

Deghosting dual-component streamer data using demigration-based supervised learning

Thomas de Jonge^{1,2}  | Vetle Vinje² | Gordon Poole³ | Peng Zhao² | Einar Iversen¹

¹Department of Earth Science, University of Bergen, Bergen, Norway

²CGG Services Norway AS, Oslo, Norway

³CGG Services, R&D Crompton Way, Manor Royal Estate Crawley, Crawley, West Sussex, UK

Correspondence

Thomas de Jonge, University of Bergen, Department of Earth Science, Allégaten 41, Bergen 5007, Norway; CGG Services Norway AS, Lilleakerveien 6A, Box 43, Lilleaker, NO 0283 Oslo, Norway.
 Email: thomas.jonge@uib.no

Funding information

The Research Council of Norway, Grant/Award Number: 305450; The University of Bergen; CGG Services Norway

Abstract

Ghost reflections from the free surface distort the source signature and generate notches in the seismic amplitude spectrum. For this reason, removing ghost reflections is essential to improve the bandwidth and signal-to-noise ratio of seismic data. We have developed a novel approach that involves training a convolutional neural network to remove source and receiver ghosts from marine dual-component data. High-quality training data is essential for the network to produce accurate predictions on real data. We have used the demigration of a stacked depth-migrated image to create training shot gathers. Demigrated pressure and vertical velocity data are used to train the network. We apply the trained network on real pressure and vertical velocity data with ghosts. The network's output may be either source deghosting and receiver deghosting, or both. We test our method on synthetic Marmousi and real North Sea data with dual-component streamers. The method is compared with conventional dual-component deghosting using the summation of pressure and vertical velocity. Results show that the method can accurately remove the ghosts with only minor errors in synthetic data. Based on a decimation test, the method is less affected by spatially aliased data than a conventional method, which could benefit data with high frequencies and/or large receiver or cable separations. On real data, the results show consistency with conventional deghosting, both within and outside the training area. This indicates that the method is a viable alternative to conventional methods on real data.

KEYWORDS

data processing, modelling, multi-component, noise, seismics, signal processing

INTRODUCTION

A seismic ghost is a reflection of seismic waves from the sea surface. The sea surface pressure-wave reflection coefficient is close to -1 , meaning that the ghost has the opposite polarity with almost the same amplitude as the up-going waves. Towed streamer acquisitions generate three ghosts, one from the source side, one from the receiver side, and finally, a ghost

from both the source and receiver side, as shown in Figure 1. Because of the extra distance the ghosts travel, they arrive at the receiver with a slight time delay relative to the primary. The ghosts elongate and distort the seismic signal and its frequency spectrum. Both the pressure and particle velocity measurements contain ghosts. However, pressure and particle velocity data have the opposite receiver ghost polarity relative to the primary polarity (Figure 1b,c). This is due to the fact

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 CGG Services Norway AS. *Geophysical Prospecting* published by John Wiley & Sons Ltd on behalf of European Association of Geoscientists & Engineers.

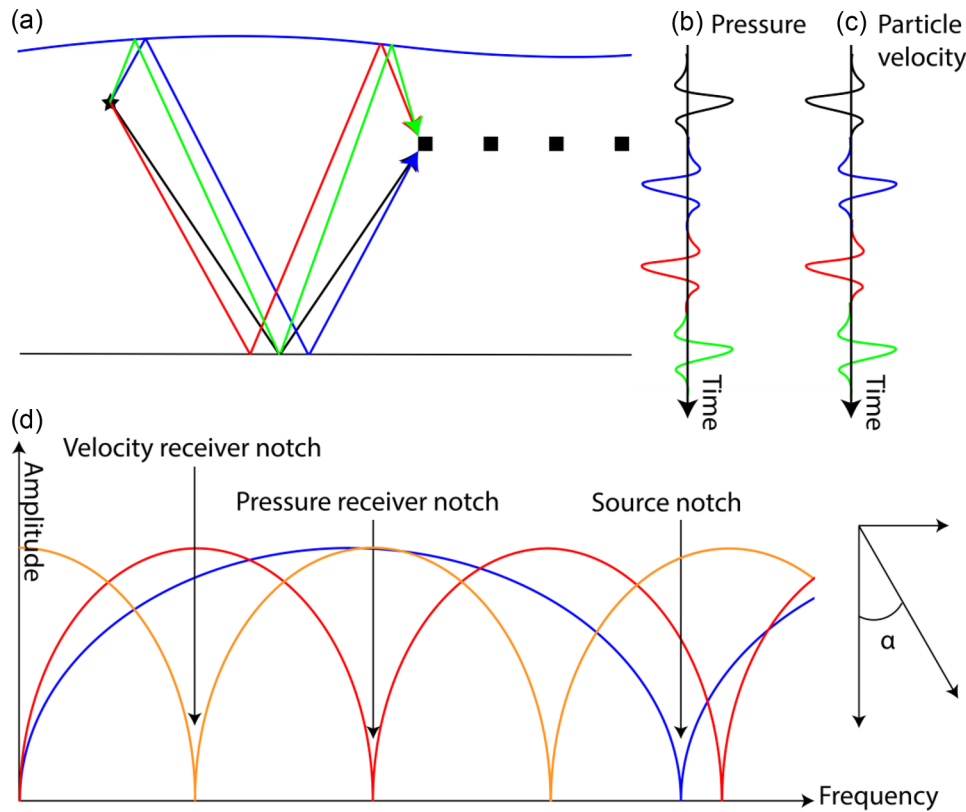


FIGURE 1 (a) Illustration showing four different ray paths from a source to a receiver where the black, blue, red and green ray paths correspond to the primary, source ghost, receiver ghost and source-receiver ghost, respectively. (b) Measurement of pressure. (c) Measurement of particle velocity. (d) The amplitude spectrum illustrating notches for the pressure and velocity data. The receiver ghost arrives later and has a lower notch frequency than the source ghost because it is located deeper than the source. *Source:* Adopted from de Jonge, Vinje, Zhao et al. (2022).

that pressure measurements do not differentiate between up- and down-going wavefields arriving at the receiver, whereas particle velocity is a vector, measuring opposite polarities in up- and down-going wavefields. In the pressure amplitude spectrum, there are ghost notches (Figure 1d) at frequencies given by (Aytun, 1999)

$$f_n^P = \left| n \frac{v_W}{2\Delta z \cos(\alpha)} \right|, \quad n = 0, 1, \dots, \quad (1)$$

where v_W , Δz and α are the water velocity, source or receiver depth and incidence angle (positive downwards, Figure 1), respectively. The amplitude spectrum of the vertical particle velocity (V_z) has receiver notches (Figure 1d) at frequencies given by (Carlson et al., 2007)

$$f_n^V = \left| \frac{v_W}{2\Delta z \cos(\alpha)} \left(\frac{1}{2} + n \right) \right|, \quad n = 0, 1, \dots \quad (2)$$

The receiver notch frequencies are different for hydrophone and particle velocity data because the primary reflection and its receiver ghost have opposite polarities in pressure data but the same polarities in particle velocity data (Figure 1b,c). However, on the source side, both pressure and particle veloc-

ity have the same notch frequency described by Equation (1). This is because the primary reflection and its source ghost are recorded with opposite polarities in both pressure and particle velocity data. Increasing the source or receiver depth results in a larger time separation of the primary and ghost arrivals and, consequently, a lower frequency notch. Ghost notches are problematic because they attenuate some frequencies, which reduce the temporal resolution (Carlson et al., 2007; Hammond, 1962; Schneider et al., 1964). Usually, the ghosts are removed during processing to improve the bandwidth, resolution and signal-to-noise ratio. Deghosting is also beneficial for seismic inversion and geological interpretation (Song et al., 2015).

Historically, the ghost problem in seismic acquisition was solved by localizing the sources and receivers at shallow depths, usually between 5 and 9 m, to push the notch above the usable frequency range. More recently, other acquisition methods were proposed, such as slant streamers (Bearth & Moore, 1989), variable-depth streamers (Soubaras & Dowle, 2010), over-under streamers (Hill et al., 2006) and multi-component streamers (Carlson et al., 2007). Slant and variable-depth streamers aim to attenuate the receiver ghost using the notch-diversity along the streamer. Over-under

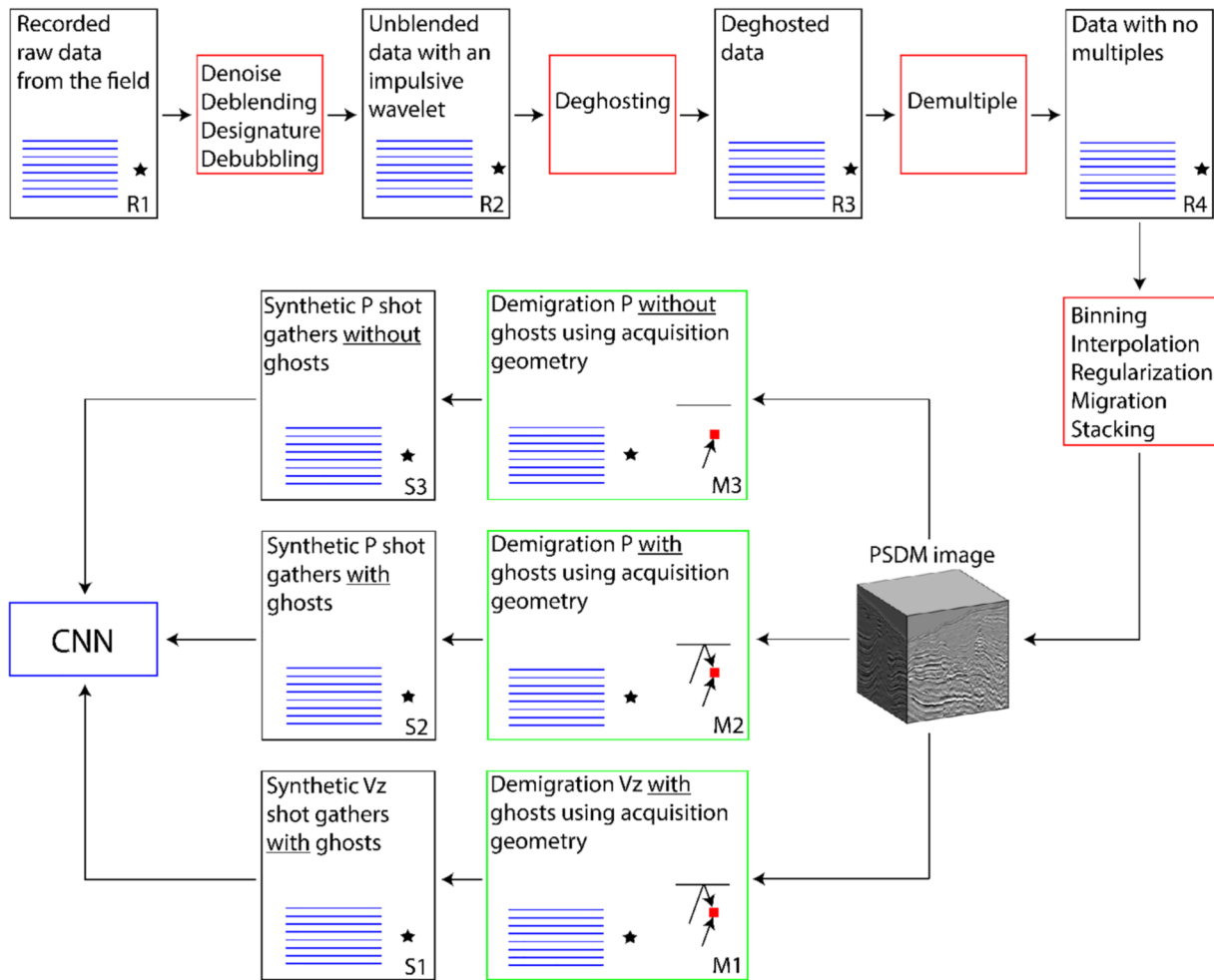


FIGURE 2 Illustrates the deghosting using demigration-based supervised learning (DEGDDEM) workflow with dual-component data used in this paper. *Source:* Adopted from de Jonge, Vinje, Zhao et al. (2022).

streamers have different ghost notches. Consequently, combining the data from these two streamers can help fill the receiver ghost notches. The dual-component streamer contains both hydrophones and geophones measuring pressure and V_z . As shown in Figure 1, ghost notches in the hydrophone data correspond to ghost peaks in the vertical geophone data. This complementary amplitude behaviour makes these data well suited for receiver deghosting, which means instead of towing shallow and relying on hydrophone-only recordings, we can tow deeper and use vertical geophone energy to fill in the hydrophone notches. Towing deeper has the advantage of reducing swell noise (Tenghamn & Dhelie, 2009). In this paper, we will focus on receiver deghosting using dual-component (pressure and V_z) measurements.

A central problem with the dual-component streamers is that V_z data can be noisy at low frequencies to the extent that it can be difficult to incorporate them in conventional deghosting approaches (Mellier & Tellier, 2018; Peng et al., 2014; Poole & Cooper, 2018). Therefore, the low-frequency deghosting relies heavily on the hydrophone data.

For hydrophone-only data, the simplest deghosting method uses the far-field source signature and deterministic deconvolution (Jovanovich et al., 1983), assuming vertical ray paths. Recently, more advanced deghosting methods have been developed, such as frequency–slowness domain inversion (Zhang et al., 2018), $\tau - p$ domain inversion (King & Poole, 2015; Masoomzadeh & Woodburn, 2013; Poole, 2013; Poole & Cooper, 2018; Rickett et al., 2014), inverse Fourier transform deghosting (Amundsen & Zhou, 2013), space-domain deghosting based on Green's functions (Amundsen, Zhou, et al., 2013; Amundsen, Weglein, et al., 2013) or joint deconvolution of a migration and a mirror migration (Soubaras, 2010).

For dual-component data, the conventional approach is to sum the pressure and V_z data, after an obliquity correction on the V_z data to correct for the fact that it records particle velocity only in the vertical direction (Tenghamn & Dhelie, 2009). As mentioned above, low-frequency noise on V_z data can contaminate this sum, so normally, only pressure data is used in the deghosting of the lowest frequencies. As opposed to

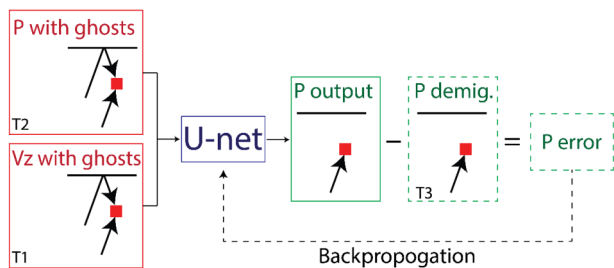


FIGURE 3 Pressure and V_z with ghosts are the input of the network. Pressure without ghosts is the output. The demigrated pressure data without ghosts is used to calculate the deghosting error and update the weight of the network using backpropagation.

the pressure and V_z ($P-V_z$) sum, a hydrophone-only deghosting requires knowledge of the water-surface reflectivity and source and receiver locations. Note, however, that some methods still use pressure and V_z data also for the low frequencies but avoid contamination of V_z noise, such as Telling and Grion (2022), who used an inversion-based approach with a hybrid operator in frequency and space, and Poole and Cooper (2018), who used an inversion-based approach in the $\tau-p$ domain. However, like many multi-component deghosting methods, both approaches use a weighting scheme to reduce the influence of the V_z data for lower frequencies.

