Model I (Subsection topology — attention U-net)	9	SGD	2 × 2, 2 × 2, 2 × 2, 2 × 2

model parameterizations. Notice that the resulting range of values indicates the percentage with respect to the output signal amplitude. As a result, if the different models agree on the multiple-free solutions and their absolute amplitudes, then the uncertainty is low. Otherwise, the uncertainty is high.

Figure 10a–10d shows four prestack gathers from a real data set and their associated uncertainties for a set of experiments (see Table 3). These uncertainty figures show the areas of the prestack gather where the models have a lack of knowledge, resulting in a certain ambiguity within the multiple removal process. In practice, this manifests itself as variations in the amplitude or shape of the removed events across parameterized models. Low uncertainty is displayed in black or dark purple, high uncertainty is displayed in pink and yellow. Given that the demultiple model is not perfect and hence its epistemic uncertainties are not zero, one has to target a model that does not exhibit high amplitude uncertainties that align with primaries. Otherwise, this would suggest that some realizations of the model remove or suppress primary energy, which is highly undesirable. However, uncertainties following a parabolic or a linear moveout are tolerated, as they potentially belong to a multiple. Such uncertainty suggests that the model is not certain about whether the event is a multiple or there is a mismatch in the amplitude. We observe that Models B, C, E, and H exhibit a clear increased uncertainty throughout the entire gather, hinting that some of these model realizations do affect the amplitudes of primaries. On the contrary, Models A, D, F, G, and I only produce uncertainties with significant values that follow parabolic events which we presume to be multiples. Finally, although these five models provide similar uncertainty maps, Models A, F, and I achieve the best qualitative performance (see the previous section). Therefore, as already mentioned, we prefer Model A because it offers a better trade-off between quality, size, and flexibility.

SYNTHETIC EXAMPLE

Figure 11a shows the outcomes of our method and compares them to the results obtained from the Radon-based demultiple technique. The assessment is carried out on gathers obtained from a synthetic data set. This data set is created using a 3D finite-difference method that incorporates a free-surface boundary condition. The gathers are represented in the depth-offset domain, and our deep-learning approach was directly applied in this domain. Both our method and the Radon-based demultiple technique successfully eliminate the clearly defined parabolic events within a depth range of 3–5 km. However, in the far-offset shallow section, our deep-learning approach exhibits superior performance in removing steeply dipping linear noise when compared with the Radon-based demultiple method. For deep-learning approaches, which take seismic data as input and produce seismic data as output, amplitude preservation of the primaries is of utmost importance. Figure 11b shows the amplitude preservation capabilities of the U-net (our deep-learning model) and the Radon-based demultiple results. Displayed amplitudes are extracted along the red and blue lines from the raw gather and plotted above their respective gathers. The red line follows a potential phase-reversal event with a positive intercept, whereas the blue line traces an event with a negative intercept and a positive gradient. Both the deep-learning approach and the Radon-based method preserve the overall amplitude trend. In addition to multiple removal, an AVO-preserving denoising effect of the deep-learning approach can be observed. In the difference plots, marked by arrows, we observe how high amplitude events in the removed energy along the lines align closely for both approaches.

FIELD EXAMPLES

In addition to tests on synthetic data, the trained model has been tested on numerous real postmigration data sets without any additional fine-tuning. Figure 12 shows the results of our method as compared with a traditional Radon-based demultiple approach on a real data set from the Norwegian Sea, subsequently referred to

