Seismic Data Interpolation Through Convolutional Autoencoder

Sara Mandelli, Federico Borra, Vincenzo Lipari, Paolo Bestagini, Augusto Sarti and Stefano Tubaro, Politecnico di Milano, Italy*

SUMMARY

A common issue of seismic data analysis consists in the lack of regular and densely sampled seismic traces. This problem is commonly tackled by rank optimization or statistical features learning algorithms, which allow interpolation and denoising of corrupted data. In this paper, we propose a completely novel approach for reconstructing missing traces of pre-stack seismic data, taking inspiration from computer vision and image processing latest developments. More specifically, we exploit a specific kind of convolutional neural networks known as convolutional autoencoder. We illustrate the advantages of using deep learning strategies with respect to state-of-the-art by comparing the achieved results over a well-known seismic dataset.

INTRODUCTION

Economic limitations, environmental constraints and elimination of badly acquired traces cause irregular spatial sampling in almost all seismic acquisitions. However, many seismic data processing pipelines (e.g., for denoising, multiple removal, imaging, etc.) need data to be regularly and densely sampled. Therefore, a large number of interpolation algorithms to densely reconstruct coarse data have been proposed so far, based on different strategies.

Model-based methods are related to continuation operators, which implement an implicit migration-demigration pair (Stolt, 2002; Fomel, 2003). However, in case of complex structural burden, the performances of these algorithms are strongly affected. Other approaches are based on prediction filters (Spitz, 1991; Abma and Claerbout, 1995), which assume seismic data to be a (local) superposition of plane waves. Fourier reconstruction methods are another way for handling the interpolation problem (Duijndam et al., 1999): Fourier coefficients are estimated from input data, which can be recovered on any desired grid. Nevertheless, these last two methods are effective only for regularly sampled data, which is a heavy limitation. More recently, compressive sensing has been exploited for the interpolation problem. For instance, (Xu et al., 2005) assumes data sparsity in the frequency-wavenumber domain, and (Herrmann and Hennenfent, 2008; Gan et al., 2015; Wang et al., 2014) exploit sparsity in curvelet, seislet and dreamlet transforms' domain. Similarly, (Özbek et al., 2009; Nguyen et al., 2011; Fioretti et al., 2015) exploits Matching Pursuit algorithm to enforce data sparsity, while recent strategies propose to recast interpolation as a low rank matrix/tensor completion problem (Oropeza and Sacchi, 2011; Yang et al., 2013; Kumar et al., 2014; Adamo et al., 2015).

However, in the latest years, innovative strategies for data interpolation based on deep learning have been proposed in manifold image processing tasks. Indeed, solutions based on convolutional neural networks are ever more competing with traditional methodologies, often exceeding state-of-the-art results.

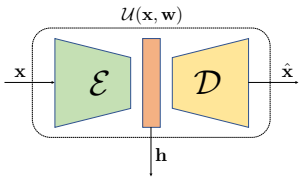

Figure 1: Scheme of an autoencoder architecture.

In this paper, we aim at interpolating irregular pre-stack seismic data, investigating them in the shot-gather domain. Inspired by the important contributions achieved in several image processing problems, we propose to exploit a convolutional autoencoder as a novel and strongly competitive strategy for reconstruction of missing traces in pre-stack seismic images. Results show promising performances of our method compared to recent state-of-the-art techniques.

BACKGROUND ON AUTOENCODERS

In this section we quickly introduce the idea behind an autoencoder that is useful to understand the rest of the paper. For a thorough autoencoder review, please refer to (Goodfellow et al., 2016).

An autoencoder is a specific kind of neural network whose architecture can be logically split in two separate components. To be more precise, let us refer to Fig. 1 for analyzing the autoencoder structure: (i) the encoder, represented by the operator $\mathcal{E}$, maps the input $\mathbf{x}$ into the hidden representation $\mathbf{h} = \mathcal{E}(\mathbf{x})$; (ii) the decoder, represented by the operator $\mathcal{D}$, transforms the hidden representation into an estimate of the input $\hat{\mathbf{x}} = \mathcal{D}(\mathbf{h})$.

