Deep learning applied to seismic attribute computation

Donald P. Griffith[1], S. Ahmad Zamanian[1], Jeremy Vila[1], Antoine Vial-Aussavy[1], John Solum[1], R. David Potter[1], and Francesco Menapace[1]

Abstract

We have trained deep convolutional neural networks (DCNs) to accelerate the computation of seismic attributes by an order of magnitude. These results are enabled by overcoming the prohibitive memory requirements typical of 3D DCNs for segmentation and regression by implementing a novel, memory-efficient 3D-to-2D convolutional architecture and by including tens of thousands of synthetically generated labeled examples to enhance DCN training. Including diverse synthetic labeled seismic in training helps the network generalize enabling it to accurately predict seismic attribute values on field-acquired seismic surveys. Once trained, our DCN tool generates attributes with no input parameters and no additional user guidance. The DCN attribute computations are virtually indistinguishable from conventionally computed attributes while computing up to 100 times faster.

Introduction

The quantity of geoscience data available for analysis is growing at an exponential rate, and interpreters find it increasingly difficult to incorporate all available information in their analyses of the subsurface. Deep convolutional neural networks (DCNs) offer the possibility of obtaining consistent, exhaustive, and accurate interpretations in a fraction of the time it takes to perform manual interpretations with commonly used interactive seismic interpretation software. Having DCNs perform aspects of the interpretation thereby extends the interpreter's capability to understand the subsurface while spending less time at the workstation performing repetitive tasks and more time devoted to high-level strategy and risk assessment.

DCNs have only been used by the scientific community for a short time. Their powerful ability to solve vision tasks became widely understood starting in 2012 when a single entrant used a DCN (Krizhevsky et al., 2012) in the ImageNet large scale visual recognition challenge to significantly reduce object classification error levels compared with previous challenge winners. By 2015, all contest participants were using DCNs and classification performance for the winning entrant was better than human performance (He et al., 2015; Russakovsky et al., 2015).

The medical industry has been quick to adopt DCNs to classify and segment medical images for improving abnormality detection and patient care. Applications of deep learning (DL) in the medical field include retinal disease screening and diagnosis (Hajabdollahi et al., 2018), real-time liver CT scan analysis to assist radiologists (Rafiei et al., 2018), cardiac segmentation (Poudel et al., 2016), and multiple approaches to brain-lesion segmentation (De Brebisson and Montana, 2015; Casamitjana et al., 2017; Erden et al., 2017; Kamnitsas et al., 2017; Mernagh and Pendse, 2017; Salehi et al., 2017). One of the most influential DCN architectures, named U-Net, was initially developed in two dimensions for medical applications (Ronneberger et al., 2015). The method was then extended to three dimensions to demonstrate semantic segmentation tasks (Cicek et al., 2016). The use of U-Net and 3D architectures is also applicable for seismic.

One of the first examples of DL for geoscience subsurface problems demonstrated how to locate faults in prestack synthetically generated seismic data (Araya-Polo et al., 2017). DCNs were also used with poststack synthetic seismic to predict faults (Huang et al., 2017). Although applied in different domains, both approaches used 2D slices of fully labeled synthetic data to train their DCN models. In other approaches, many investigators have reported using manual interpretation on a small portion of 2D cross sections subselected from poststack 3D seismic surveys to train a DCN, then inferring with the trained model on the remainder of the survey to assist with interpretation. This was demonstrated for interpreting the location of salt diapirs (Gramstad and Nickel, 2018; Shi et al., 2018; Waldeland et al., 2018), faults (Guo et al., 2018; Maniar et al., 2018; Zhao

[1]Shell International E&P Inc., Houston, Texas, USA. E-mail: donald.griffith@shell.com; sam.zamanian@shell.com; jeremy.vila@shell.com; a.vial-aussavy@shell.com; j.solum@shell.com; d.potter@shell.com; francesco.menapace@shell.com.

Manuscript received by the Editor 30 November 2018; revised manuscript received 3 March 2019; published ahead of production 09 April 2019; published online 12 June 2019. This paper appears in *Interpretation*, Vol. 7, No. 3 (August 2019); p. SE141–SE150, 9 FIGS.

http://dx.doi.org/10.1190/INT-2018-0227.1.

and Mukhopadhyay, 2018), and channels (Pham et al., 2018). Another fault interpretation study trained a 3D DCN entirely on synthetic seismic and then ran predictions on field-recorded poststack seismic (Wu et al., 2018).