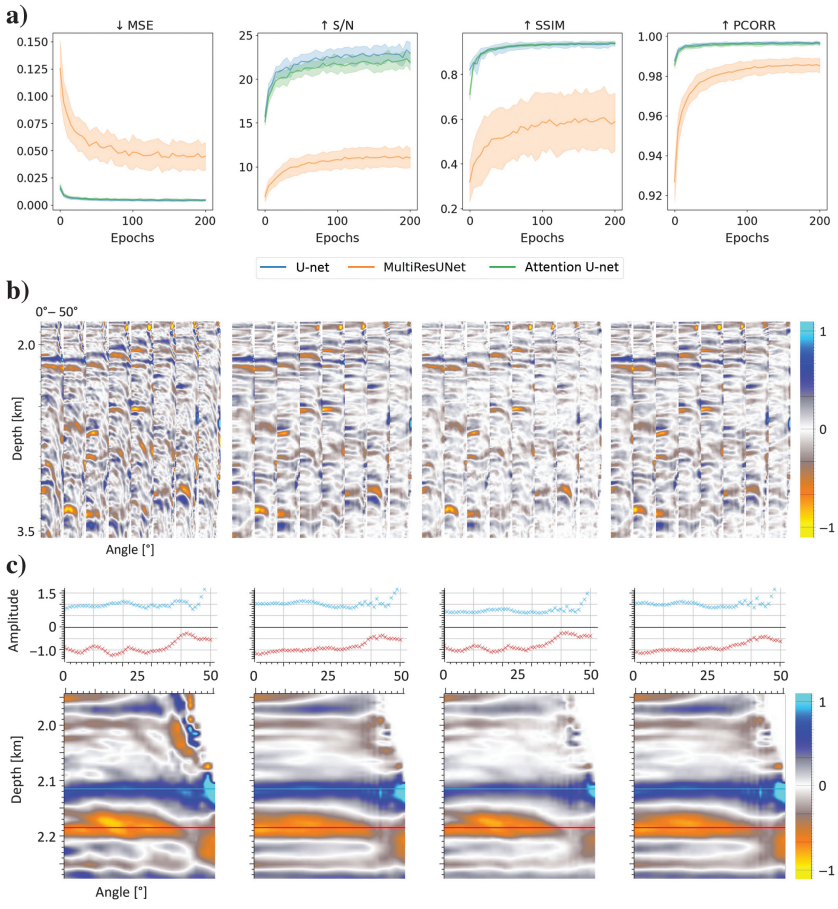


Figure 9. Results of an assessment of three alternative U-net topologies. (a) A topology assessment based on different quality metrics (calculated on testing synthetic data), (b) a collection of eight gathers from real data, and (c) amplitudes extracted along two reflectors from a real data set. For (b and c): From left to right, the input data (with multiples), the output from standard U-net, the output from MultiResUNet, and the output from attention U-net.

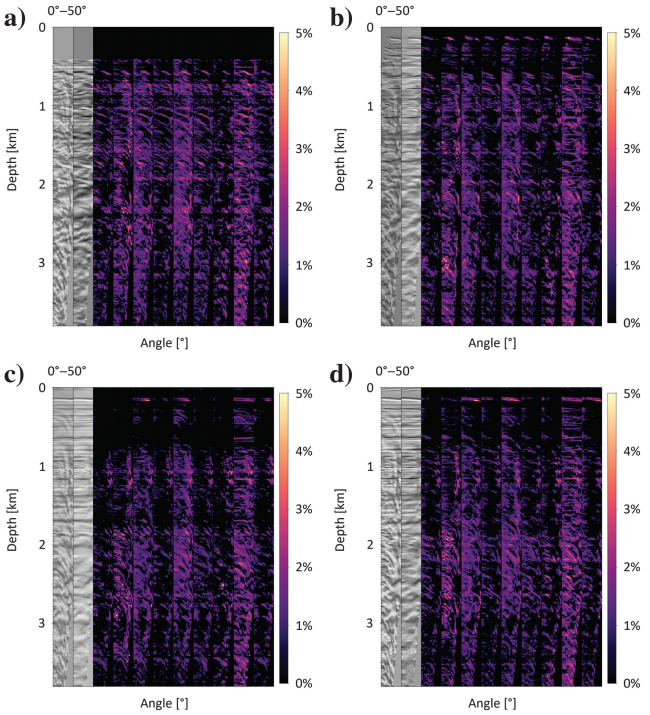


Figure 10. Four prestack gathers of a real data set from the Norwegian Sea, the U-net demultiple result from our best model (Model A), and the collection of seven associated uncertainties. Looking at the uncertainty maps, from left to right: results from Model A, B, C, D, E, F, G, H, and I. The colorbar displays the standard deviation between runs within the same gather.