Machine learning and, more specifically, artificial neural networks have recently been used for various processing steps such as swell noise attenuation (Zhao et al., 2019), debubbling (de Jonge, Vinje, Poole, et al., 2022), seismic inference noise attenuation (Sun et al., 2019; Sun & Hou, 2022) and interpolation (Greiner et al., 2019; Hlebnikov et al., 2022). Several papers have also described the use of neural networks for deghosting (Almuteri & Sava, 2021; de Jonge, Vinje, Zhao, et al., 2022; Peng et al., 2021; Vrolijk & Blacquiere, 2020). However, none of these methods used multi-component data for deghosting. de Jonge, Vinje, Zhao et al. (2022) used demigration to create realistic pressure data with and without ghosts to train a convolutional neural network. The method was called deghosting using demigration-based supervised learning (DEGDDEM) and used a neural network to deghost real seismic pressure data.

This paper aims to expand the DEGDDEM method to also include V_z data during training and prediction in contrast to the paper by de Jonge, Vinje, Zhao et al. (2022). Most importantly, we want to improve deghosting at low frequencies exhibiting substantial V_z noise. However, this requires training data with both pressure and V_z generated by demigration. Below we describe how we produce V_z data using demigration and train the network using pressure and V_z data. Dual-component DEGDDEM is tested on synthetic Marmousi data and real data from the Northern North Sea. We compare our results with conventional dual-component deghosting as used in the seismic industry. In this paper, the term ‘con-

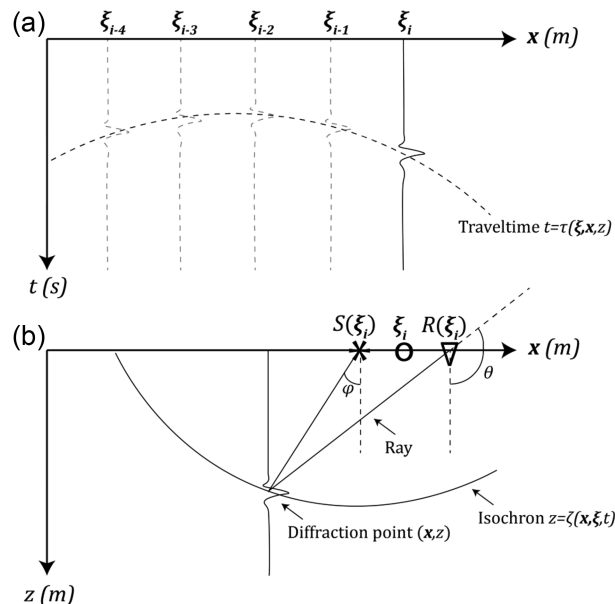


FIGURE 4 An illustration showing the basic concept of constant-offset migration and demigration. The ray paths and two-way traveltime are calculated from S and R to all depth points. Migration is a mapping from the data domain (a) to the depth domain (b), and demigration is the reverse process.

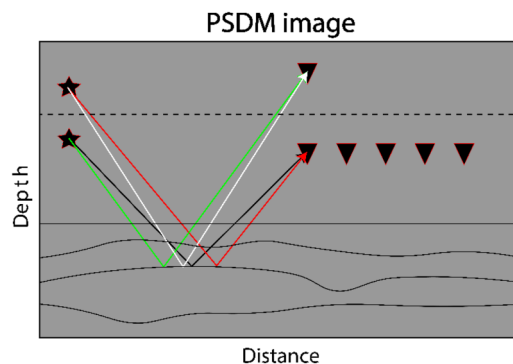


FIGURE 5 Illustration of how the source, receiver and mirror positions are used to create four different datasets. Equations (9) or (10) and these four datasets can be used to create data with source and receiver ghosts. *Source:* Adopted from de Jonge, Vinje, Zhao et al. (2022).

ventional deghosting’ refers to $\tau-p$ domain inversion-based pressure-only deghosting at low frequencies and $P-V_z$ sum at high frequencies.

METHODOLOGY

Convolutional neural networks

This paper uses a convolutional neural network (CNN) with a U-net structure (Ronneberger et al., 2015). A CNN uses

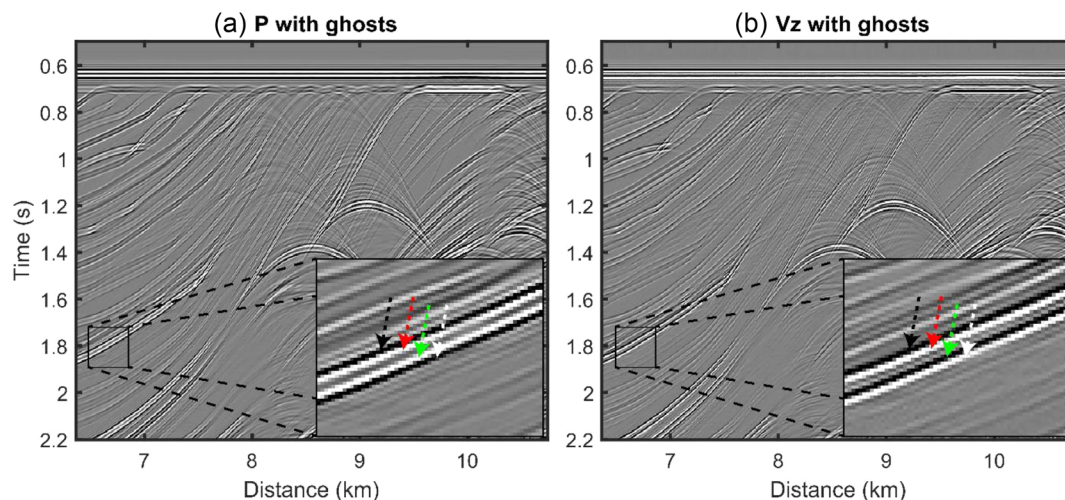


FIGURE 6 Common-channel gathers showing (a) demigration of pressure data with ghosts using Equation (9) and (b) demigration of V_z data with ghosts using Equation (10). A zoom is used in both figures focusing on a single event where arrows highlight the primary (black), source ghost (red), receiver ghost (green) and source-receiver ghost (white). The V_z data is scaled with $-\rho v_w$ for illustration purposes.

convolutions in contrast to a dense neural network that uses matrix multiplications. As a result, the CNN has sparse interactions between neurons in the network that make the network more efficient. The kernel size determines the number of interactions of neurons from one layer to the next. The U-net structure uses an encoder to down-sample the number of pixels and a decoder to up-sample the number of pixels. The U-net also contains skip connections that copy feature maps from one layer to another. Pooling operations down-sample the size of the feature maps by using some functions to summarize subregions (e.g. max. pooling and average pooling). Transposed convolution up-samples the size of the feature maps and is an operation that goes in the opposite direction of a standard convolution. Dumoulin and Visin (2018) showed examples of transposed convolution and pooling functions. In contrast to CNNs without down-sampling or up-sampling, the U-net structure is beneficial because of the increased receptive field due to the down- and up-sampling (Lucas et al., 2018). In addition, down- and up-sampling makes it more efficient.

A few other papers have also used the U-net structure for pressure-only deghosting with high-quality results (de Jonge, Vinje, Zhao, et al., 2022; Peng et al., 2021; Vrolijk & Blacquière, 2021). In our case, we are confident that a U-net structure will also give good results on dual-component data. The exact structure used in our paper is also shown in the paper by de Jonge, Vinje, Poole et al. (2022) and de Jonge, Vinje, Zhao et al. (2022), which is similar to the original U-net by Ronneberger et al. (2015) but with more down- and up-sampling layers. That said, the goal of this paper is not to find the optimal network structure but to develop a new CNN based training method for deghosting dual-component data.

Deghosting using demigration-based supervised learning workflow

This research is an extension of the work done by de Jonge, Vinje, Zhao et al. (2022) on pressure-only deghosting. In the following section, we review the basic features of that method. The approach is called deghosting using demigration-based supervised learning (DEGD) and involves generating synthetic shot gathers using Kirchhoff demigration (Santos et al., 2000a) from a seismic image. We create the seismic image in our workflow by migrating seismic data, usually with pre-stack depth migration (PSDM). Kirchhoff migration and demigration are reverse processes (Santos et al., 2000b). As a result, the demigrated data will resemble the seismic data before migration. This is described in more detail in the ‘Modelling demigrated dual-component data’ section.

Using demigrated data to train the network has two main advantages first, the training data resembles the recorded seismic data, and second, it is possible to generate training data with no ghosts. The workflow is shown in Figure 2 and is a modification of the workflow shown in de Jonge, Vinje, Zhao et al. (2022) as it includes dual-component data. After recording the data (Figure 2 R1), it usually goes through a standard processing flow with multiple steps. These processing steps include denoising, deblending, debubbling, designation, deghosting and demultiple, followed by binning, interpolation, regularization, migration and stacking to produce a PSDM image. We use the PSDM image and a smooth PSDM velocity model for the demigration (Figure 2 M1, M2 and M3) to create shot gathers containing pressure (Figure 2 S2 and S3) and vertical particle velocity (V_z) (Figure 2 S1) data with and without ghosts. We train a CNN using these shot

gathers. Figure 3 illustrates how we train the network. Pressure and V_z data with ghosts (Figure 3 T1 and T2) are the input to the network. Pressure data without ghosts is the desired output of the network. During training, demigrated pressure data without ghosts (Figure 3 T3) is used to update the weights of the network using back-propagation. Thousands to tens of thousands of shot gathers are used to train the network. This trained network is later used on real pressure and V_z data with ghosts to predict pressure data without ghosts.

The shot and receiver positions from the real marine acquisition were used when creating the training data. It is possible to apply the network (to predict deghosted data) to data from the same area from which the training data was created. Alternatively, it is possible to use a small portion of an area to create training data and apply the network to the entire area. de Jonge, Vinje, Zhao et al. (2022) showed on synthetic and real data that the deghosting quality is still high even if the training and prediction areas are not the same. They also showed that the deghosting required to create the PSDM image for demigration does not have to be optimal for the network to perform well. In addition, DEGDEM is quite robust to errors in sea-surface reflection coefficients and swell waves. In theory, it is possible to train a neural network on data from one survey geometry and apply it to another, but de Jonge, Vinje, Zhao et al. (2022) did not explore this. It is believed that similar geology is an advantage when applying the network to another area.

Modelling demigrated dual-component data

Kirchhoff demigration was used to produce pressure data for training. Kirchhoff demigration can produce synthesized seismograms shown by Santos et al. (2000a). True-amplitude Kirchhoff demigration and true-amplitude Kirchhoff migration are considered reverse processes (Santos et al., 2000b). The Kirchhoff constant-offset migration integral for any migration domain point (\mathbf{x}, z) can be represented by (Bleistein, 1987; Santos et al., 2000a)

$$\text{IM}(\mathbf{x}, z) = -\frac{1}{2\pi} \iint d^2\xi W_M(\xi, \mathbf{x}, z) \frac{\partial D(\xi, t)}{\partial t} \Big|_{t=\tau(\xi, \mathbf{x}, z)}, \quad (3)$$

where $\text{IM}(\mathbf{x}, z)$ is the migrated data, $W_M(\xi, \mathbf{x}, z)$ is the true-amplitude migration weight function, and $D(\xi, t)$ is a trace in the data domain for a given surface location $\xi = (\xi_x, \xi_y)$. The two-way traveltimes $\tau(\xi, \mathbf{x}, z)$, shown in Figure 4, is calculated by ray tracing from constant-offset sources $S(\xi)$ and receivers $R(\xi)$ down to a fixed migration domain point (\mathbf{x}, z) . Equation (3) integrates traces along $\tau(\xi, \mathbf{x}, z)$ in the data domain. Figure 4 shows how we integrate data in the data domain and map it to a point (\mathbf{x}, z) in the migration domain (Bleistein et al., 2001; Schleicher et al., 2007). Kirchhoff demigration

involves a weighted mapping process from the depth migrated image to the seismic data. The Kirchhoff demigration integral is (adopted from Santos et al., 2000a; Tygel et al., 1996) as follows:

$$P(\xi, t) = \frac{1}{2\pi} \iint d^2\mathbf{x} W_D(\mathbf{x}, \xi, t) \frac{\partial \text{IM}(\mathbf{x}, z)}{\partial z} \Big|_{z=\zeta(\mathbf{x}, \xi, t)}, \quad (4)$$

where $P(\xi, t)$ is the demigrated data, $W_D(\mathbf{x}, \xi, t)$ is the true-amplitude data weight function, $\text{IM}(\mathbf{x}, z)$ is data in the migrated domain, and $\zeta(\mathbf{x}, \xi, t)$ is the depth of the lateral position \mathbf{x} in the data domain corresponding to the time point t of the trace in location ξ in the data domain. For a constant t , $\zeta(\mathbf{x}, \xi, t)$ forms an isochron surface in the depth domain as shown in Figure 4.

We now turn to the problem of demigration to V_z data. We have not seen anyone attempting to do this in the literature. As mentioned above, the demigration in Equation (4) integrates the migrated data along isochron surfaces and maps to a point (ξ, t) in the data domain. The output of Equation (4) is pressure data as the input data to migration, shown in Equation (3), is pressure data and these are reverse functions. This means that each of the traces in the integrand in Equation (4) is also pressure data. So we will need to scale the pressure data in the integrand appropriately before integrating to convert from pressure to V_z . Tenghamn and Dhelie (2009) defined the obliquity filter as

$$F(\omega, k_z) = \frac{\rho\omega}{k_z}, \quad (5)$$

where ρ is the water density, ω is the angular frequency, and k_z is the vertical wavenumber. The obliquity filter is used to scale V_z data such that the up-going wavefields of pressure and V_z have equal magnitude and the opposite polarity to the down-going wavefield. This filter is used before pressure, and V_z data are summed together to remove the receiver ghost. More information about this filter is given in the next section. We want to use the inverse obliquity filter, $1/F$, to convert pressure to V_z . The vertical wavenumber, k_z , can be expressed as

$$k_z = \frac{\omega \cos(\theta)}{v_W}, \quad (6)$$

where θ is the incidence angle at the receiver (Figure 4), and v_W is the water velocity. As can be seen from Figure 4, the incidence angle θ is π for a wave travelling vertically upwards. If we insert Equation (6) into the inverse of Equation (5), we obtain

$$\frac{1}{F} = \frac{\cos(\theta)}{\rho v_W}. \quad (7)$$

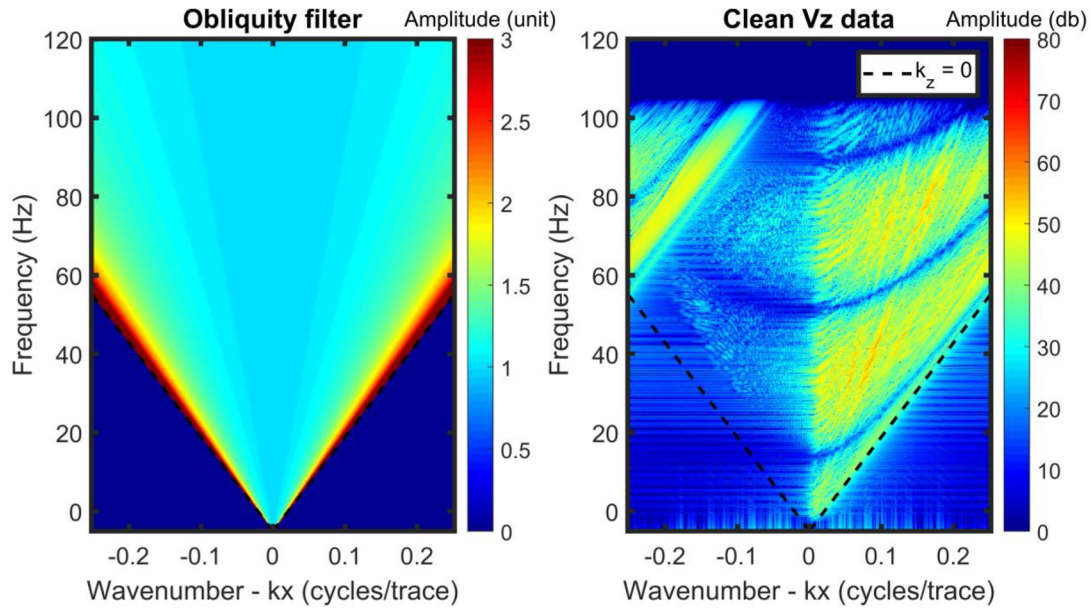


FIGURE 7 An example of an obliquity filter (left) and a finite-difference (FD)-modelled shot gather from the Marmousi model (right) in the frequency–wavenumber (FK) amplitude domain. The black stippled line indicates when $k_z = 0$.