In this paper, we refer to a specific family of neural network architectures known as *U-nets*, which can be easily adapted as autoencoders. In particular, the name *U-net* is due to the shape they are graphically represented with. In Fig. 2, it is possible to observe the typical U-shape, as well as two different paths: (i) the left path (contracting path), which can be interpreted as the encoder; (ii) the right path (expansive path), which can be interpreted as the decoder.

As in standard autoencoder architectures, both the encoder and decoder operators are composed by series of linear filtering operations (i.e., convolutions), optionally followed by non linear functions (e.g., sigmoid, hyperbolic tangent, etc.). However, in a *U-net* architecture, the representations of the input obtained at different levels of the encoder are directly concatenated to the corresponding decoder levels. For the sake of brevity, we refer the interested readers to (Ronneberger et al., 2015) for a

10.1190/segam2018-2995428.1

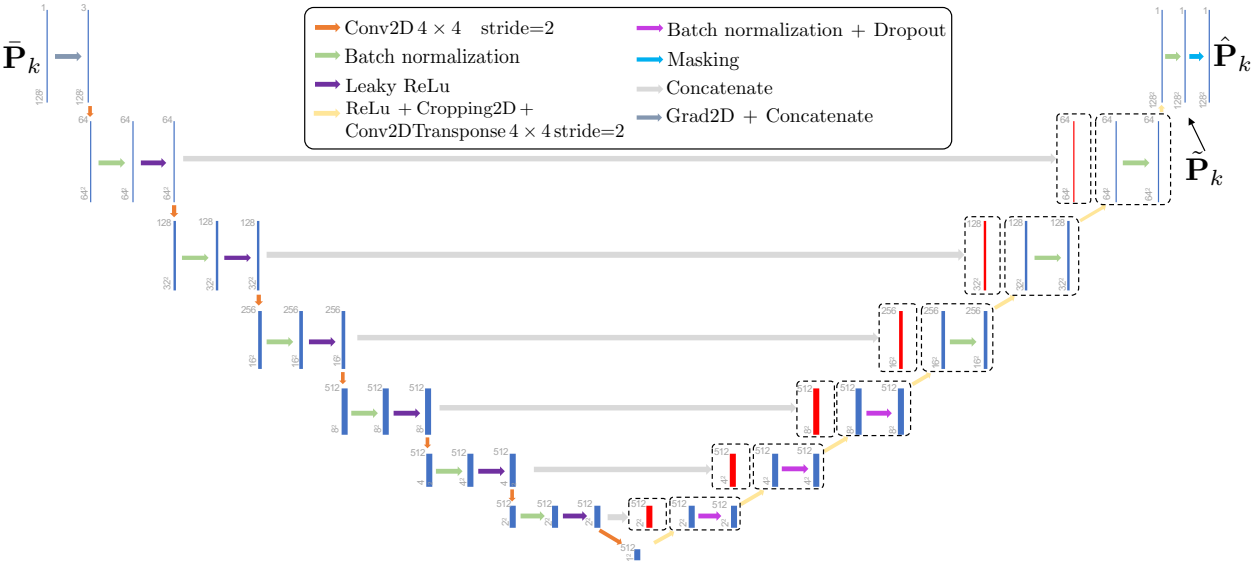

Figure 2: Architecture of the used U-Net.

detailed explanation of these architectures.

By using this kind of architectures, it is possible to estimate an almost-invertible dimensionality reduction function $\mathcal{E}$ directly from a representative set of training data (i.e., observations of $\mathbf{x}$ in Fig. 1). A typical approach consists in a priori defining a parametric network model $\mathcal{U}(\mathbf{x},\mathbf{w})$, and consequently estimating the network weights $\mathbf{w}$ through the minimization of a distance metric defined between the network input $\mathbf{x}$ and its output $\hat{\mathbf{x}}$. The distance is usually referred as loss function, and its minimization is carried out using iterative techniques (e.g., gradient descent methods, etc.).