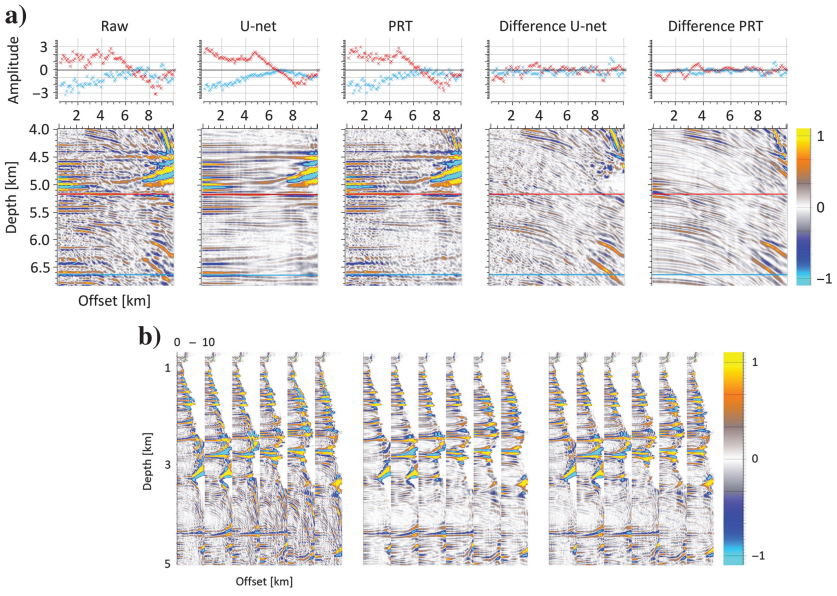


Figure 11. Part of a depth-offset prestack gather from a synthetic data set modeled with a 3D finite-difference method with a free-surface boundary. (a) From left to right, the input data (with multiples), the output from our approach, the output from the parabolic Radon approach, removed multiples by our approach, and removed multiples by the parabolic Radon approach. Amplitudes extracted along the red and blue lines in the respective gathers are plotted above. (b) Several prestack gathers in the depth-offset domain. From left to right, the input data (with multiples), the output from our approach, and the output from the parabolic Radon approach.

as Field Data A. In addition, Figure 13 shows a similar comparison using prestack gathers of the Volve field data set (Equinor, 2018) from the Norwegian North Sea, subsequently referred to as Field Data B. The deep-learning approach was applied directly in the depth-angle domain, whereas the PRT application involved depth-to-time and angle-to-offset conversions. For both real data examples, we plot the removed multiples for the proposed and the traditional methods to help visualize the main discrepancies between the two systems. From this visualization, we observe that PRT predominantly removes events along idealized parabolas, which are unlikely to closely represent multiples in real data CDP gathers. Complex overburden causes deviations from the parabolic shape, thus, the mapping of such multiples to clusters of points in *tau-p* space is in disagreement with the attempt of modern high-resolution PRTs to achieve a sparse *tau-p* representation. Our deep-learning approach, in contrast, does not make use of any specific path of the multiples and is able — based on what was shown to the network during training — to remove along its full path any given event that intersects other events with smaller RMO. In this way, it also removes converted wave energy and steeply down-dipping linear noise, and it is better suited to remove residuals of a demultiple process of the premigration steps, i.e., which appear only in the far stack. Such events can be seen in the far stack of the first three gathers in Figure 12. We also provide the results of the same data sets as full-stack sections in Figures 14 and 15. Herein, we can see how the lateral coherency of the removed events is consistent in both approaches. For Field Data A, the removed multiples by the U-net model appear to align better with the overlaying stratigraphy, resulting in sharper results.

DISCUSSION

Given postmigration prestack gathers, our deep-learning approach identifies the multiples and cancels them out from the output result based on their moveout and geometric interference with primaries in a parameter-free manner. The main success of our implementation is not only the ability to remove multiples, but to do it while preserving the high-frequency components that characterize the data, and to generalize to different data scenarios without the need to retrain. Although denoising is a common postprocessing step targeting these frequency components, a non-controlled application of it can lead to smoothing of the data, resulting in a loss of relevant features. We believe that seismic interpretation is a challenging task, therefore, any processing method needs to guarantee the preservation of these characteristics.

Despite the fact that in the past years CNNs have been extensively used in seismic applications, there is still a lack of rigorous explanation of hyperparameters choice. Thus, we think that the geophysics community would benefit from our approach to unbox neural networks to establish the relationship between the neural network parameters and their effects on the demultiple task from a deterministic and probabilistic perspective. In particular, our extensive set of experiments has