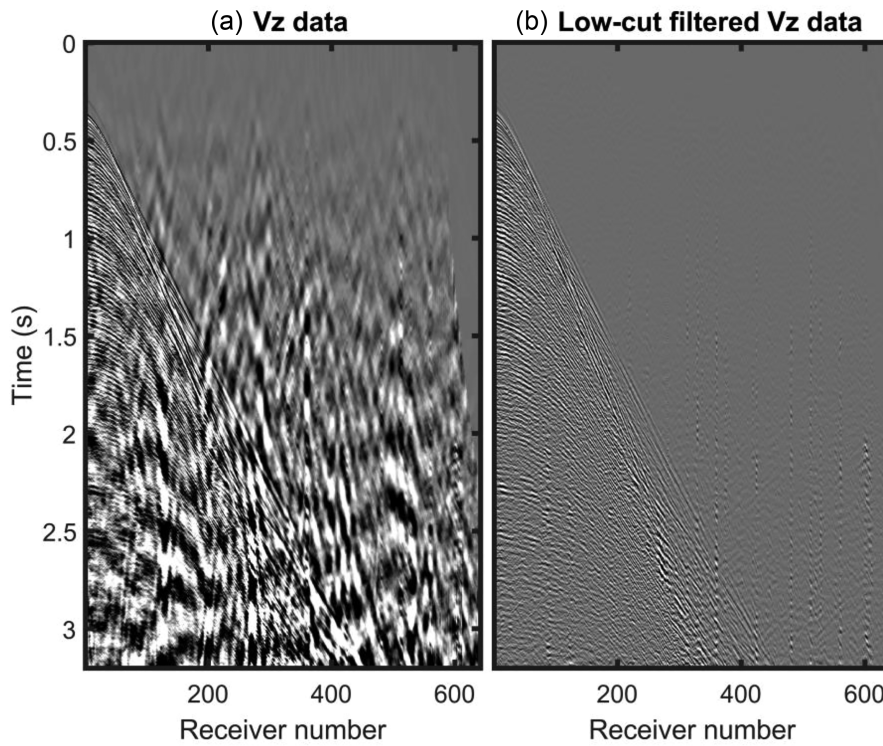


FIGURE 8 (a) An example of mechanical V_z noise on a real shot gather. (b) The same V_z shot gather, but with a low-cut filter (20 Hz) applied.

We include the inverse obliquity filter from Equation (4) into Equation (7) to have V_z as the output of demigration:

$$V_z(\xi, t) = \frac{1}{2\pi} \iint d^2\mathbf{x} W_D(\mathbf{x}, \xi, t) \frac{\cos(\theta)}{v_W \rho} \frac{\partial IM(\mathbf{x}, z)}{\partial z}$$

$$|_z = \zeta(\mathbf{x}, \xi, t). \quad (8)$$

In the further derivation of the demigrations for machine learning described below, we use Equations (4) and (8) to generate demigrated pressure and V_z data, respectively.

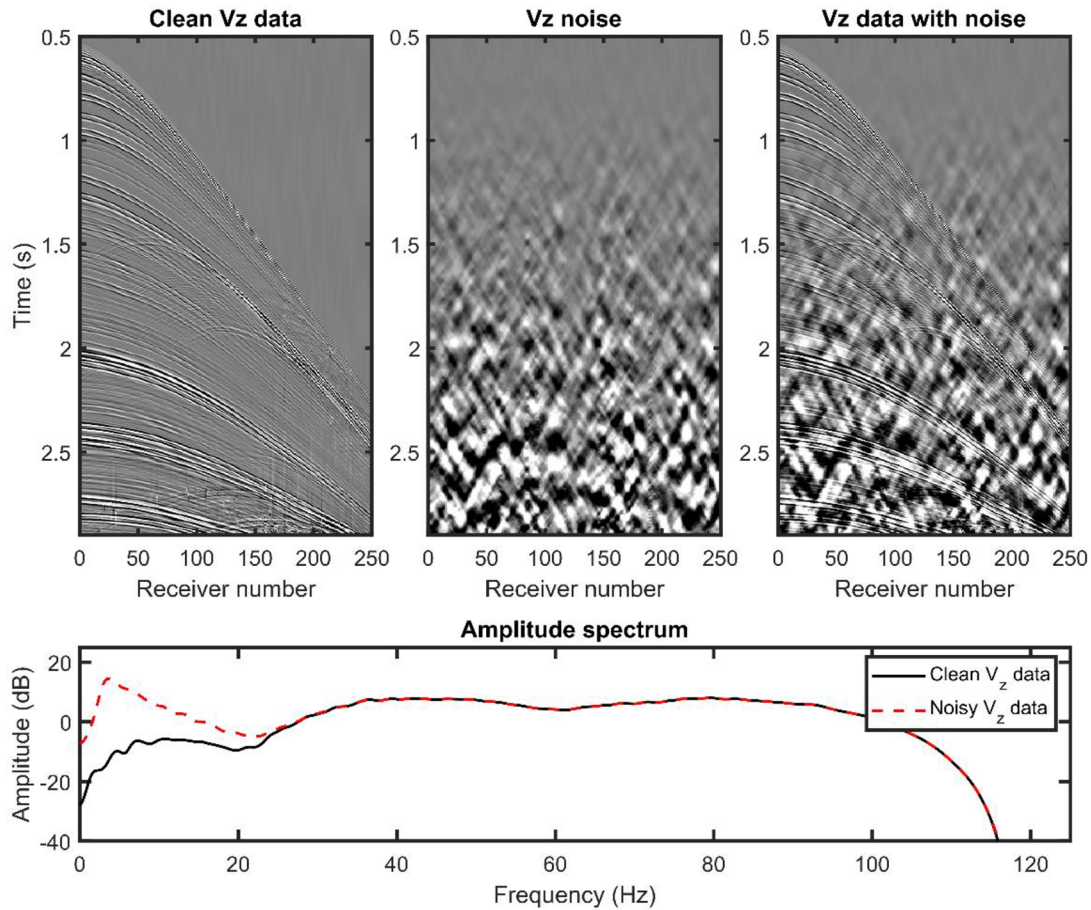


FIGURE 9 Illustration of how real V_z noise and clean demigrated V_z data are combined to create demigrated noisy V_z data. The amplitude spectrum shows that most of the added V_z noise can be found below ~ 30 Hz.

As we use an angle-muted and stacked PSDM image, we cannot get back exactly to the original seismic data before migration when we apply demigration. However, the demigrated training data looks realistic, which is beneficial when training the neural network. We also need a velocity model, which generally would be the same model used for migration. Figure 5 illustrates how we create four datasets, which can be combined to generate pressure data with ghosts using the following equation (de Jonge, Vinje, Zhao, et al., 2022):

$$D_i^P(x_j, t_k) = P_i(x_j, t_k) + rP_i^{\text{SG}}(x_j, t_k) + rP_i^{\text{RG}}(x_j, t_k) + r^2P_i^{\text{SRG}}(x_j, t_k), \quad (9)$$

where $P_i(x_j, t_k)$ is the demigrated data, shown in Equation (4), using the actual source and receiver positions, $P_i^{\text{SG}}(x_j, t_k)$ is the data using the mirror source location, $P_i^{\text{RG}}(x_j, t_k)$ is the data using the mirror receiver location, and $P_i^{\text{SRG}}(x_j, t_k)$ is the data using both the mirror source and receiver locations. Here, x_j is the offset, t_k is the time, and r is the sea surface reflection coefficient, usually close to -1 . The source and receiver mirror locations are the mirror positions above the

sea surface shown in Figure 5. To produce the V_z data with ghosts, we modify Equation (9) using V_z demigrated data, shown in Equation (8) and reversing the polarity of the V_z receiver ghost:

$$D_i^{V_z}(x_j, t_k) = V_{z,i}(x_j, t_k) + rV_{z,i}^{\text{SG}}(x_j, t_k) - rV_{z,i}^{\text{RG}}(x_j, t_k) - r^2V_{z,i}^{\text{SRG}}(x_j, t_k). \quad (10)$$

We scale V_z data with ghosts by ρv_w before using it as training and prediction data. This scale is used such that pressure and V_z have the same amplitude range. Figure 6 shows the demigrated pressure and V_z data in a common-channel gather.

Pressure and V_z summation

The V_z data with ghosts will have a different receiver and source-receiver ghost polarity to the pressure data. Conventionally, pressure and V_z data are summed ($P-V_z$ sum) together to get the up-going pressure using the following equation (Tenghamn & Dhelie, 2009):

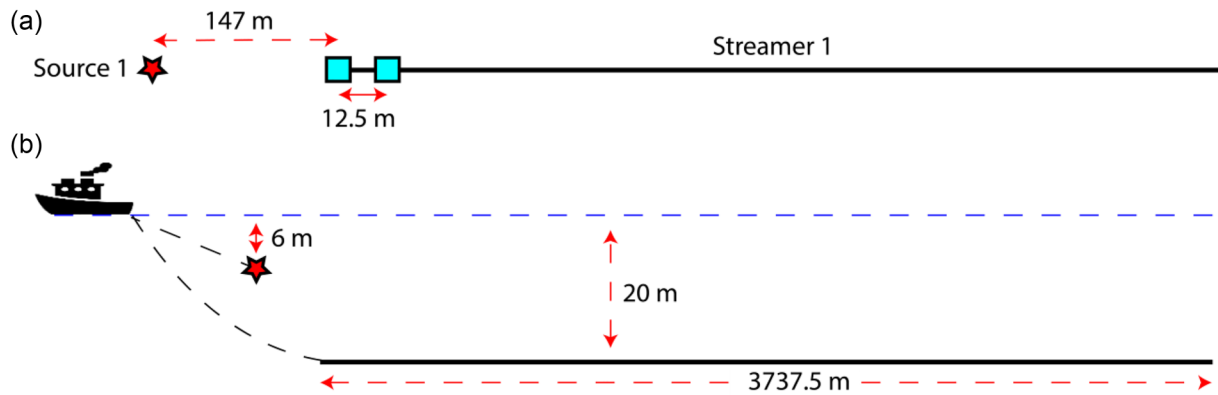


FIGURE 10 Illustration showing the acquisition geometry used to create the synthetic finite-difference (FD) data from above the ship (a) and from the side of the ship (b).

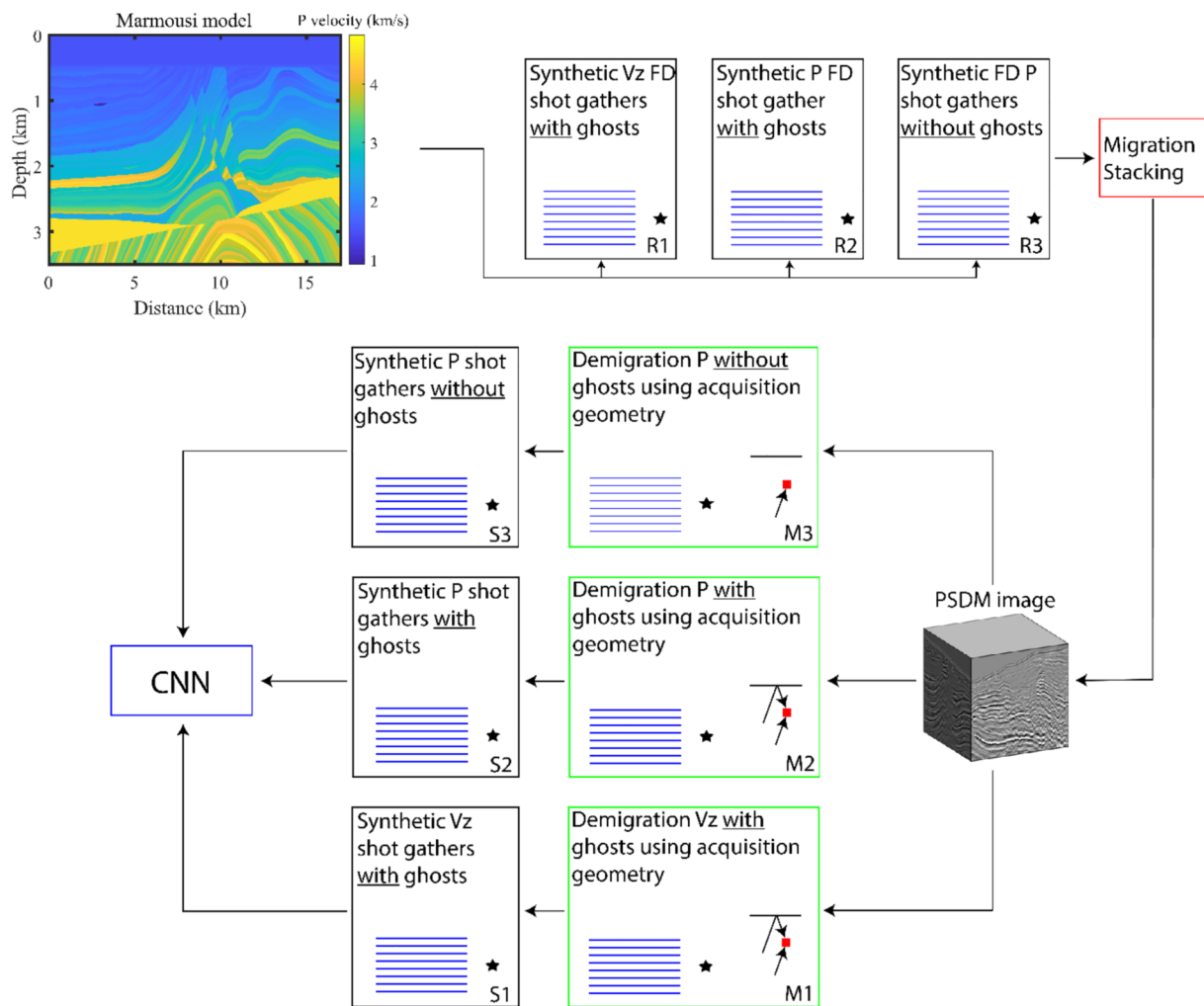


FIGURE 11 Illustrates the deghosting using demigration-based supervised learning (DEGDDEM) workflow with dual-component data used on synthetic data from the Marmousi model. *Source:* Adopted from de Jonge, Vinje, Zhao et al. (2022).