PROPOSED SOLUTION

The rationale behind the proposed solution is that autoencoders can be powerful instruments to tackle inpainting problems (Xie et al., 2012; Pathak et al., 2016). Indeed, the encoder part of an autoencoder results in a compact representation of the input that can be built also from images containing missing data. Therefore, if the compact representation is correctly built, the result of the decoder is a "dense" image without missing traces. It is therefore possible to train an autoencoder to learn a hidden representation suitable for a compact description of input pre-stack seismic images. Once trained, the autoencoder will be able to fill the holes in corrupted pre-stack seismic image by going through encoding and decoding stages. Specifically, we train the autoencoder in order to transform images containing missing traces (from now on denoted as $\bar{\mathbf{I}}$) into regularly sampled seismic images (depicted as $\mathbf{I}$). In the following, we report all details about the proposed system.

System Training: At training time, we consider to have access to regularly sampled pre-stack seismic images $\mathbf{I}$ and their corrupted versions $\bar{\mathbf{I}}$. This image dataset is split into two disjoint sets, namely training $\mathcal{D}_T$ and validation $\mathcal{D}_V$. In order to focus on local portions of the image and to ensure a sufficiently large amount of data under analysis, we propose to work in a patch-wise fashion. We divide each image into K patches of size 128×128, with patch-stride of 32 pixels. More specifically, patches $\mathbf{P}_k$ and $\bar{\mathbf{P}}_k$, $k \in [1,K]$, contain information regarding images $\mathbf{I}$ and $\bar{\mathbf{I}}$, respectively. Each patch is normalized by removing the mean and scaling it to unit variance before being fed to the network.

Let us call $\mathcal{U}(\cdot,\mathbf{w})$ the network model that represents the input-output relation of our autoencoder, where $\mathbf{w}$ represents network's parameters. Following the same rational behind (**?**), our architecture is composed by the blocks shown in Fig. 2:

1. An input layer that computes the gradients of the input patch $\bar{\mathbf{P}}_k$ in both directions and concatenates them with the original patch. This is done to ensure that the network learns information related to local image gradients, which are greatly informative for the considered problem.
2. Six stages where a 2D convolution with filter size 4×4 and stride 2, followed by batch normalization and a Leaky ReLu, are performed. These stages lead to the hidden representation (i.e., the result of the encoding part). It is worth noting that the number of filters increases from 64 to 512 as we go deep in the network.
3. Six stages where a ReLu, a Cropping, a 2D convolution with filter size 4×4 and stride 2 followed by batch normalization and dropout are performed. In each stage we concatenate the result of the corresponding encoding stage. The number of filers is gradually diminished as we go up in the left path of the network (i.e., decoding path). The output of this stages is a patch $\tilde{\mathbf{P}}_k$, of the same size of the input patch.

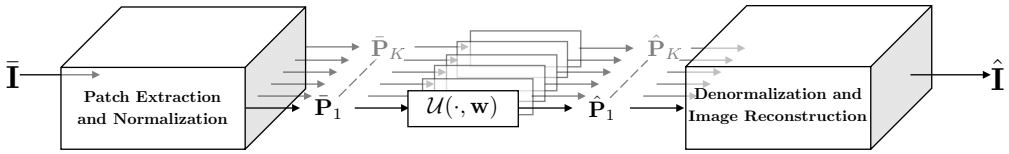

Figure 3: Proposed pipeline for reconstruction of the seismic image $\bar{\mathbf{I}}$ with missing data traces.