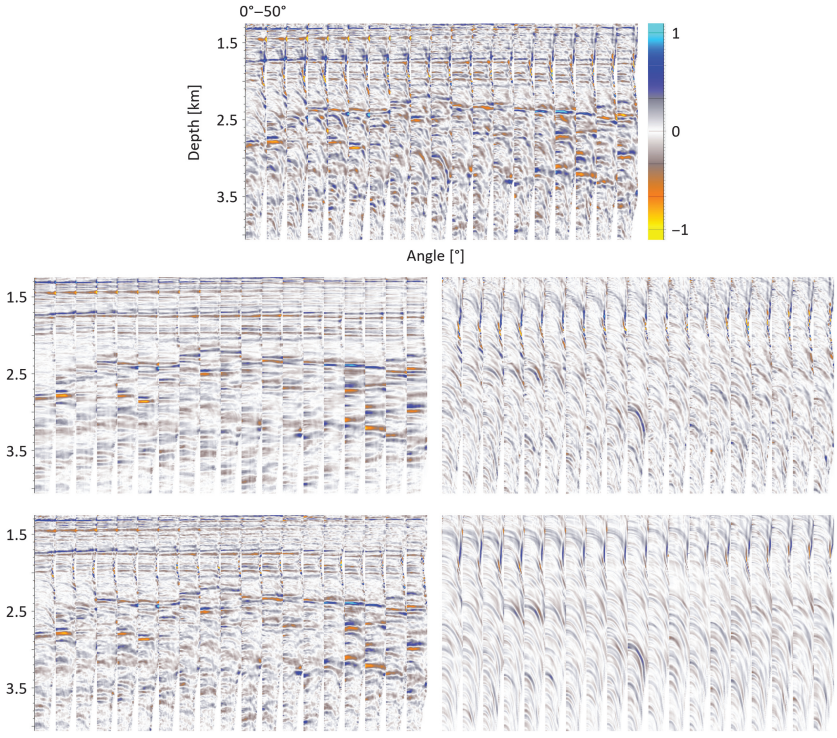


Figure 12. Several prestack angle gathers of a real data set from the Norwegian Sea. First row: migrated raw angle gathers. Second row: angle gathers, U-net demultiple result (left), and removed multiples (right). Third row: angle gathers, Radon-based demultiple (left) and removed multiples (right).

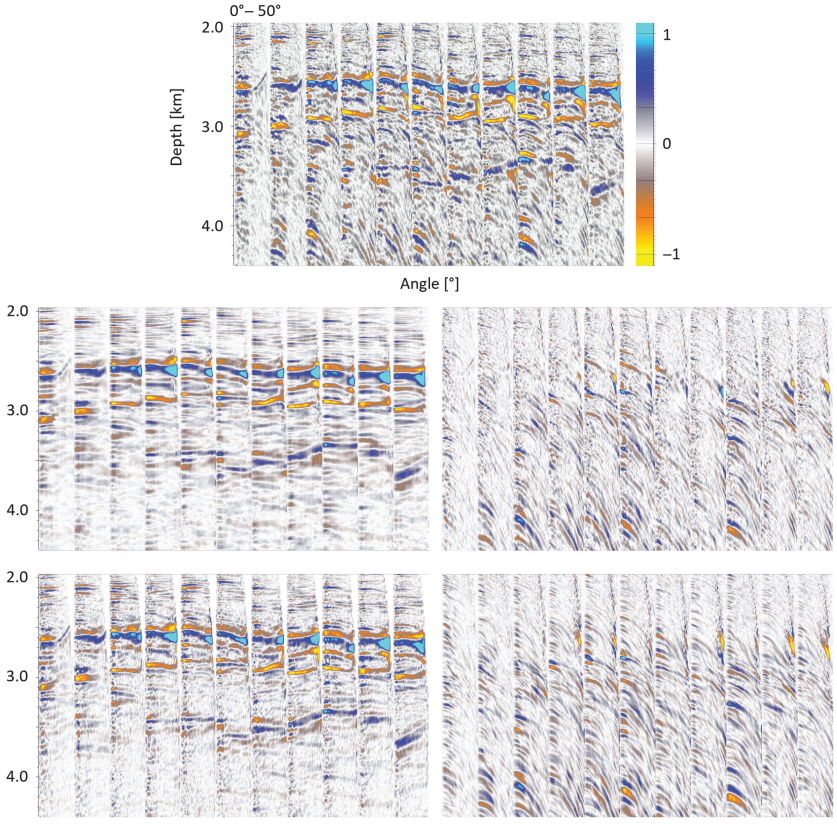


Figure 13. Several prestack angle gathers of a real data set from the Volve data set. First row: migrated raw angle gathers. Second row: angle gathers, U-net demultiple result (left), and removed multiples (right). Third row: angle gathers, Radon-based demultiple (left) and removed multiples (right).

DOI:10.1190/geo2023-0146.1

Figure 14. Full-stack of depth angle gathers of a real data set from the Norwegian Sea. First row: migrated raw full-stack section. Second row: full-stack, U-net demultiple (left), and removed multiples (right). Third row: full-stack Radon-based demultiple (left) and removed multiples (right).