This work extends several aspects of previously published approaches. An efficient 3D fully convolutional DCN architecture is introduced that performs voxel-level regression on entire cross sections using large receptive fields, so that geologic context is fully incorporated into predictions. Synthetic poststack 3D seismic is used for training the DCNs. The synthetic seismic realizations provide highly diverse, sufficiently complex simulated depositional and structural environments that also include simulated seismic acquisition and processing artifacts. Finally, DCNs are applied to regression tasks in which the goal is to accurately replicate computationally intense seismic attribute computation workflows with dramatically reduced compute times.

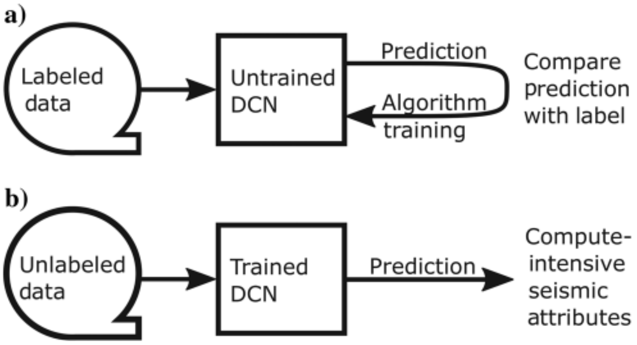

Figure 1. The DL schematic. (a) Beginning with an untrained DL algorithm, labeled examples are fed into the network for prediction. The predictions and labels are compared, and any differences are backpropagated to update the convolutional filter coefficients that comprise the DL network. (b) By repeating this process with many labeled examples, it is possible to train the algorithm to predict attributes for data that were not used during training.

Methods

Supervised learning

Like previous successful DCNs (Bittel et al., 2015; Long et al., 2015), a supervised learning approach is used in which it is assumed that for every data point x_i, there exists a corresponding label y_i. In the supervised learning framework, the objective is to minimize the total error across the N_{train} samples via

$$E = \sum_{i=1}^{N_{\text{train}}} L(f(x_i;\theta), y_i), \tag{1}$$

where L measures the loss (or error) between the predictions generated by the DCN model $y_i = f(x_i;\theta)$ and the ground truth label y_i. This assumes that accurate and consistent ground truth labels are available.

Generating labels for supervised learning problems is typically a time-intensive and arduous task. For instance, in the ImageNet data set (Russakovsky et al., 2015), more than 14 million images have been hand annotated by multiple users, with the resulting label only kept if consensus exists between experts. Our task is relatively easy in this regard because we have synthetically generated seismic data sets before and after adding noise and can compute proprietary spectral decomposition attributes as labels using existing techniques.

DCN training methodology

The DCNs are trained with thousands of 3D synthetic seismic data sets, shown in Figure 1a as “labeled data.” The main benefit of this approach is that the trained DCN is more likely to generalize to new field-acquired seismic surveys, is less prone to overfitting, and provides insights for cases in which the DCN has difficulty making accurate predictions. Once trained, the DCN can be applied to an unseen field-recorded survey, shown as “unlabeled data” in Figure 1b. One of our DCNs was trained to remove random noise from seismic and another to perform a regression-style task of replicating a complex, proprietary spectral decomposition workflow. We trained both DCNs with 14,000 synthetic seismic