4. A masking stage that fills corrupted traces of $\bar{\mathbf{P}}_k$ from the reconstructed patch $\tilde{\mathbf{P}}_k$ is used at the end of the network. More precisely, the output $\hat{\mathbf{P}}_k$ of the architecture is computed as

$$\hat{\mathbf{P}}_k = \bar{\mathbf{P}}_k + \tilde{\mathbf{P}}_k \otimes \mathbf{M}, \tag{1}$$

where $\otimes$ is the Hadamard product and $\mathbf{M}$ is a binary mask defined as

$$[\mathbf{M}]_{i,j} = \begin{cases} 1 & \text{if } \left[\bar{\mathbf{P}}_k\right]_{i,j} \text{is a missing sample} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The overall architecture is characterized by almost 42 million parameters.

In the training phase, information about the original image is exploited for estimating the autoencoder parameters $\mathbf{w}$. In particular, model weights are estimated by minimizing the squared error between $\mathbf{P}_k$ and $\hat{\mathbf{P}}_k$, over all patches in the training set, i.e.,

$$\mathbf{w} = \arg\min_{\mathbf{w}} \sum_{\mathbf{P}_k \in \mathcal{D}_T} \left\|\mathbf{P}_k - \hat{\mathbf{P}}_k\right\|_{\mathrm{F}}^2, \tag{3}$$

where, with a slight abuse of notation, $\mathbf{P}_k \in \mathcal{D}_T$ denotes patches extracted from training images and $\|\cdot\|_{\mathrm{F}}$ represents the Frobenius norm. As in standard neural network training, we follow an iterative procedure to minimize (3), stopping at the iteration where the squared error over the patches $\mathbf{P}_k \in \mathcal{D}_V$ is minimum. Specifically, we used Adam optimization algorithm (Kingma and Ba, 2014) on batches of 128 patches. The learning rate and the patience are initialized at 0.01 and 10, respectively. The former is decimated while the latter is halved in presence of plateau of the cost function. We verified that the smallest loss on validation patches is achieved within the first 20 training epochs.

System Deployment: When a new image $\bar{\mathbf{I}}$ belonging to a new evaluation set $\mathcal{D}_E$ of corrupted images is under analysis, its inpainted version is estimated following the scheme depicted in Fig. 3. A set of K patches is obtained from image $\bar{\mathbf{I}}$ as described above. Each patch $\bar{\mathbf{P}}_k$ is processed by the architecture in order to estimate the reconstructed patch $\hat{\mathbf{P}}_k$. Finally, in order to reconstruct the image $\hat{\mathbf{I}}$, all the estimated patches $\hat{\mathbf{P}}_k$ are re-assembled together by sample-wise averaging the overlapping portions.

RESULTS

In this section we report the results achieved by the conducted experimental campaign. First, we present the datasets exploited for evaluating results, then we compare the performances of the proposed method with the solution given by a rank-reduction method for seismic data reconstruction (Oropeza and Sacchi, 2011).

Datasets: We exploited the well known Mobil Avo Viking Graben Line 12 dataset (Keys and Foster, 1998) for generating acquisitions with missing traces. This dataset, from now on $\mathcal{V}$ for brevity, consists of $N = 1001$ gray-scale seismic images. Specifically, each image $\mathbf{I} \in \mathbb{R}^{1024\times128}$: rows represent time domain, sampled every 4ms; columns are in spatial domain with 25m of sampling. In order to simulate the lack of seismic acquisitions, we delete a percentage H of the available data traces. To be precise, for each acquired image $\mathbf{I}$, we randomly delete the $H\%$ of its traces, $H \in \{10, 30, 50\}$, obtaining a holed image $\bar{\mathbf{I}}$. We generate 3 different datasets $\mathcal{V}_H$: $\mathcal{V}_{10}$, $\mathcal{V}_{30}$, $\mathcal{V}_{50}$, each one including N seismic images with increasing amount of missing traces. For the sake of clarity, Fig. 4 reports an example of original image $\mathbf{I}$ and related $\bar{\mathbf{I}}$ according to different percentages H.

Autoencoder set-up: In order to properly evaluate machine learning based algorithms, we split each dataset $\mathcal{V}_H$ into training, validation and evaluation, using 57% of images for training set $\mathcal{D}_T$, 18% for validation set $\mathcal{D}_V$ and the remaining for evaluation set $\mathcal{D}_E$. Since we are working in patch-wise fashion, the process involves more than 16000 training patches, more than 5400 validation patches, and more than 7200 testing patches for each dataset $\mathcal{V}_H$.