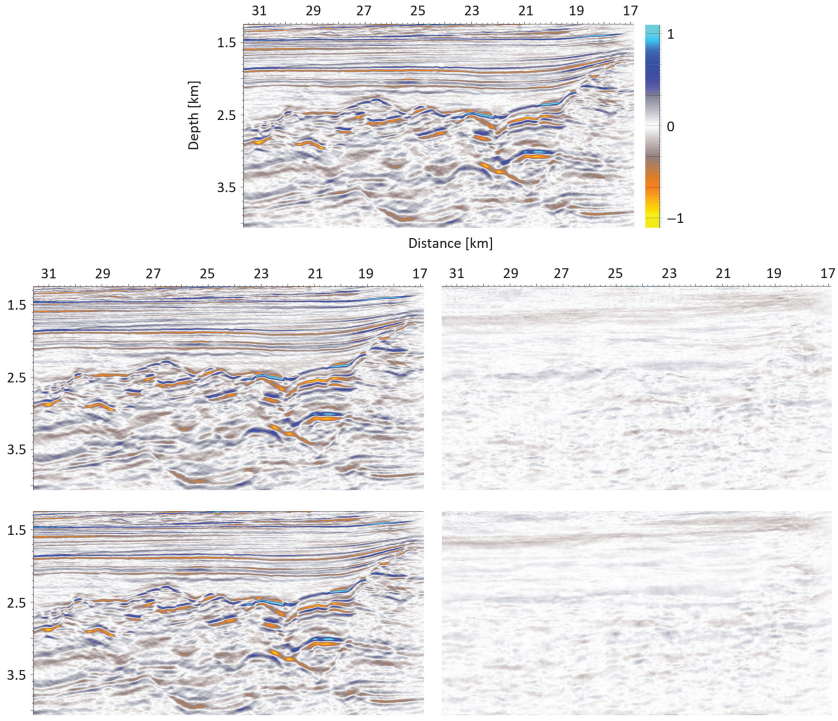
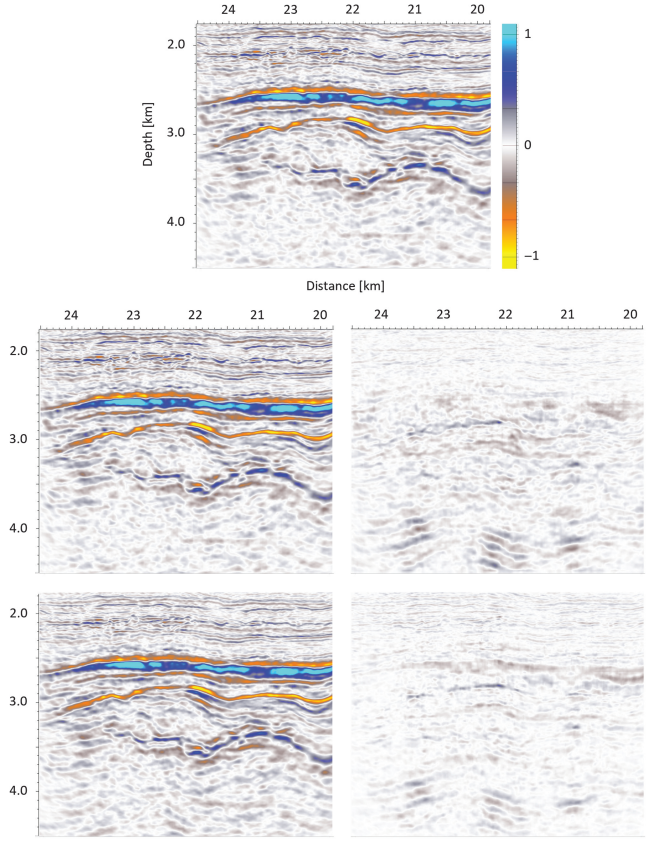


Figure 15. Full-stack of depth angle gathers of a real data set from the Volve data set. First row: migrated raw full-stack section. Second row: full-stack, U-net demultiple (left), and removed multiples (right). Third row: full-stack Radon-based demultiple (left) and removed multiples (right).



determined that for multiple removal (1) the SGD optimizer is a better candidate than Adam, as it leads to more stable, consistent results; (2) the choice of the sampling operations seems to play a minor role and thus, we prefer to keep the model simpler with less demanding up- and down-sampling operators; (3) as for the kernel size, we empirically have found that square- and small-sized kernels do consistently outperform other kernel shapes when applied on real data; (4) both direct and inverse loss functions provide similar results; and finally (5) the depth of the network has a dramatic effect on the performance and consequently, one needs to determine the correct trade-off between network capacity and inference time. Although the results are encouraging, the empirical assessment only represents a subset of the total number of possible hyperparameter configurations. Nonetheless, they are sufficient to decide which hyperparameters play an important role in improving the transferability of features learned from synthetic to real data applications. As demonstrated by Breuer et al. (2020), similar neural network topologies can be used for different gather-to-gather processing steps, such as trim-statics. Hence, our hyperparameter analysis should also be of value for other seismic gather-to-gather approaches based on the U-net architecture.