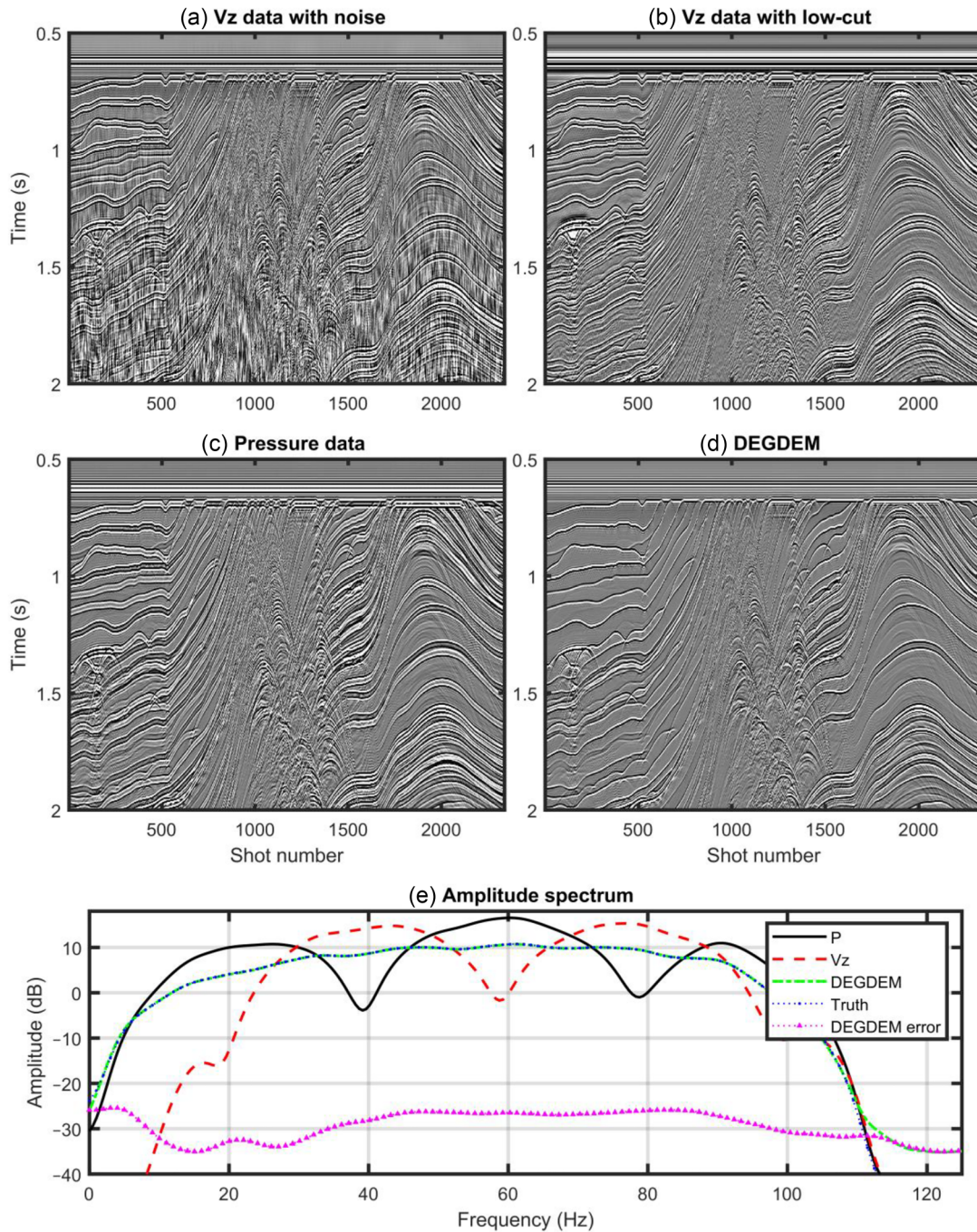


FIGURE 12 Receiver deghosting in common-channel gather (receiver 10): (a) V_z data with ‘raw’ noise and ghosts; (b) low-cut (20 Hz) filtered V_z data with ghosts; (c) pressure data with ghosts; (d) deghosting using demigration-based supervised learning (DEGDEM) using both pressure and low-cut filtered V_z data as input; (e) amplitude spectrum of pressure with ghosts, low-cut filtered V_z data with ghosts, DEGDEM, the ground-truth and DEGDEM error.

$$P_{UP} = \frac{1}{2} (P - FV_z), \quad (11)$$

where the obliquity or scaling filter, F , is given by Equation (5). We have chosen a polarity convention for the V_z data such that the up-going wavefields of pressure and V_z data have an opposite polarity (Figure 1) which is why we use a negative sign in Equation (11) to remove the down-going

wavefield. The vertical wavenumber, k_z , in Equation (5), is estimated from the in-line wavenumber, k_x , using the following equation:

$$k_z = \sqrt{\left(\frac{\omega}{v_w}\right)^2 - k_x^2}. \quad (12)$$

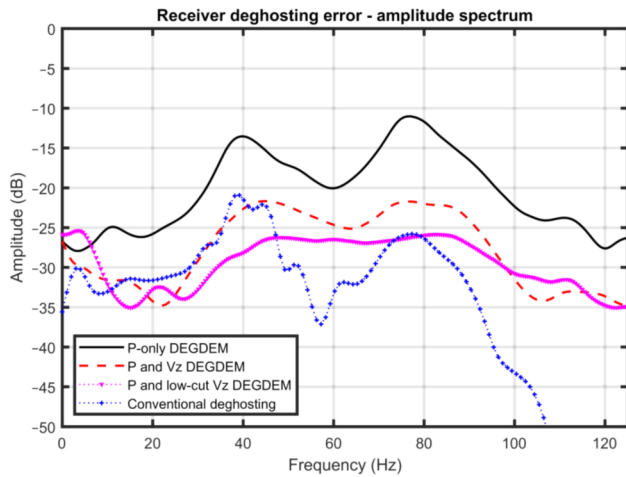


FIGURE 13 Amplitude spectrum of the receiver deghosting error (common-channel gather– receiver 10). Four different results are shown: (1) pressure-only deghosting using demigration-based supervised learning (DEGDEM), (2) pressure and ‘raw’ V_z DEGDEM, (3) pressure and low-cut filtered V_z DEGDEM and (4) conventional deghosting.

The in-line wavenumber, k_x , is usually sampled by receivers every 12.5 m along the seismic streamer. An example of a normalized obliquity filter and a finite-difference-modelled shot gather from the Marmousi model in the frequency–wavenumber amplitude domain are shown in Figure 7. As we approach $k_z = 0$, meaning horizontal travelling waves, the obliquity filter amplitude tends to infinity. Therefore, it is necessary to mute the filter before the amplitude becomes too big. The V_z data shown in Figure 7 becomes aliased as we go beyond the Nyquist wavenumber, and we observe wavenumber wrap-around due to spatial aliasing. As a result, scaling aliased data with the obliquity filter will not be correct.

It is also possible to create a 3D obliquity filter, $F(\omega, k_x, k_y)$. However, the cable separation is often large compared to the in-line receiver separation, which results in sparse data in the crossline direction. Too sparse data will lead to aliasing in the wavenumber domain in the crossline direction.

A problem with V_z data is the low-frequency mechanical tow noise measured during acquisition (Carlson et al., 2007; Tengeham & Dhelie, 2009). Figure 8 shows an example of V_z noise on a shot gather. On raw data, the amplitude of the noise will not increase with time. However, as we have used a t^2 gain correction, it appears to increase in this figure. A low-cut filter is often used (e.g. at 20 Hz) on the V_z data to avoid propagating noise onto the $P-V_z$ sum. As a result, we might not use V_z data below a specific frequency (e.g. 20 Hz). Conventionally, we use a hydrophone-only deghosting method below a specific frequency. In this paper, we use a $\tau - p$ inversion-based deghosting method by Poole (2013) for

the low frequencies and $P-V_z$ sum for the highs. In this paper, this is called conventional deghosting.

Using V_z noise in training data

In our training process, V_z noise is essential because we expect it, or part of it, to be present in the real data. To find suitable V_z noise for our training, we directly use part of the real V_z data containing low-frequency V_z noise but no signal. This signal-free data can be data at the end of the sail line where no source is fired. Alternatively, we can use data before the first arrival on the shot gathers. The low-frequency noise is then added to the V_z training data, as shown in Figure 9. It is clear from Figure 9 that the amplitude of the noise decreases with increasing frequency and is almost absent above 30 Hz.

RESULTS

This section looks at both synthetic data and real data. The synthetic data will verify if the deghosting using demigration-based supervised learning (DEGDEM) method works and understand the problems and advantages under controlled circumstances. Real data can show if the method gives good results on one example from the Northern Viking Graben (NVG) area in the North Sea.

Synthetic data

We used the P-wave velocities and densities of the elastic Marmousi2 model (Martin et al., 2006) and acoustic finite-difference (FD) to model input data. Figure 10 shows the survey configuration used. The shot and receiver spacings are 6.25 and 12.5 m, respectively. The offset to the first receiver is 147 m. The offset to the last receiver is 3884.5 m. The receiver number increases as we move away from the vessel. We modified the workflow in Figure 2 for our synthetic tests (shown in Figure 11). The FD-modelled data without ghosts was migrated for different offset classes. These pre-stack depth migration (PSDM) offset classes were then stacked together to create one PSDM image. This PSDM image, a smooth velocity model (shown in de Jonge, Vinje, Zhao, et al., 2022), and Kirchhoff demigration create shot gathers without ghosts and with ghosts for pressure (Figure 11 S2 and S3) and vertical particle velocity (V_z) (Figure 11 S1) data. We trained a convolutional neural network (CNN) with these shot gathers and use the CNN on FD data with ghosts (Figure 11 R1 and R2). As mentioned in the ‘Methodology’ section, the training V_z data (Figure 11 S1) contained mechanical V_z noise from real data. The FD V_z data (Figure 11 R1) also contains real V_z noise. However, we use V_z noise extracted from different parts of

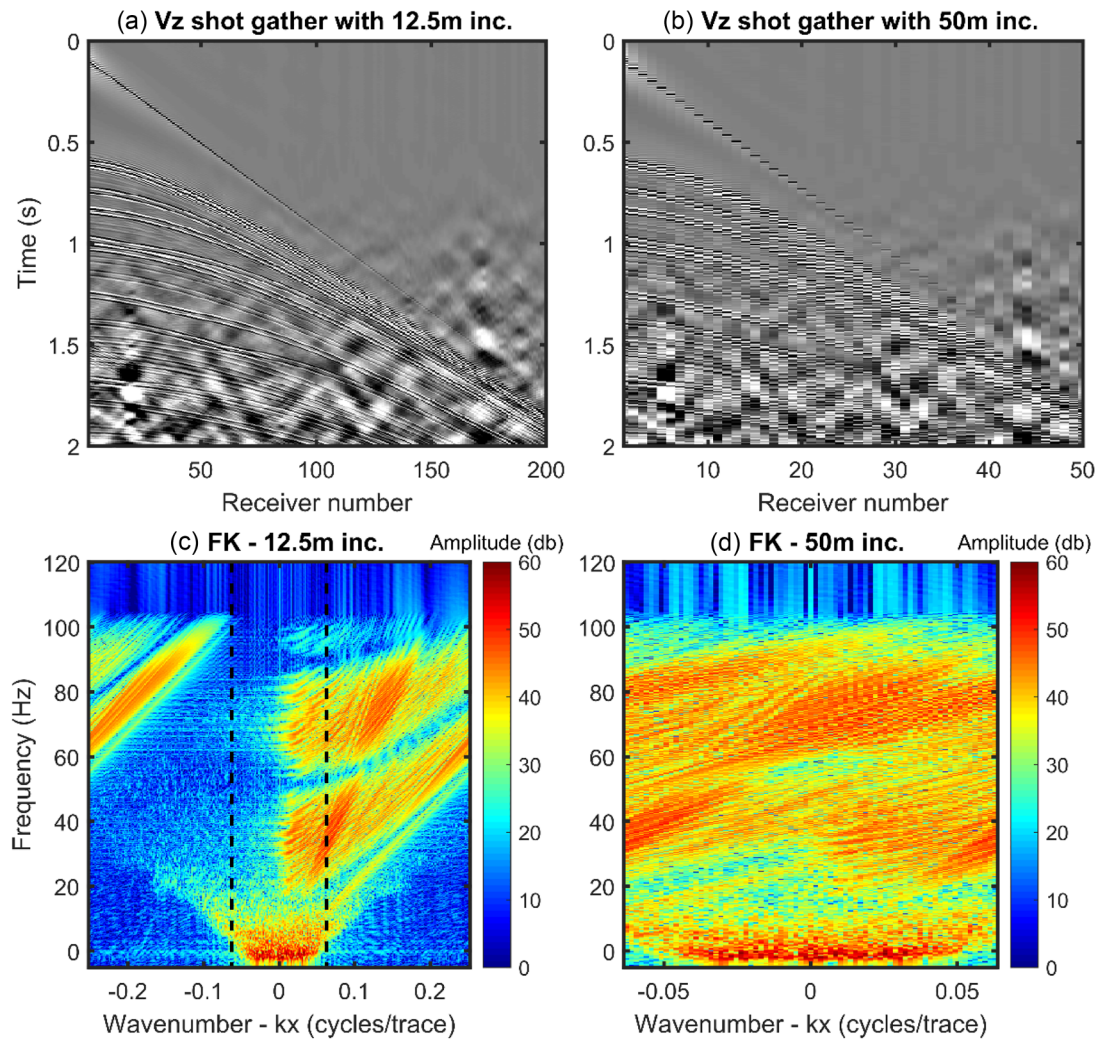


FIGURE 14 (a) and (b) V_z shot gathers with 12.5 and 50 m receiver separation, respectively. (c) and (d) Frequency–wavenumber (FK) amplitude spectra of the shot gathers for 12.5 and 50 m receiver separation, respectively.

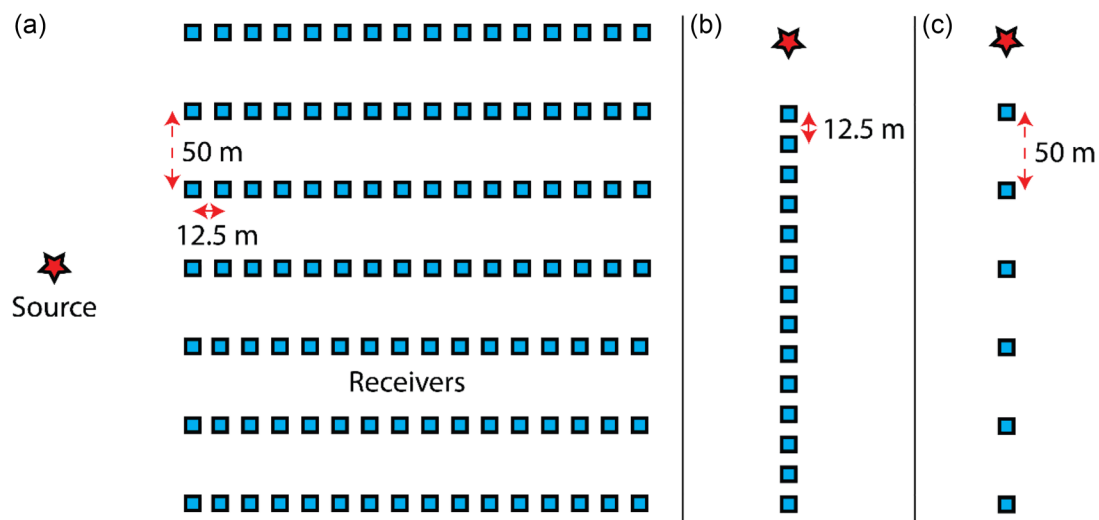


FIGURE 15 (a) Illustration of typical receiver positions in a 3D marine seismic acquisition. (b) A 2D shot gather with 12.5 m receiver separation. (c) A 2D shot gather with 50 m receiver separation.