Results: We evaluate the performances of our method in reconstructing the entire seismic images belonging to the evaluation set $\mathcal{D}_E$, choosing Signal to Noise Ratio (SNR) as accuracy metrics. Precisely, SNR is defined as

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma^2(\mathbf{I})}{\sigma^2(\mathbf{I} - \hat{\mathbf{I}})} \quad \mathbf{I} \in \mathcal{D}_E, \tag{4}$$

where σ^2 is the image variance. Fig. 5 shows results obtained for the image previously reported in Fig. 4. As a matter of fact, the trained autoencoder is able to almost perfectly reconstruct even the most corrupted images, minimally introducing any spatial artifact.

As state-of-the-art comparison, we tested a rank-reduction method for seismic data reconstruction. In particular, we tested an algorithm known as Multichannel Singular Spectrum Analysis (MSSA) (Oropeza and Sacchi, 2011). This method is based on a rank-reduction algorithm exploiting Truncated Singular

10.1190/segam2018-2995428.1

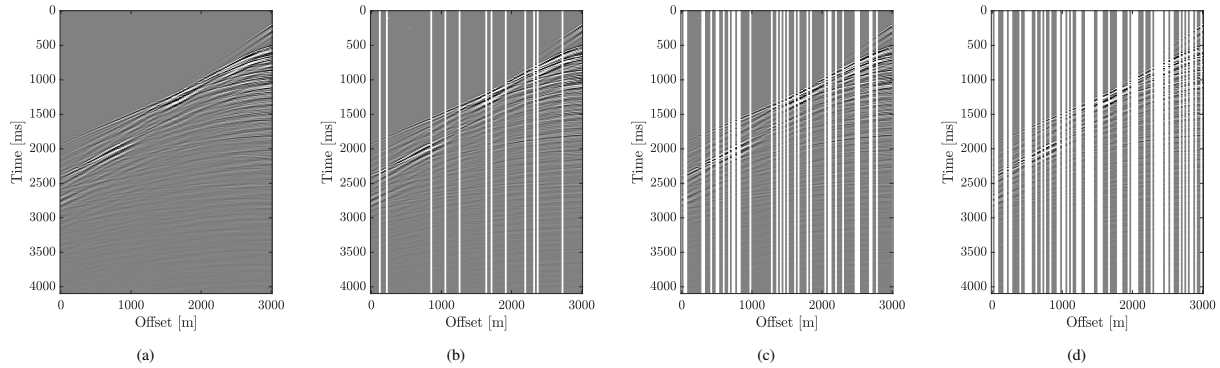

Figure 4: (a) Original dense image $\mathbf{I}$. (b) Image $\bar{\mathbf{I}} \in \mathcal{V}_{10}$, with 10% of missing traces. (c) Image $\bar{\mathbf{I}} \in \mathcal{V}_{30}$, with 30% of missing traces. (d) Image $\bar{\mathbf{I}} \in \mathcal{V}_{50}$, with 50% of missing traces.