In general, it is relatively trivial to train a neural network that can yield accurate results on a synthetic data set. However, it is highly challenging to obtain similar performance on unseen real data with potentially very different acquisition, geology, and processing settings. For this reason, producing synthetic data that realistically mimic subsurface events is crucial ongoing research (Durall et al., 2021). During our experimental evaluation, we have iteratively modeled different synthetic training data to investigate the effects on real data. This data-driven methodology has allowed us to generate a concise multiple-oriented data set, with high generalization properties. Instead of focusing on the large-scale periodic relationships between primary and multiple events, in our approach, we use their geometric shapes and localized interactions. Counterintuitively, the proposed approach does not require a global view of the gather to complete the task. As a result, training the model with larger or elongated max-pooling kernels to increase the receptive field size does not enhance performance; instead, it introduces unwanted compression-decompression artifacts on the primary features (Figure 10). Nonetheless, for tasks where a global view of the gather is of critical importance (e.g. approaches using periodic relationships between events), elongated max-pooling kernels might prove beneficial. Moreover, the main objective of our hyperparameter study was the ability of the model to generalize. To that end, we test the intermediate models on numerous data sets and evaluate their performance qualitatively, as opposed to solely benchmarking using quantitative metrics on synthetic testing data. This fact, together with a feature-rich training data set containing primaries and multiples of various frequencies, moveouts, densities, and noise levels, allows us to reliably process data sets of various characteristics.

The model is applicable to both offset and angle gathers in the time and depth domains, using a parameter-free approach. In this way, our approach can expedite interpretation tasks, providing human experts with assistance in managing extensive volumes of real data.

CONCLUSION

In this work, we propose a demultiple model that can be interpreted as an image-to-image transformation system in the category of separation-based multiple removal approaches. Thanks to elaborate hyperparameter analysis using ensemble methods and iterative synthetic training data generation, our approach has proven to generalize well when applied to various synthetic and real field data without the necessity to retrain the model. The events removed by our method and PRT are mostly similar, with occasional advantages for the proposed methodology. This advantage is pronounced in cases where the remnant multiple energy is concentrated in the far stack. Due to its parameter-free nature and independence of the CDP gather domain (i.e., offset, angle, depth, and time), this approach has the potential to drastically reduce the turn-over time for postmigration gather conditioning.

ACKNOWLEDGMENTS

The first and second authors contributed equally to this paper. The authors wish to express their gratitude to the members of the Fraunhofer ITWM DLSeis consortium (http://dlseis.org) for their generous financial support. Additionally, we extend our appreciation to Equinor ASA, Vår Energy ASA, Petoro AS, and ConocoPhillips Skandinavia AS for granting us permission to utilize their Field Data A, and to ExxonMobil for providing the synthetic data set featured in this paper. Furthermore, we acknowledge Equinor and the Volve License partners for making the Volve seismic field data (Field Data B) available under an Equinor Open Data Licence.

DATA AND MATERIALS AVAILABILITY

Data associated with this research are confidential and cannot be released.

REFERENCES

Abma, R., N. Kabir, K. H. Matson, S. Michell, S. A. Shaw, and B. McLain, 2005, Comparisons of adaptive subtraction methods for multiple attenuation: The Leading Edge, **24**, 277–280, doi: 10.1190/1.1895312.

Araujo, A., W. Norris, and J. Sim, 2019, Computing receptive fields of convolutional neural networks: Distill, **4**, e21, doi: 10.23915/distill.00021.

Berkhout, A., 1985, Seismic migration: Imaging of acoustic energy by wave field extrapolation: Elsevier, Developments in Solid Earth Geophysics, 14.

Berryhill, J., and Y. Kim, 1986, Deep-water peg legs and multiples: Emulation and suppression: Geophysics, **51**, 2177–2184, doi: 10.1190/1.1442070.

Breiman, L., 1996, Bagging predictors: Machine Learning, **24**, 123–140, doi: 10.1007/BF00058655.