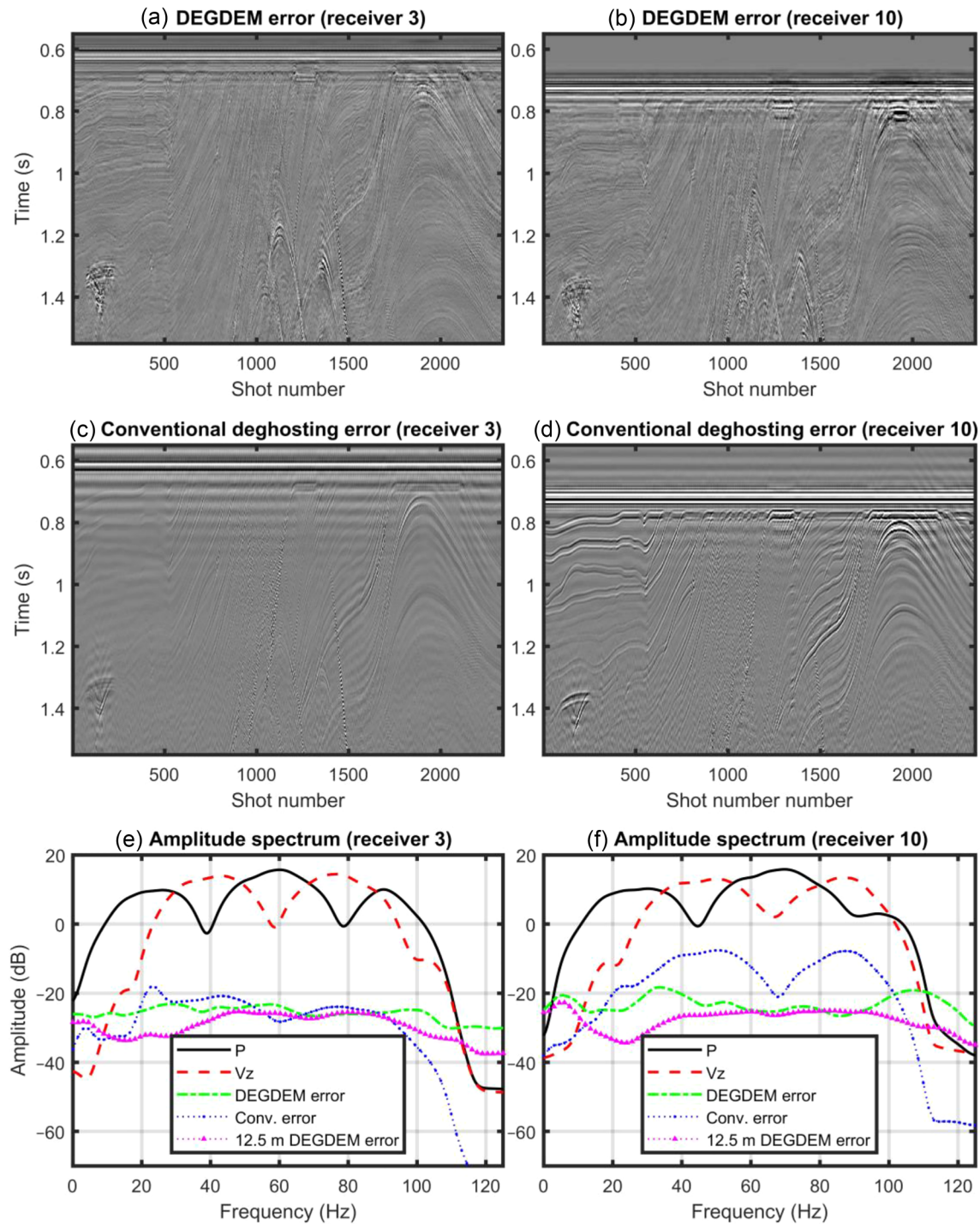


FIGURE 16 Deghosting error in common-channel gathers: (a) deghosting using demigration-based supervised learning (DEGDEM) error for receiver 3; (b) DEGDEM error for receiver 10; (c) conventional deghosting error for receiver 3; (d) conventional deghosting error for receiver 10; (e) and (f) amplitude spectrum for receivers 3 and 10, respectively, showing pressure with ghosts, V_z with ghosts, DEGDEM error, conventional deghosting error and 12.5 m DEGDEM error.

the survey for the training and prediction data. Using the same noise in training and prediction could give a biased result.

We test two different training approaches: (1) Training data contains pressure data and ‘raw’ V_z data. We then train a neural network using this training data. After training, we

test the network on FD modelled data that contains pressure data and ‘raw’ V_z data. (2) Training data contains pressure data and V_z data with a low-cut filter at 20 Hz to remove most of the V_z noise. We then train a neural network using these training data. After training, we test the network on FD

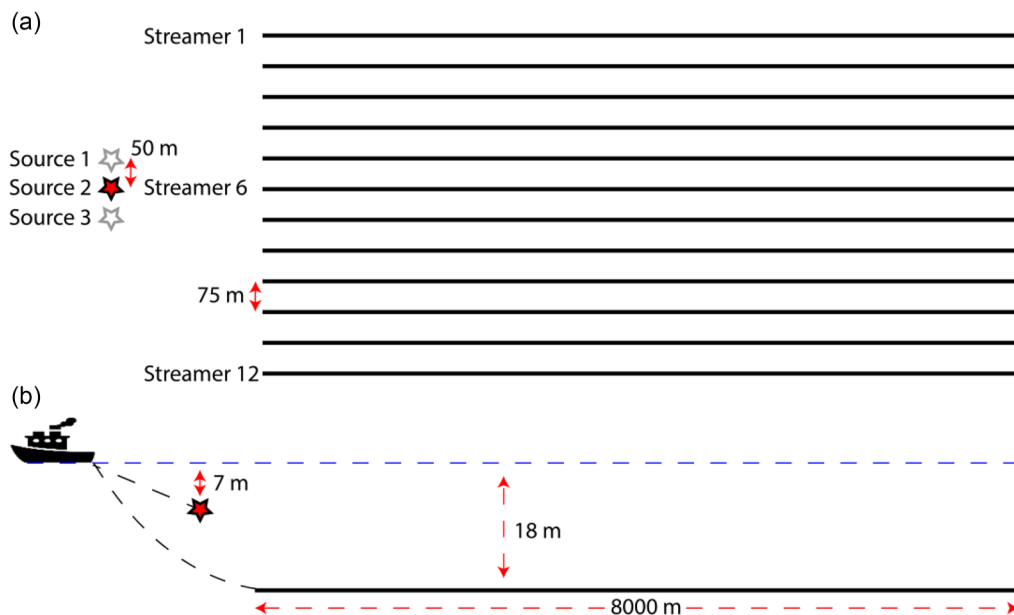


FIGURE 17 Illustration showing the acquisition geometry used by the Northern Viking Graben (NVG) survey from above the ship (a) and from the side of the ship (b).

modelled data that contains pressure data and low-cut filtered V_z data. The V_z noise amplitude is similar to what we would expect on real data. Figure 12 shows the receiver deghosting results on a common-channel gather (receiver 10) using pressure and low-cut filtered V_z data as input to the network (i.e. approach 2). Notice that we remove most of the V_z noise with a low-cut filter at 20 Hz (Figure 12a,b). The DEGDEM error is negligible, shown in the amplitude spectrum in Figure 12. The receiver deghosting results using approaches 1 and 2 are indistinguishable when displayed as common-channel gathers. Therefore, we have not shown the DEGDEM results using V_z data without a low-cut filter (i.e. approach 1) in Figure 12.

We also compare dual-component DEGDEM (using approach 1 or 2) with pressure-only DEGDEM and conventional dual-component deghosting. In the conventional approach, we combined the pressure and V_z ($P-V_z$) sum and low-frequency deghosting at 33 Hz, which was the optimal combination. Figure 13 shows the results of the amplitude spectrum of the error. Using V_z data with a low-cut filter as input to DEGDEM gives a better result than the 'raw' V_z data. Dual-component DEGDEM improves the results significantly for all frequencies compared to pressure-only DEGDEM. Conventional deghosting is generally better than dual-component DEGDEM above 50 Hz and worse below 50 Hz, but both errors are small. The V_z noise contamination is most likely the reason why DEGDEM has less error compared to $P-V_z$ sum below 50 Hz. In addition, conventional deghosting is better than pressure-only DEGDEM for all frequencies.

Figure 14S shows a shot gather with a 12.5 m receiver spacing, typical for many towed streamer datasets; the cor-

responding frequency–wavenumber (FK) spectrum is given in Figure 14c. Even though many recording systems include either analogue or digital arrays to reduce spatial aliasing, strongly dipping energy may still be aliased at higher frequencies, evident in Figure 14c above 60 Hz. This may result in some inaccuracies when applying the obliquity filter in the FK domain. In the crossline direction, the cable separation is usually between 50 and 100 m. This large spacing, along with a lack of any receiver array, makes the crossline aliasing problem even more challenging. Figure 14b shows the data from Figure 14a after decimation to a 50 m spacing. Figure 14d shows the corresponding FK domain, where we can see the energy has now become aliased at 15 Hz, a frequency four times lower than the 12.5 m spacing case. The black lines in Figure 14c indicate the Nyquist wavenumber for 50 m cable separation. For this reason, the obliquity correction is challenging in the crossline direction and will be the focus of the following synthetic example.

In our first synthetic test, we used training and prediction of data with a 12.5 m receiver spacing (Figure 15b). As an experiment, we removed every fourth receiver such that the receiver separation was 50 m (Figure 15c). We then trained a network on this sparse data and applied it to sparse prediction data. The conventional deghosting method was applied to sparse data to see how it could cope with a significant amount of energy being aliased. Due to the severe aliasing, we used a simple $P-V_z$ sum without the obliquity filter for the conventional route. Figure 16 shows the results for DEGDEM and conventional deghosting in common-channel gathers for receivers 3 and 10. Data from receiver 3 contains mostly non-aliased energy, whereas data from receiver 10 contains more

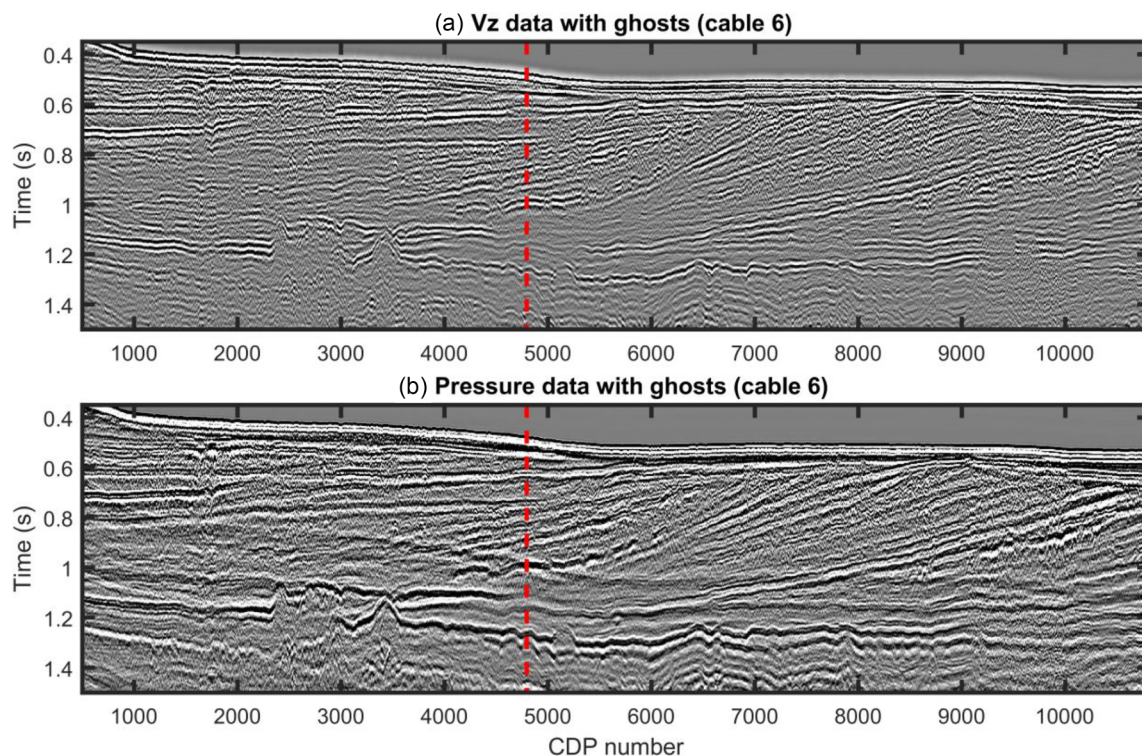


FIGURE 18 Real Northern Viking Graben (NVG) data shown as CMP stacks for the central cable. This data was used as input for deghosting using demigration-based supervised learning (DEGDDEM) and conventional deghosting: (a) V_z data with ghosts and a low-cut filter and (b) pressure data with ghosts. The area to the left of the dashed red line is used to create training and validation data.

aliased energy. The results show that conventional deghosting works well for receiver 3 but struggles for receiver 10. However, DEGDDEM performs well for both receivers 3 and 10. Figure 16 also shows the DEGDDEM results using 12.5 m data. The difference between DEGDDEM trained on 12.5 or 50 m receiver spacing data is small compared to the difference between DEGDDEM and conventional deghosting on aliased data. Current practices may include dealiasing input data through the use of interpolation, applying the obliquity correction in a sparse transform domain, or making assumptions about the kinematic behaviour of events (e.g. hyperbolic moveout). It is beyond the scope of the paper to examine these options on real data.

Real data – Northern Viking Graben dataset

The real data used in this paper is from a survey in the NVG area in the Northern North Sea off the western coast of Norway. Between 2020 and 2022, 26,000 km² of East–West data was acquired to provide a dual-azimuth survey of NVG (CGG, 2022). The survey used a multi-component streamer with a constant depth of 18 m. Figure 17 shows the survey geometry.

Before the final processing flow of this data, a fast-track processing flow was completed to produce a PSDM image. In the fast track, several processing steps were expedited to

allow a preliminary evaluation of a migrated image. Consequently, the data and processing steps (including deghosting) are not optimized as fully as in the final processing flow. The PSDM image and a smooth velocity model acquired from the fast track cover a smaller area than the whole acquisition area. We used the PSDM image to create pressure and V_z data using demigration. We trained the neural network on pressure and low-cut filtered V_z data and applied it to real pressure and low-cut filtered V_z data before deghosting (Figure 18). We used the area to the left of the dashed red line in Figure 18 to create training and validation data. Next, we tested the trained network on all the data shown in Figure 18. The neural network input window size is 3.2 s. In addition, we compared our method with conventional dual-component deghosting used in the final processing flow as quality control. We combined source and receiver deghosting for both DEGDDEM and conventional deghosting. The results are shown in different domains (common midpoint [CMP] stacks, common-channel gathers and shot gathers) for both the outer and central cables. Figures 19 and 20 show CMP stacks for the outer and central cables, respectively. Figures 21 and 22 show common-channel gathers. Shot gathers are shown in Figure 23. Finally, we show frequency panels of CMP stacks in Figure 24. Both DEGDDEM and conventional deghosting show similar results. However, some differences at the lower frequencies (0–10 Hz), around the first receiver notch (~ 42 Hz) and on the higher