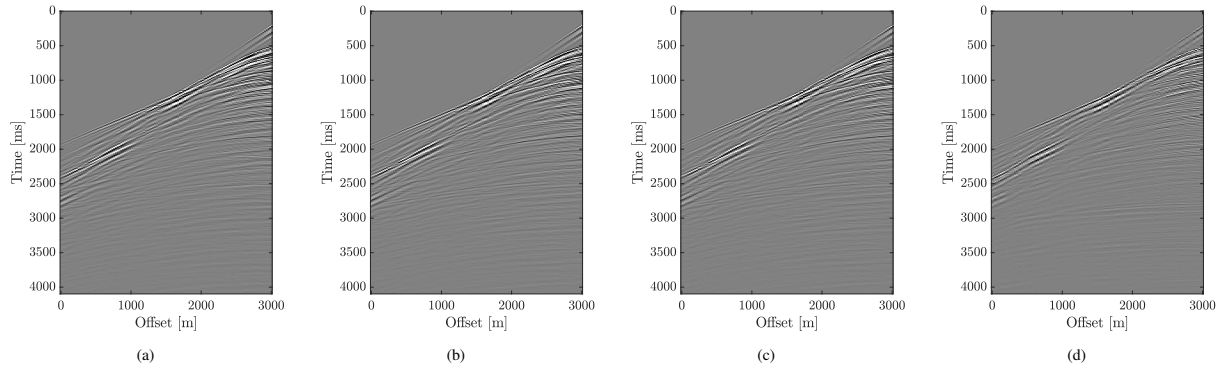

Figure 5: (a) Original dense image $\mathbf{I}$. (b) Image $\hat{\mathbf{I}}$, starting from 10% of missing traces, SNR $= 27$ dB. (c) Image $\hat{\mathbf{I}}$, starting from 30% of missing traces, SNR $= 18.2$ dB. (d) Image $\hat{\mathbf{I}}$, starting from 50% of missing traces, SNR $= 12.2$ dB.

Value Decomposition for data reconstruction. We adopted the MSSA open-source Matlab implementation provided by (Chen et al., 2016).

Table.1 depicts the accuracies obtained over the 3 datasets $\mathcal{V}_{10}$, $\mathcal{V}_{30}$, $\mathcal{V}_{50}$, averaging the results for the images in $\mathcal{D}_E$, comparing the proposed method (*U-net*) and MSSA. It is noticeable that our method improves over the MSSA solution. This is mainly due to the great capability of the autoencoder in learning features of the seismic images in the training phase. Moreover, the proposed method has a further advantage, which is the low computational effort in reconstructing a generic image in evaluation phase. As a matter of fact, if a risible amount of time is needed for training the network model parameters, the evaluation phase is very efficient: barely a second is necessary for estimating each inpainted image $\hat{\mathbf{I}}$.

SNR [dB]	$\mathcal{V}_{10}$	$\mathcal{V}_{30}$	$\mathcal{V}_{50}$
U-net	22.5	15.1	10.2
MSSA	15.6	8.9	5.1

Table 1: Average SNR for *U-net* and MSSA.

CONCLUSIONS

In this paper, we proposed a method for reconstruction of corrupted seismic data, focusing on interpolation of missing prestack data traces in the shot-gather domain. In particular, our approach follows a completely innovative point of view with respect to common techniques exploited in the seismic processing pipeline. We introduced convolutional neural network architectures for interpolating missing data traces. The proposed method proved to be simple and computationally efficient. Moreover, we compared our results with state-of-the-art solutions, showing significant performances. In light of the promising results achieved with convolutional neural networks, we believe these tools can pave the way towards even more efficient and accurate solutions. Therefore, future work will be devoted to investigations of deep learning strategies for better reconstruction of corrupted data in presence of strong noise.

ACKNOWLEDGMENTS

The authors would like to thank Nicola Bienati for insightful discussions on the topic.

10.1190/segam2018-2995428.1

REFERENCES

Abma, R., and J. Claerbout, 1995, Lateral prediction for noise attenuation by t-x and f-x techniques: Geophysics, **60**, 1887–1896, https://doi.org/10.1190/1.1443920.

Adamo, A., P. Mazzucchelli, and N. Bienati, 2015, Irregular interpolation of seismic data through low-rank tensor approximation: 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 4292–4295, https://doi.org/10.1109/IGARSS.2015.7326775.

Chen, Y., W. Huang, D. Zhang, and W. Chen, 2016, An open-source Matlab code package for improved rank-reduction 3D seismic data denoising and reconstruction: Computers & Geosciences, **95**, 59–66, https://doi.org/10.1016/j.cageo.2016.06.017.