Breuer, A., N. Ettrich, and P. Habelitz, 2020, Deep learning in seismic processing: Trim statics and demultiple: 90th Annual International Meeting, SEG, Expanded Abstracts, 3199–3203, doi: 10.1190/segam2020-3427887.1.

Bugge, A. J., A. K. Evensen, J. E. Lie, and E. H. Nilsen, 2021, Demonstrating multiple attenuation with model-driven processing using neural networks: The Leading Edge, **40**, 831–836, doi: 10.1190/tle40110831.1.

Carvalho, F., A. B. Weglein, and R. H. Stolt, 1991, Examples of a nonlinear inversion method based on the T matrix of scattering theory: Application to multiple suppression: 61st Annual International Meeting, SEG, Expanded Abstracts, 1319–1322, doi: 10.1190/1.1889114.

Choi, D., C. J. Shallue, Z. Nado, J. Lee, C. J. Maddison, and G. E. Dahl, 2019, On empirical comparisons of optimizers for deep learning: arXiv preprint, doi: 10.48550/arXiv.1910.05446.

Durall, R., V. Tschannen, N. Ettrich, and J. Keuper, 2021, Generative models for the transfer of knowledge in seismic interpretation with deep learning: The Leading Edge, **40**, 534–542, doi: 10.1190/tle40070534.1.

Equinor, 2018, Volve field dataset (Equinor open data license), Data set accessed 16 December 2020 at https://www.equinor.com/energy/volve-data-sharing.

Gal, Y., J. Hron, and A. Kendall, 2017, Concrete dropout: Proceedings of the 31st Conference on Neural Information Processing Systems, 3581–3590.

Gavrikov, P., and J. Keuper, 2022, An empirical investigation of model-to-model distribution shifts in trained convolutional filters: Proceedings of the 35th Conference on Computer Vision and Pattern Recognition, 139–147.

Gisolf, D., and E. Verschuur, 2010, The principles of quantitative acoustical imaging: EAGE Publications bv.