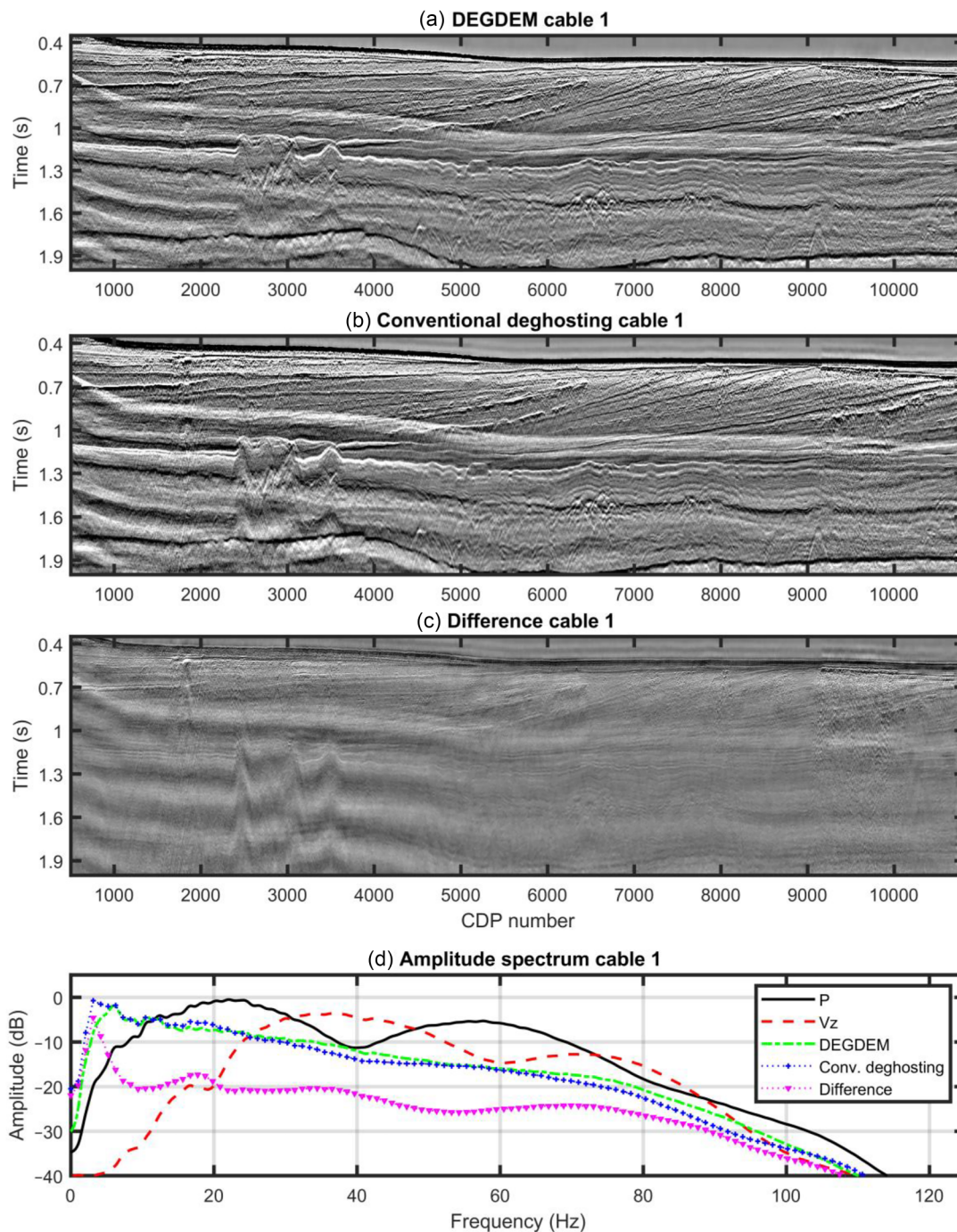


FIGURE 19 Source and receiver deghosting on the outer cable are shown as CMP stacks. (a) and (b) P - V_z deghosting using demigration-based supervised learning (DEGDEM) and conventional deghosting, respectively, with pressure and low-cut filtered V_z data as input. (c) Difference between conventional deghosting and P - V_z DEGDEM. (d) Amplitude spectrum showing pressure with ghosts, low-cut filtered V_z with ghosts, DEGDEM and conventional deghosting.

frequencies (above 70 Hz) are shown in the amplitude spectra. The frequency panels between 0 and 10 Hz (Figure 24a) also show noticeable differences which are reviewed further in the ‘Discussion’ section.

When we include the network training process, DEGDEM is approximately 2.3 times slower in time using the same resources compared to conventional deghosting when

deghosting the sail line used here. Both methods are running on 4 node GPU’s. This includes creating demigrated training data, training the neural network and deghosting the data. However, if we use the already trained network to predict the ghost-free data, it is approximately 125 times faster in time when using the same resources than conventional deghosting which makes DEGDEM suited for large data volumes. This

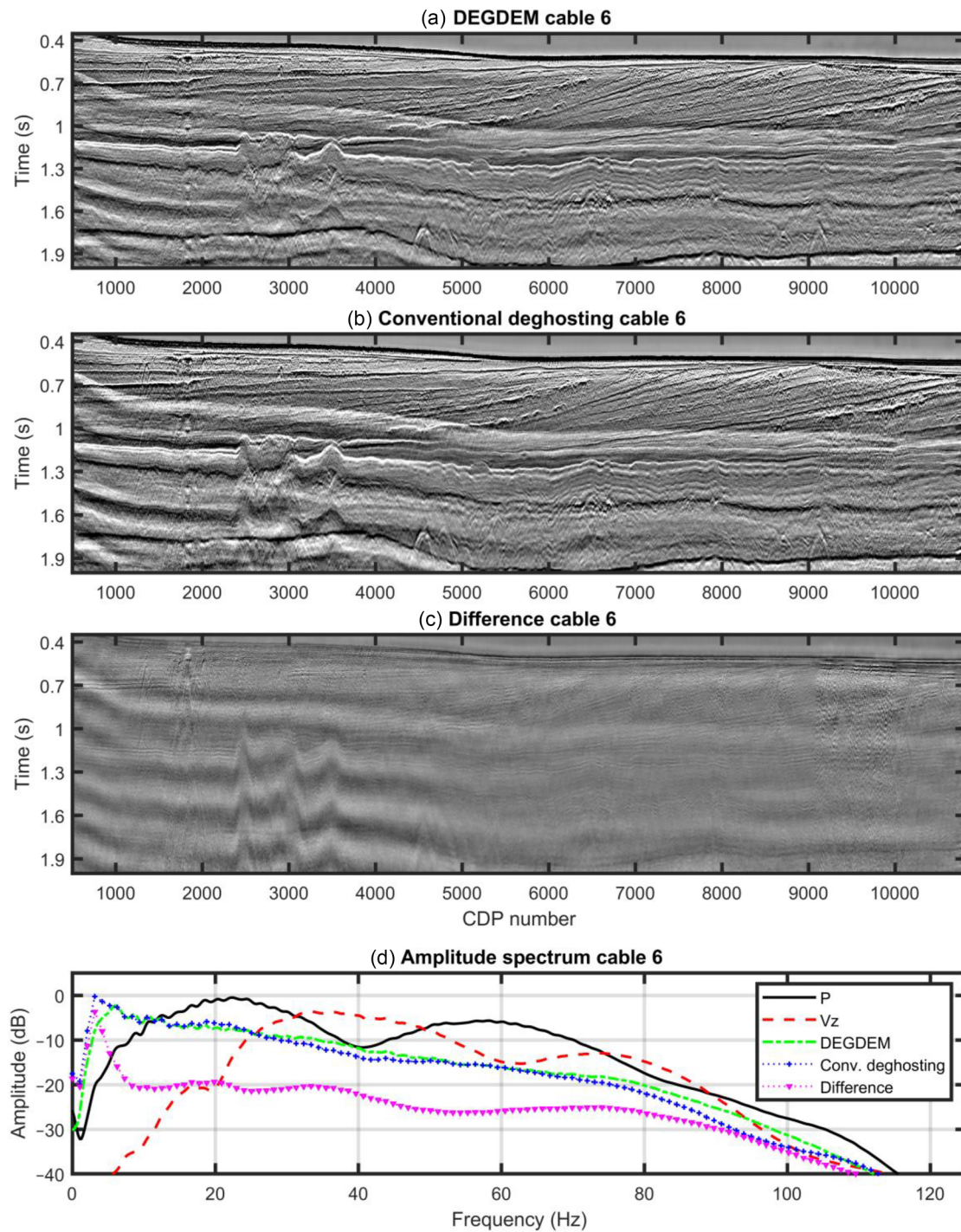


FIGURE 20 Source and receiver deghosting on the central cable are shown as CMP stacks. Descriptions of (a)–(d) are given in Figure 19.

method would also be useful when, for example, a dataset area is extended from 1 year to the next, or for new vintages of data for 4D seismic. However, if the acquisition geometry is different from the current one (i.e. different source and receiver depths), we cannot use the same trained network. In this case, we have to create demigrated training data that matches the new vintage and train a new neural network.

DISCUSSION

de Jonge, Vinje, Zhao et al. (2022) compared pressure-only deghosting using demigration-based supervised learning (DEGDEM) with conventional $\tau - p$ pressure-only deghosting by Poole (2013); both showed similar results on synthetic data. They carried out various tests on synthetic data, such as including multiples in the prediction data but not the training

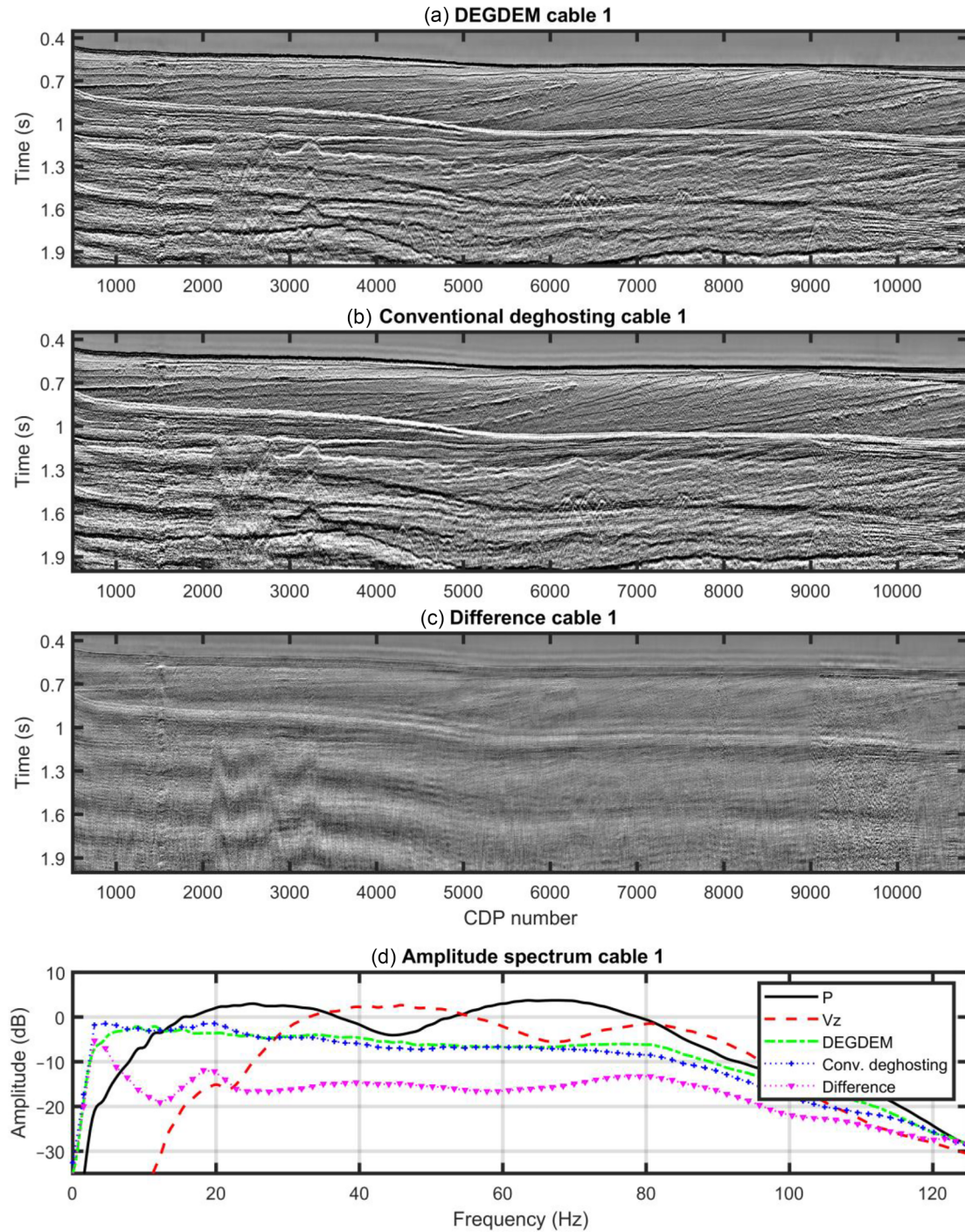


FIGURE 21 Source and receiver deghosting on the outer cable are shown as common-channel gathers (receiver 10). Descriptions of (a)–(d) are given in Figure 19.

data. In addition, they added residual ghost noise to the pre-stack depth migration (PSDM) image to simulate the case of sub-optimal initial deghosting. Even though DEGDEM was trained on data from the PSDM image with a residual ghost, it improved the initial deghosting significantly. They tested the effects of swell waves and changing the sea-surface reflection coefficient used in the training and prediction data. The swell waves and sea-surface reflection coefficient tests indicated

that DEGDEM was more robust than inversion/modelling-based conventional deghosting. The robustness of this method could be an advantage on real data where, for example, we could have swell waves or incomplete control of the receiver positions and the sea surface reflection coefficient. The results from de Jonge, Vinje, Zhao et al. (2022) should also be relevant to the dual-component DEGDEM method shown here.

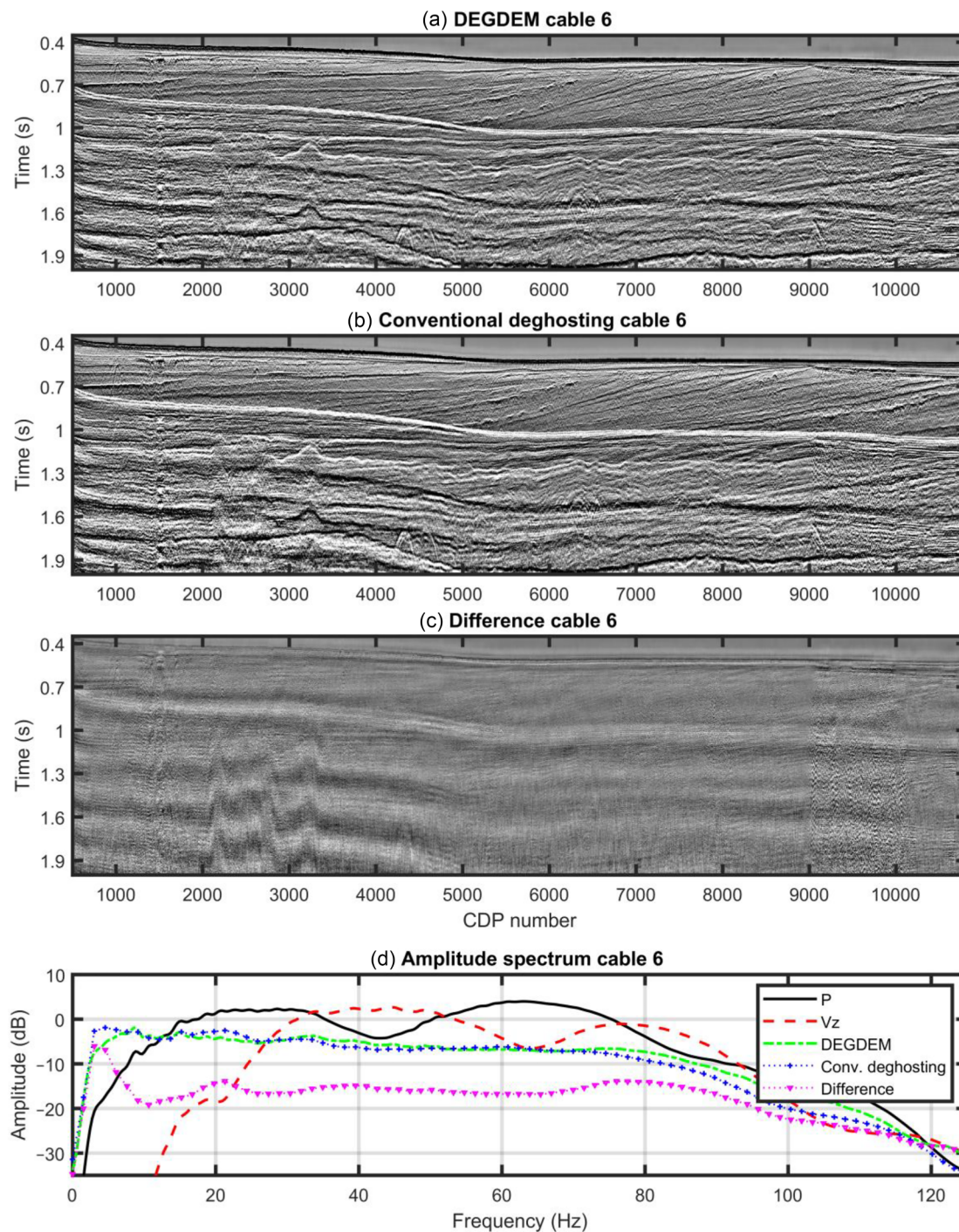


FIGURE 22 Source and receiver deghosting on the central cable are shown as common-channel gathers (receiver 10). Descriptions of (a)–(d) are given in Figure 19.