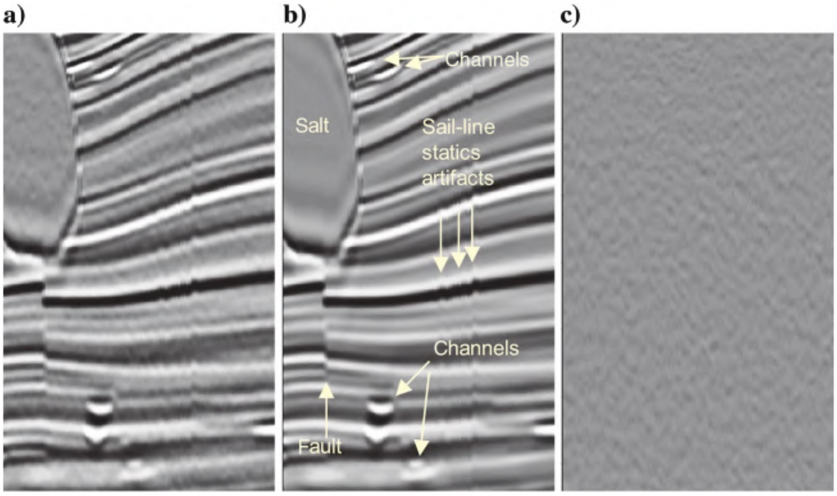

Figure 2. (a) Example cross section from one of many synthetic seismic cubes. (b) Noise-free synthetic seismic used as labels during denoising DCN training. (c) Pseudorandom background noises that the DCN is trained to remove.

volumes in the training set and 6000 volumes in the validation set.

Synthetic seismic modeling methodology

Tasks are ideally suited for DL when humans can perform the task in a few seconds, but need to repeat it thousands or millions of times. Synthetic seismic modeling is performed to train DCNs, so the goal is to create synthetic seismic that is difficult for human observers to distinguish from field-acquired data when only given a few seconds to make a comparison. This philosophy is used to guide requirements for how realistic the synthetic seismic needs to be and the methods used to generate it.

To generate a large, accurately labeled training data set, we leveraged synthetic data generation in which we have full control of how labels are created from known models of the subsurface. A similar approach was taken in Araya-Polo et al. (2017) on synthetic prestack seismic data. We build on his work to generate numerous and diverse examples of the subsurface to ensure predictions generalize to new, unseen geologic settings. The geology (e.g., layering, faulting, stratigraphic variations, and rock properties) is generated randomly, within realistic ranges based on subsurface measurements in field-recorded data (Marrett and Allmendinger, 1991; Schlische et al., 1996; Gross et al., 1997; Nixon, 2013), and also includes simulation of acquisition and processing noises to create a diverse data set meant to include most geologic and data quality regimes seen in real seismic. One example synthetic seismic cross section from the many thousands we used to train our DCNs is shown in Figure 2. It shows simulated seismic (Figure 2a), which in this randomly realized cube includes simulated faults, channels, salt, and residual sail-line static artifacts. Noise-free values used as labels to train a denoising DCN application are shown in Figure 2b. Pseudorandom noises that the DCN should learn to remove are shown in Figure 2c. The noises are never used but are included here for visual comparison.

More than 20,000 poststack synthetic seismic data sets have been generated to date. A convolutional model is used. A layered model of the earth is first created. It is then convolved with a band-limiting wavelet. Different layers are defined by variations in acoustic impedance, though we also include smaller variations in acoustic impedance within layers and in layer thicknesses. Geologic features such as channels and salt are inserted by replacing voxels in the synthetic 3D cube in which those features are present. Voxel amplitudes for pseudorandom noise are drawn randomly from an exponential distribution. Results are filtered with a band-pass filter and lateral mixing filters to generate spatial and temporal correlations. Pseudorandom noises are created separately so that synthetic cubes are available with noise added and as noise-free synthetic seismic cubes. Noise-free seismic cubes are used as labels for the denoising DCN.

This process produces synthetic data sets with unique combinations of simulated basement structure, faults, channels, salt diapirs, erosional unconformities, signal-to-noise ratio, bandwidth, acquisition artifacts, layer properties and dips, and seafloor depths. The synthetic seismic cubes are small with dimension of (300, 300, and 800) samples, but could be any size. Typical percentages for geologic features (faults, unconformity surfaces, etc.) are 0%–3% of voxels. Including additional features can be impactful even in small amounts. For example, simulating sail-line statics or common post-processing normalization techniques (e.g., automatic gain control) is useful when applied to as few as 5%–10% of the synthetic models. Sampling rates, bandwidth, and signal-to-noise ratio are chosen randomly for