Hampson, D., 1986, Inverse velocity stacking for multiple elimination: 56th Annual International Meeting, SEG, Expanded Abstracts, 422–424, doi: 10.1190/1.1893060.
Hardt, M., B. Recht, and Y. Singer, 2016, Train faster, generalize better: Stability of stochastic gradient descent: Proceedings of the 33rd International Conference on Machine Learning, 1225–1234.
Ibtehaz, N., and M. Rahman, 2020, MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation: Neural Networks, **121**, 74–87, doi: 10.1016/j.neunet.2019.08.025.
Jetley, S., N. Lord, N. Lee, and P. H. S. Torr, 2018, Learn to pay attention: arXiv preprint, doi: 10.48550/arXiv.1804.02391.
Kingma, D. P., and J. Ba, 2014, Adam: A method for stochastic optimization: arXiv preprint, doi: 10.48550/arXiv.1412.6980.
Kostov, C., F. X. de Melo, A. Raj, A. Zarkhidze, A. Cooke, G. Miers, and J. Bacon, 2015, Multiple attenuation for shallow-water surveys: Notes on old challenges and new opportunities: The Leading Edge, **34**, 760–768, doi: 10.1190/tle34070760.1.
Lakshminarayanan, B., A. Pritzel, and C. Blundell, 2017, Simple and scalable predictive uncertainty estimation using deep ensembles: Proceedings of the 31st International Conference on Neural Information Processing Systems, 6402–6413.
Lopez, G. A., and D. J. Verschuur, 2015, Closed-loop surface-related multiple elimination and its application to simultaneous data reconstruction: Geophysics, **80**, no. 6, V189–V199, doi: 10.1190/geo2015-0287.1.
Ma, C., Q. Fu, and A. B. Weglein, 2019, Comparison of the inverse scattering series free-surface multiple elimination (ISS FSME) algorithm with the industry-standard surface-related multiple elimination (SRME): Defining the circumstances in which each method is the appropriate toolbox choice: Geophysics, **84**, no. 5, S459–S478, doi: 10.1190/geo2018-0411.1.
Maddox, W. J., P. Izmailov, T. Garipov, D. P. Vetrov, and A. G. Wilson, 2019, A simple baseline for Bayesian uncertainty in deep learning: Proceedings of the 33rd International Conference on Neural Information Processing Systems, 13153–13164.
Nair, V., and G. E. Hinton, 2010, Rectified linear units improve restricted Boltzmann machines: Proceedings of the 27th International Conference on Machine Learning, 807–814.
Oktay, O., J. Schlemper, L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. Mcdonagh, N. Hammerla, and B. Kainz, 2018, Attention U-Net: Learning where to look for the pancreas: arXiv preprint, doi: 10.48550/arXiv.1804.03999.
Qu, S., E. Verschuur, D. Zhang, and Y. Chen, 2021, Training deep networks with only synthetic data: Deep-learning-based near-offset reconstruction for (closed-loop) surface-related multiple estimation on shallow-water field data: Geophysics, **86**, no. 3, A39–A43, doi: 10.1190/geo2020-0723.1.
Ronneberger, O., P. Fischer, and T. Brox, 2015, U-Net: Convolutional networks for biomedical image segmentation: Proceedings of the 18th Conference on Medical Image Computing and Computer-Assisted Intervention, 234–241.
Sacchi, M. D., and M. Porsani, 1999, Fast high resolution parabolic radon transform: 69th Annual International Meeting, SEG, Expanded Abstracts, 1477–1480, doi: 10.1190/1.1820798.
Sacchi, M. D., and T. J. Ulrych, 1995, High-resolution velocity gathers and offset space reconstruction: Geophysics, **60**, 1169–1177, doi: 10.1190/1.1443845.
Shuey, R. T., 1985, A simplification of the Zoeppritz equations: Geophysics, **50**, 609–614, doi: 10.1190/1.1441936.
Trad, D., T. Ulrych, and M. Sacchi, 2003, Latest views of the sparse radon transform: Geophysics, **68**, 386–399, doi: 10.1190/1.1543224.
Verschuur, D. J., 1991, Surface-related multiple elimination, an inversion approach: Doctoral thesis, TU Delft.
Verschuur, D. J., A. Berkhout, and C. Wapenaar, 1992, Adaptive surface-related multiple elimination: Geophysics, **57**, 1166–1177, doi: 10.1190/1.1443330.
Wang, B., J. Cai, M. Guo, C. Mason, S. Gajawada, and D. Epili, 2011, Post-migration multiple prediction and removal in the depth domain: Geophysics, **76**, no. 5, WB217–WB223, doi: 10.1190/geo2011-0010.1.
Wang, K., T. Hu, S. Wang, and J. Wei, 2022, Seismic multiple suppression based on a deep neural network method for marine data: Geophysics, **87**, no. 4, V341–V365, doi: 10.1190/geo2021-0206.1.
Weglein, A. B., F. V. Araújo, P. M. Carvalho, R. H. Stolt, K. H. Matson, R. T. Coates, D. Corrigan, D. J. Foster, S. A. Shaw, and H. Zhang, 2003, Inverse scattering series and seismic exploration: Inverse Problems, **19**, R27, doi: 10.1088/0266-5611/19/6/R01.
Weglein, A. B., F. A. Gasparotto, P. M. Carvalho, and R. H. Stolt, 1997, An inverse-scattering series method for attenuating multiples in seismic reflection data: Geophysics, **62**, 1975–1989, doi: 10.1190/1.1444298.
Weglein, A. B., S.-Y. Hsu, P. Terenghi, X. Li, and R. H. Stolt, 2011, Multiple attenuation: Recent advances and the road ahead (2011): The Leading Edge, **30**, 864–875, doi: 10.1190/1.3626494.
Wiggins, J. W., 1988, Attenuation of complex water-bottom multiples by wave-equation-based prediction and subtraction: Geophysics, **53**, 1527–1539, doi: 10.1190/1.1442434.
Wilson, A., P. Izmailov, M. Hoffman, Y. Gal, Y. Li, M. Pradier, S. Vikram, A. Foong, S. Lotfi, and S. Farquhar, 2022, Evaluating approximate inference in Bayesian deep learning: Proceedings of the 36th Conference on Neural Information Processing Systems, 113–124.
Wilson, A. C., R. Roelofs, M. Stern, N. Srebro, and B. Recht, 2017, The marginal value of adaptive gradient methods in machine learning: Proceedings of the 31st Conference on Neural Information Processing Systems, 4148–4158.
Zhang, D., and D. J. E. Verschuur, 2021, Closed-loop surface-related multiple estimation with full-wavefield migration-reconstructed near offsets for shallow water: Geophysics, **86**, no. 5, WC21–WC30, doi: 10.1190/geo2020-0879.1.
Zhang, D., D. J. E. Verschuur, and Y. Chen, 2021, Fast local primary-and-multiple orthogonalization for surface-related multiple leakage estimation and extraction: Geophysics, **86**, no. 4, V353–V360, doi: 10.1190/geo2020-0420.1.

Biographies and photographs of the authors are not available.