One drawback of DEGDEM is that we depend on a PSDM or a reflectivity image to create the training data. However, in a standard modern processing project, the workflow shown in Figure 2, from the ‘raw’ data to the PSDM image, is usually repeated iteratively to create ever better images. Therefore, a PSDM image should be available at an early stage in most processing projects. In addition, DEGDEM is not the

only method dependent on a PSDM image. Some demultiple methods (e.g. Brittan et al., 2011; Martin et al., 2011) or some velocity-building methods (e.g. Chang et al., 1996) also require a PSDM image. It is also possible to use a PSDM image from another acquisition. Conventional modelling or inversion-based methods often require comprehensive testing. However, they are not dependent on a PSDM image and

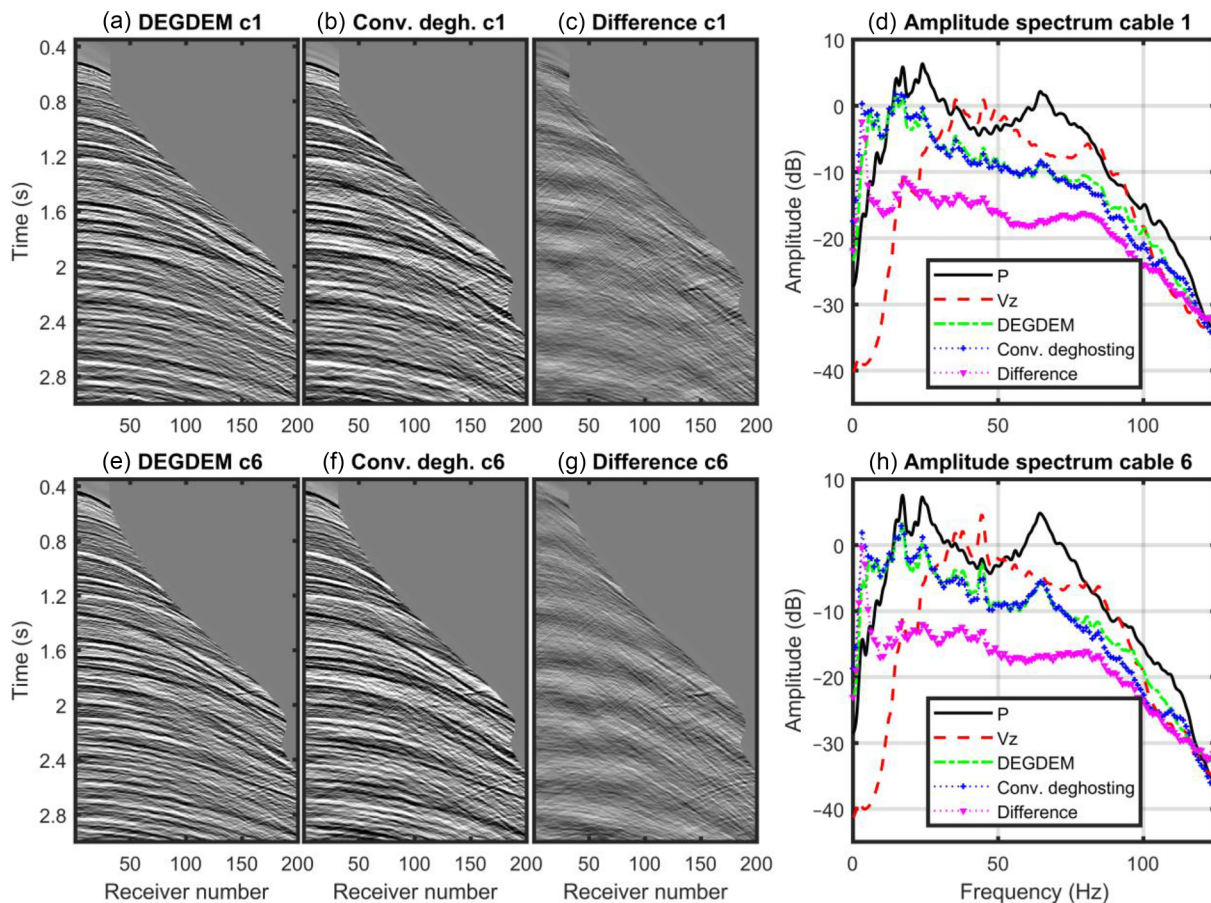


FIGURE 23 Source and receiver deghosting on the outer and central cables are shown as shot gathers. (a)–(c) P – V_z deghosting using demigration-based supervised learning (DEGDDEM), conventional deghosting and difference for the outer cable, respectively. (e)–(g) P – V_z DEGDEM, conventional deghosting and difference for the central cable, respectively. (d) and (h) Amplitude spectrums showing pressure with ghosts, low-cut filtered V_z with ghosts, DEGDEM and conventional deghosting for the outer and central cable, respectively.

are not affected by changing geology. An advantage of our method is that it is easy to use, robust, and can give good results.

Another drawback of DEGDEM is that demigration is not able to create refractions or multiples. The results from de Jonge, Vinje, Zhao et al. (2022) show that despite multiples not being in the training data, the network was able to deghost multiples. While multiples behave similarly to primary reflections, refractions behave differently, and for this reason, our neural network struggled more to deghost them compared to multiples.

Peng et al. (2021) showed that it is possible to create training data directly from data after conventional deghosting (pre-stack). One issue with this method is that the trained neural network will most likely never improve the deghosting as it will reproduce the weaknesses of conventional deghosting. The main advantage of their method is that it can reduce the computation cost by training a network on a small part of a survey and applying it to the rest of the survey or another survey. However, our method can use the same strategy dur-

ing training to reduce the cost. In addition, our method has the possibility of improving the deghosting (de Jonge, Vinje, Zhao, et al., 2022).

It is also possible to use time demigration instead of depth demigration to generate training data. Time migration is less computationally expensive than depth migration, and preliminary time-domain images are often available relatively early in the processing sequence (Iversen et al., 2012). It should also be possible to use reverse time migration (RTM) to produce a seismic image and then use demigration to create training data. It is unclear if using RTM instead of Kirchhoff migration before demigration influences the final results. However, it should be mentioned that demigration is a reverse process to Kirchhoff migration. Therefore, demigration from Kirchhoff migrated images will be more similar to the data before migration.

In the following paragraphs, we will discuss the results and conclusion on the synthetic and real-data tests.

The dual-component DEGDEM results on synthetic data are encouraging. Figure 12c,d shows that DEGDEM removed

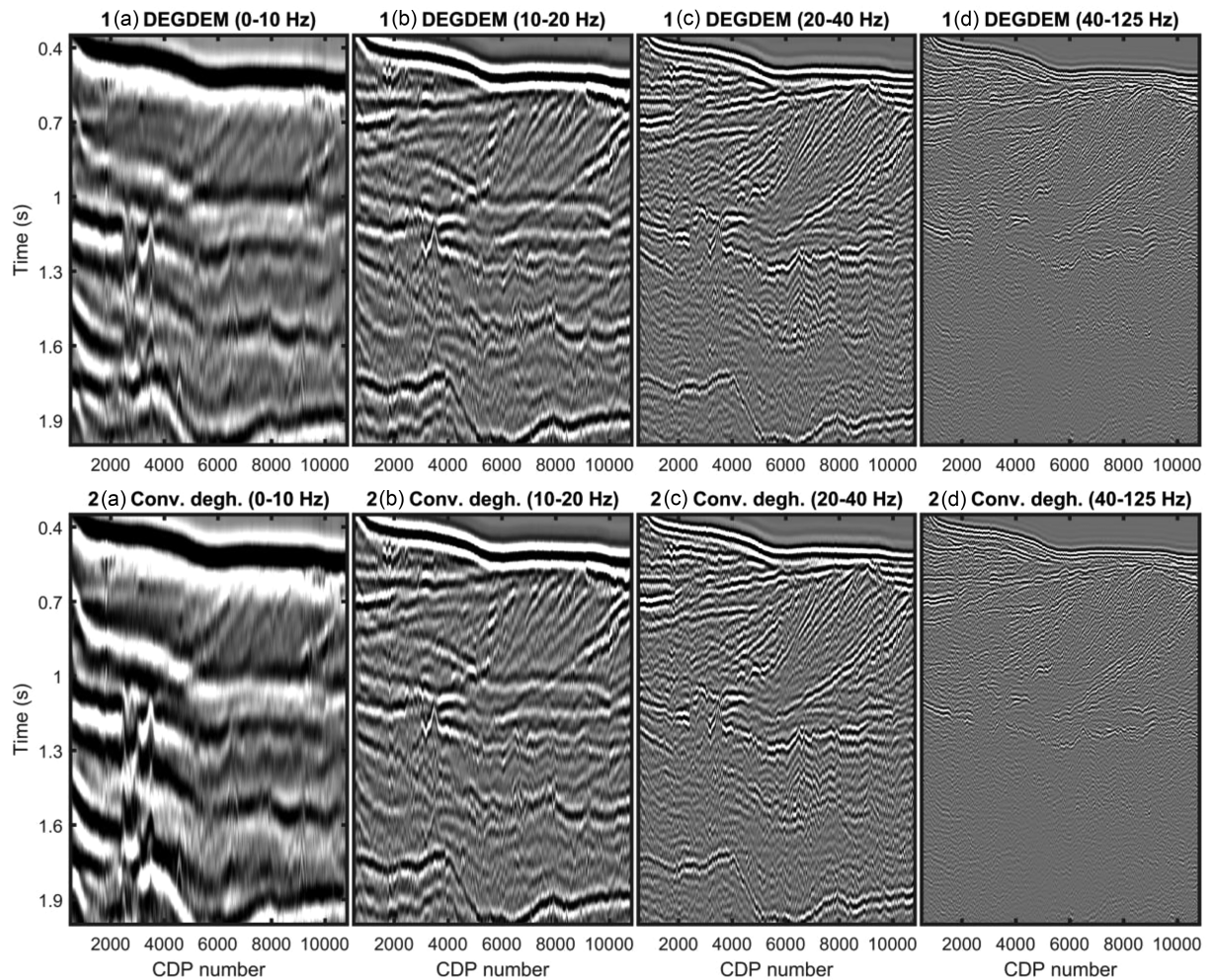


FIGURE 24 Source and receiver deghosting on the central cable are shown as CMP stacks in frequency panels. (1) P - V_z deghosting using demigration-based supervised learning (DEGDDEM) deghosting and (2) conventional deghosting. Subparts (a)–(d) are frequencies 0–10, 10–20, 20–40 and 40–125 Hz, respectively.

the ghosts effectively. In Figure 12E, we have almost identical amplitude spectra for the ground-truth and dual-component DEGDEM. In addition, Figure 13 shows that dual-component DEGDEM is better than pressure-only DEGDEM, with an improvement between ~ 5 and 15 dB for all frequencies. Conventional deghosting gave good results, especially above 50 Hz. However, below 50 Hz, dual-component DEGDEM gave a smaller error at most frequencies. Based on our results, we could use the pressure and vertical particle velocity (P - V_z) sum for the higher frequencies (e.g. above 50 Hz) and DEGDEM on the lower frequencies (e.g. below 50 Hz) for the optimal result. We speculate that it might be easier for a neural network to separate ghosts in this synthetic data compared to real data because of the ‘spiky’ nature of the synthetic data.

Figure 16 illustrates that on synthetic data, DEGDEM was not significantly affected by spatially aliased data compared to conventional deghosting. This result indicates that DEGDEM could have an advantage on sparse data where we have prob-

lems using an obliquity filter on the V_z data. DEGDEM could also have an advantage on 3D data with wide cable separation. For example, if we used 3D data as input to the network using all cables (one cable for each channel), we would have sparse data in the crossline direction. However, testing this is outside the scope of this paper. It is worth mentioning that there are some ways of dealing with aliased or sparse data. Streamers usually have anti-alias filters that prevent some aliased energy. It is also possible to interpolate traces in the inline or crossline direction (Gulunay, 2003; Wang et al., 2019) before using the P - V_z sum. An alternative way to estimate the crossline wavenumber and the obliquity factor is to use a 1D velocity model to construct hyperbolic wavefronts. The differential of these wavefronts can be used to estimate the incidence angle and, consequently, the crossline wavenumber.

Our results show that dual-component DEGDEM was able to deghost real 3D data. We compared DEGDEM with conventional dual-component deghosting to quality control our results. DEGDEM and conventional deghosting gave

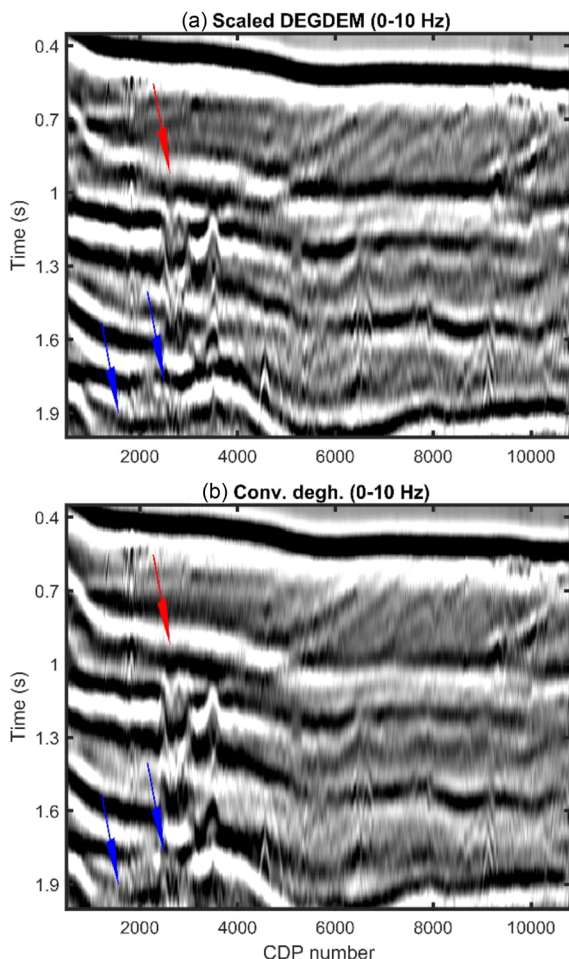


FIGURE 25 Source and receiver deghosting on the central cable are shown as CMP stacks from 0 to 10 Hz: (a) scaled P - V_z deghosting using demigration-based supervised learning (DEGD_{EM}) deghosting and (b) conventional deghosting.

similar results for both source and receiver deghosting, as shown in Figures 19–24. However, we observe differences in the lower frequencies (0–10 Hz), around the first receiver notch (~ 42 Hz) and on the higher frequencies (above 70 Hz). The biggest difference seems to be below 5 Hz. Figure 24a,d shows a difference in the amplitude of the low frequencies. At higher frequencies, the waveforms are more similar. To investigate the low frequency difference further, we scaled the amplitude of the DEGD_{EM} result slightly to match the amplitude of conventional deghosting. Figure 25 shows both DEGD_{EM} and conventional deghosting with the same amplitude on the low frequencies. We can observe that the waveforms are similar apart from three places where we observe some inconsistencies marked with arrows. The red arrows indicate places where conventional deghosting seems to have a more coherent signal, and the blue arrows indicate places where DEGD_{EM} seems to have a more coherent signal. Conventional deghosting may over-boost the amplitude on

the low-frequencies and/or DEGD_{EM} may suppress them. It is unclear why we observe a difference in amplitude on the low frequencies, and this is a subject of future research. These results give us confidence that dual-component DEGD_{EM} works on real data with good results. We used only part of the area from Figure 18 to create training data. However, the data quality inside and outside the training area is the same.