Duijndam, A., M. Schonewille, and C. Hindriks, 1999, Reconstruction of band-limited signals, irregularly sampled along one spatial direction: Geophysics, **64**, 524–538, https://doi.org/10.1190/1.1444559.

Fioretti, L., P. Mazzucchelli, and N. Bienati, 2015, Comparison of pursuit algorithms for seismic data interpolation imposing sparseness: 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 3095–3098, https://doi.org/10.1109/IGARSS.2015.7326471.

Fomel, S., 2003, Seismic reflection data interpolation with differential offset and shot continuation: Geophysics, **68**, 733–744, https://doi.org/10.1190/1.1567243.

Gan, S., S. Wang, Y. Chen, Y. Zhang, and Z. Jin, 2015, Dealiased seismic data interpolation using seislet transform with low-frequency constraint: IEEE Geoscience and Remote Sensing Letters, **12**, 2150–2154, https://doi.org/10.1109/LGRS.2015.2453119.

Goodfellow, I., Y. Bengio, and A. Courville, 2016, Deep learning: MIT Press **1**.

Herrmann, F. J., and G. Hennenfent, 2008, Non-parametric seismic data recovery with curvelet frames: Geophysical Journal International, **173**, 233–248, https://doi.org/10.1111/j.1365-246X.2007.03698.x.

Keys, R. G., and D. J. Foster, 1998, A data set for evaluating and comparing seismic inversion methods, in R. G. Keys, and D. J. Foster, eds., Comparison of seismic inversion methods on a single real data set: SEG, 1–12.

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

Kumar, R., A. Y. Aravkin, E. Esser, H. Mansour, and F. J. Herrmann, 2014, SVD-free low-rank matrix factorization—Wavefield reconstruction via jittered subsampling and reciprocity: 76th Annual International Conference and Exhibition, EAGE, Extended Abstracts, https://doi.org/10.3997/2214-4609.20141394.

Nguyen, T., and R. Winnett, 2011, Seismic interpolation by optimally matched Fourier components: 81st Annual International Meeting, SEG, Expanded Abstracts, 3085–3089, https://doi.org/10.1190/1.3627836.

Oropeza, V., and M. Sacchi, 2011, Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis: Geophysics, **76**, no. 3, V25–V32, https://doi.org/10.1190/1.3552706.

øzbek, A., A. K. øzdemir, and M. Vassallo, 2009, Interpolation by matching pursuit: 79th Annual International Meeting, SEG, Expanded Abstracts, 3254–3258, https://doi.org/10.1190/1.3255534.

Pathak, D., P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, 2016, Context encoders: Feature learning by inpainting: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2536–2544, https://doi.org/10.1109/CVPR.2016.278.

Ronneberger, O., P. Fischer, and T. Brox, 2015, U-net: Convolutional networks for biomedical image segmentation: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 234–241.

Spitz, S., 1991, Seismic trace interpolation in the f-x domain: Geophysics, **56**, 785–794, https://doi.org/10.1190/1.1443096.

Stolt, R. H., 2002, Seismic data mapping and reconstruction: Geophysics, **67**, 890–908, https://doi.org/10.1190/1.1484532.

Wang, B., R.-S. Wu, Y. Geng, and X. Chen, 2014, Dreamlet-based interpolation using POCS method: Journal of Applied Geophysics, **109**, 256–265, https://doi.org/10.1016/j.jappgeo.2014.08.008.

Xie, J., L. Xu, and E. Chen, 2012, Image denoising and inpainting with deep neural networks: Proceedings of the 25th International Conference on Neural Information Processing Systems — Volume 1, Curran Associates Inc., 341–349.

Xu, S., Y. Zhang, D. Pham, and G. Lambaré, 2005, Antileakage Fourier transform for seismic data regularization: Geophysics, **70**, no. 4, V87–V95, https://doi.org/10.1190/1.1993713.

Yang, Y., J. Ma, and S. Osher, 2013, Seismic data reconstruction via matrix completion: Inverse Problems and Imaging, **7**, 1379–1392, https://doi.org/10.3934/ipi.