DEGD_{EM} is computationally inexpensive if a trained neural network is available. However, creating demigrated training data and training a neural network carries a significant computational cost. It is possible to train a neural network on a smaller area and apply the trained network to the rest of the survey, making this method quite efficient. DEGD_{EM} is approximately 2.3 times slower in time using the same resources than conventional deghosting when we include demigration, training and prediction. In this calculation, we include the real data results along the sail-line shown in this paper for all guns and cables. However, if the network is already trained, it is approximately 125 times faster in time than conventional deghosting. Computation time depends on the parameters used for conventional deghosting, demigration and the neural network.

CONCLUSION

In this paper, we demonstrate that dual-component deghosting using demigration-based supervised learning (DEGD_{EM}) can remove ghosts. Tests on synthetic data show that using DEGD_{EM} with dual-component data improves deghosting compared to using only pressure data. In addition, DEGD_{EM} improved deghosting below 50 Hz on synthetic data compared to conventional deghosting. We also tested our method on spatially aliased synthetic data and the results indicate that the DEGD_{EM} is less affected by aliasing compared to conventional deghosting. This is an indication that DEGD_{EM} could be beneficial on sparse data.

We used DEGD_{EM} on real data from the North Sea and compared our method with conventional deghosting. Our results show that both methods can remove the ghosts and are similar in quality for most frequencies. These results are an indication that DEGD_{EM} works well on real data. In addition, a pre-trained DEGD_{EM} will be up to a couple of orders of magnitude faster than the conventional deghosting.

ACKNOWLEDGEMENTS


The authors would like to thank CGG and the University of Bergen for providing the working environment for this research. We are grateful to CGG Earth Data for permission to use their NVG data in our study. We would also like to thank CGG internal reviewers, Kawin Nimsaila, James Cooper and Rob Schouten, for helpful comments and suggestions that improved this work. We extend our gratitude to

the Research Council of Norway, the University of Bergen and CGG for funding this work through an industrial Ph.D. grant, project no. 305450. We wish to pay tribute to former geophysics professor Norman (Norm) Bleistein who recently passed away. Norm is recognized as a key contributor to the field of seismic imaging. In this paper, his basic data mapping concept has been applied in the form of seismic demigration.

DATA AVAILABILITY STATEMENT

The real data associated with this research is confidential and cannot be released. The Marmousi model can be found online: https://wiki.seg.org/wiki/AGL_Elastic_Marmousi.

ORCID

Thomas de Jonge  <https://orcid.org/0000-0001-7834-9526>

REFERENCES

- Almuteri, K. & Sava, P. (2021) A convolutional neural network approach for ghost removal Seismic deghosting using CNNs. In: *First international meeting for applied geoscience & energy expanded abstracts*. SEG. pp. 2550–2554.
- Amundsen, L., Zhou, H., Reitan, A. & Weglein, A.B. (2013) On seismic deghosting by spatial deconvolution. *Geophysics*, 78(6), V267–V271. <https://doi.org/10.1190/GEO2013-0198.1>
- Amundsen, L., Weglein, A.B. & Reitan, A. (2013) On seismic deghosting using integral representation for the wave equation: use of Green's functions with Neumann or Dirichlet boundary conditions. *Geophysics*, 78(4), T89–T98. <https://doi.org/10.1190/GEO2012-0305.1>
- Amundsen, L. & Zhou, H. (2013) Low-frequency seismic deghosting. *Geophysics*, 78(2), WA15–WA20. <https://doi.org/10.1190/GEO2012-0276.1>
- Aytun, K. (1999) The footsteps of the receiver ghost in the f-k domain. *Geophysics*, 64(5), 1618–1626. <https://doi.org/10.1190/1.1444666>
- Bearth, R.E. & Moore, N.A. (1989) Air gun-slant cable seismic results in the Gulf of Mexico. In: *SEG technical program expanded abstracts*. SEG. pp. 649–652. <https://doi.org/10.1190/1.1889556>
- Bleistein, N. (1987) On the imaging of reflectors in the earth. *Geophysics*, 52(7), 931–942. <https://doi.org/10.1190/1.1442363>
- Bleistein, N., Stockwell, J. & Cohen, J.K. (2001) *Mathematics of multidimensional seismic imaging, migration, and inversion*. New York, NY: Springer New York. <https://doi.org/10.1007/978-1-4613-0001-4>
- Brittan, J., Martin, T., Bekara, M. & Koch, K. (2011) 3D shallow water demultiple – extending the concept. *First Break*, 29(9), 97–101. <https://doi.org/10.3997/1365-2397.29.9.53730>
- Carlson, D., Long, A., Söllner, W., Tabti, H., Tenhamn, R. & Lunde, N. (2007) Increased resolution and penetration from a towed dual-sensor streamer. *First Break*, 25(12), 71–77. <https://doi.org/10.3997/1365-2397.25.12.27722>
- CGG. (2022) *NVG 3D seismic*. CGG. Available at: <https://www.cgg.com/earth-data/multi-client-seismic/northern-viking-graben>
- Chang, H., VanDyke, J.P., Solano, M., McMechan, G.A. & Epli, D. (1996) 3-D prestack Kirchhoff depth migration: from prototype to production in a MPP environment. *Geophysics*, 63(2), 546–556. <https://doi.org/10.1190/1.1826254>
- Dumoulin, V. & Visin, F. (2018) A guide to convolution arithmetic for deep learning. ArXiv preprint. arXiv: 1603.07285v2. Available at: <https://arxiv.org/abs/1603.07285v2>
- de Jonge, T., Vinje, V., Poole, G., Song, H. & Iversen, E. (2022) De-bubbling seismic data using a generalized neural network. *Geophysics*, 87(1), V1–V14. <https://doi.org/10.1190/geo2021-0053.1>
- de Jonge, T., Vinje, V., Zhao, P., Poole, G. & Iversen, E. (2022) Source and receiver deghosting by demigration-based supervised learning. *Geophysical Prospecting*, 2022, 1–26. <https://doi.org/10.1111/1365-2478.13253>
- Greiner, T.L., Kolbjørnsen, O., Lie, J.E., Nilsen, E.H., Evensen, A.K. & Gelius, L. (2019) Cross-streamer wavefield interpolation using deep convolutional neural network. In: *SEG technical program expanded abstracts*. SEG. pp. 2207–2211. <https://doi.org/10.1190/segam2019-3214009.1>
- Gulunay, N. (2003) Seismic trace interpolation in the Fourier transform domain. *Geophysics*, 68(1), 355–369. <https://doi.org/10.1190/1.1543221>
- Hammond, J.W. (1962) Ghost elimination from reflection records. *Geophysics*, 27(1), 48–60. <https://doi.org/10.1190/1.1438977>
- Hill, D., Combee, L. & Bacon, J. (2006) Over/under acquisition and data processing: the next quantum leap in seismic technology? *First Break*, 24(6), 81–95. <https://doi.org/10.3997/1365-2397.24.1096.26991>
- Hlebnikov, V., Greiner, T.L., Vinje, V., Lie, J.E., Poole, G. (2022) De-migration-based supervised learning for interpolation and regularization of 3D offset classes. *Geophysical Prospecting*, 2022, 1–17. <https://doi.org/10.1111/1365-2478.13206>
- Iversen, E., Tygel, M., Ursin, B. & de Hoop, V. (2012) Kinematic time migration and demigration of reflections in pre-stack seismic data. *Geophysical Journal International*, 189(3), 1635–1666. <https://doi.org/10.1111/j.1365-246X.2012.05435.x>
- Jovanovich, D.B., Sumner, R.D. & Akins-Easterlin, S.L. (1983) Ghosting and marine signature deconvolution: a prerequisite for detailed seismic interpretation. *Geophysics*, 48(11), 1468–1485. <https://doi.org/10.1190/1.1441431>
- King, S. & Poole, G. (2015) Hydrophone-only receiver deghosting using a variable sea surface datum. In: *SEG technical program expanded abstracts*. SEG. pp. 4610–4614. <https://doi.org/10.1190/segam2015-5891123.1>
- Lucas, A., Iliadis, M., Molina, R. & Katsaggelos, A.K. (2018) Using deep neural networks for inverse problems in imaging: beyond analytical methods. *IEEE Signal Processing Magazine*, 35(1), 20–36. <https://doi.org/10.1109/MSP.2017.2760358>
- Martin, G.S., Wiley, R. & Marfurt, K.J. (2006) Marmousi2: an elastic upgrade for Marmousi. *Leading Edge*, 25(2), 156–166. <https://doi.org/10.1190/1.2172306>
- Martin, T., Brittan, J., Bekara, M. & Koch, K. (2011) 3D shallow water demultiple – extending the concept. In: *73rd EAGE Conference and Exhibition 2011: unconventional resources and the role of technology. Incorporating SPE EUROPEC 2011*. EAGE. pp. 2040–2045. <https://doi.org/10.3997/2214-4609.20149267>
- Masoomzadeh, H. & Woodburn, N. (2013) Broadband processing of conventional streamer data – optimized de-ghosting in the tau-P domain. In: *75th EAGE annual conference and exhibition* EAGE. pp. 1–5. <https://doi.org/10.3997/2214-4609.20130093>
- Mellier, G. & Tellier, N. (2018) *Considerations about multi-sensor solid streamer design*. Oslo, Norway: EAGE Marine Acquisition Workshop.

- Peng, C., Jin, H. & Wang, P. (2014) Noise attenuation for multi-sensor streamer data via cooperative de-noising. In: *SEG technical program expanded abstracts*. SEG. pp. 1878–1882. <https://doi.org/10.1190/segam2014-0300.1>
- Peng, H., Messud, J., Salaun, N., Hammoud, I., Jeunesse, P., Lesieur, T. et al. (2021) Proposal of the DUNet neural network architecture: deghosting example and theoretical analysis. In: *82nd EAGE annual conference & exhibition*. EAGE. pp. 1–5. <https://doi.org/10.3997/2214-4609.202112820>
- Poole, G. (2013) Pre-migration receiver de-ghosting and re-datuming for variable depth streamer data. In: *SEG technical program expanded abstracts*. SEG. pp. 4216–4220. <https://doi.org/10.1190/segam2013-0541.1>
- Poole, G. & Cooper, J. (2018) Multi-sensor receiver deghosting using data domain sparseness weights. In: *80th EAGE conference and exhibition: opportunities presented by the energy transition*. EAGE. <https://doi.org/10.3997/2214-4609.201801003>
- Rickett, J.E., van Manen, D.J., Loganathan, P., Seymour, N. (2014) Slanted-streamer data-adaptive deghosting with local plane waves. In: *76th EAGE conference & exhibition*. EAGE. pp. 1–5. <https://doi.org/10.3997/2214-4609.20141453>
- Ronneberger, O., Fischer, P. & Brox, T. (2015) U-net: convolutional networks for biomedical image segmentation. In: *International conference on medical image computing and computer-assisted intervention*. Springer. pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28
- Santos, L.T., Schleicher, J. & Tygel, M. (2000a) Modeling, migration, and demigration. *Leading Edge*, 19(7), 712–715. <https://doi.org/10.1190/1.1438696>
- Santos, L.T., Schleicher, J., Tygel, M. & Hubral, P. (2000b) Seismic modeling by demigration. *Geophysics*, 65(4), 1281–1289. <https://doi.org/10.1190/1.1444819>
- Schleicher, J., Tygel, M. & Hubral, P. (2007) *Seismic true-amplitude imaging*. Society of Exploration Geophysicists. <https://doi.org/10.1190/1.9781560801672>
- Schneider, W.A., Lerner, K.L., Burg, J.P. & Backus, M. (1964) Data-processing arrivals on technique reflection for the elimination. *Geophysics*, 29(5), 783–805. <https://doi.org/10.1190/1.1439419>
- Song, J.G., Gong, Y.L. & Li, S. (2015) High-resolution frequency-domain Radon transform and variable-depth streamer data deghosting. *Applied Geophysics*, 12(4), 564–572. <https://doi.org/10.1007/s11770-015-0525-x>
- Soubaras, R. (2010) Deghosting by joint deconvolution of a migration and a mirror migration. In: *SEG technical program expanded abstracts*. SEG. pp. 3406–3410. <https://doi.org/10.1190/1.3513556>
- Soubaras, R. & Dowle, R. (2010) Variable-depth streamer – a broadband marine solution. *First Break*, 28(12), 89–96. <https://doi.org/10.3997/1365-2397.28.12.44692>
- Sun, J., Slang, S., Elboth, T., Greiner, T.L., McDonald, S. & Gelius, L. (2019) Attenuation of marine seismic interference noise employing a customized U-Net. *Geophysical Prospecting*, 68(3), 845–871. <https://doi.org/10.1111/1365-2478.12893>
- Sun, J. & Hou, S. (2022) Improving signal fidelity for deep learning-based seismic interference noise attenuation. *Geophysical Prospecting*, 2022. <https://doi.org/10.1111/1365-2478.13268>
- Telling, R. & Grion, S. (2022) Multicomponent de-ghosting using a hybrid operator in frequency and space. In: *83rd EAGE annual conference & exhibition*. EAGE. pp. 1–5. <https://doi.org/10.3997/2214-4609.202210526>
- Tenghamn, R. & Dhelie, P.E. (2009) GeoStreamer – increasing the signal-to-noise ratio using a dual-sensor towed streamer. *First Break*, 27(10), 45–51. <https://doi.org/10.3997/1365-2397.2009017>
- Tygel, M., Schleicher, J. & Hubral, P. (1996) A unified approach to 3-D seismic reflection imaging, Part II: Theory. *Geophysics*, 61(3), 759–775. <https://doi.org/10.1190/1.1444002>
- Vrolijk, J.W. & Blacquièrre, G. (2021) Source deghosting of coarsely sampled common-receiver data using a convolutional neural network. *Geophysics*, 86(3), V185–V196. <https://doi.org/10.1190/geo2020-0186.1>
- Vrolijk, J.W. & Blacquièrre, G. (2020) Source deghosting of coarsely sampled common-receiver data using machine learning. In: *SEG technical program expanded abstracts*. SEG. pp. 3294–3298. <https://doi.org/10.1190/segam2020-3413112.1>
- Wang, B., Zhang, N., Lu, W. & Wang, J. (2019) Deep-learning-based seismic data interpolation: a preliminary result. *Geophysics*, 84(1), V11–V20. <https://doi.org/10.1190/geo2017-0495.1>
- Zhang, Z., Masoomzadeh, H. & Wang, B. (2018) Evolution of deghosting process for single-sensor streamer data from 2D to 3D. *Geophysical Prospecting*, 66(5), 975–986. <https://doi.org/10.1111/1365-2478.12614>
- Zhao, X., Lu, P., Zhang, Y., Chen, J. & Li, X. (2019) Swell-noise attenuation: a deep learning approach. *Leading Edge*, 38(12), 934–942. <https://doi.org/10.1190/tle38120934.1>

How to cite this article: de Jonge, T., Vinje, V., Poole, G., Zhao, P. & Iversen, E. (2024) Deghosting dual-component streamer data using demigration-based supervised learning. *Geophysical Prospecting*, 72, 68–91. <https://doi.org/10.1111/1365-2478.13407>