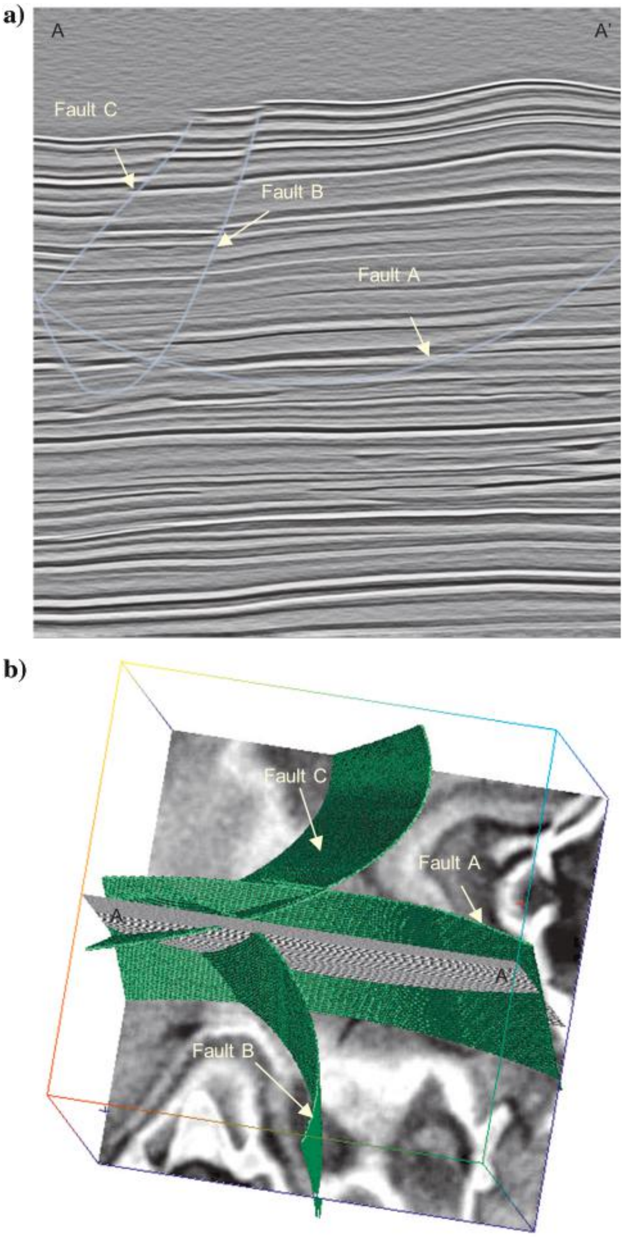

Figure 3. Fault A has the strike nearly parallel to the seismic cross section. Faults B and C have the strike orthogonal to the seismic cross section. Note the increased difficulty in identifying the fault in the near-parallel strike case relative to the orthogonal strike case.

each data set and cover the full range observed in field-acquired data. Some of the features are simulated with more sophistication than others. For example, faulting patterns are simulated for en-echelon faults, horst and grabens, and growth faults. Some faults are simulated as a single, curving surface, whereas others include simulation of fault zones, where closely spaced partial displacements collectively form a fault. Layer thickness and reflectivity are simulated using random variations at multiple lateral scales such that amplitude and thickness maps are subjectively similar to amplitude extractions from field-acquired seismic. Whereas some sophistication is included for faults, salt diapirs are simulated very crudely. They do not currently include any interaction with nearby faults, nor does the simulated salt influence nearby bed thickness. Sophistication is improved for each geologic feature only as necessary to improve DCN training, or as required for use with more DL tasks.

To further expand the impact of synthetic data on training, we leverage common DL data-augmentation techniques (Taylor and Nitschke, 2017). This includes making mirror-image versions of the labeled data in the x-y spatial dimensions and randomly stretching and squeezing by as much as 25% via interpolation along all three dimensions. Stretching and squeezing is constant for the x- and y-axes of each cube, so that only minor variations in relative scales of geologic features are introduced. This mimics field data, for example, where trace spacing in one survey is 25×25 m and another has 37.5×50 m $(25/25 = 1,\ 50/37.5 = 1.33)$. Slowly varying vertical stretching and squeezing is also applied to the synthetic seismic. It generally compresses the shallow section and expands the deep section to simulate time-to-depth conversion, and follows patterns observed in field-acquired seismic. Augmentation is done to generate a wider variety of geologic geometries and to include a broad corpus of spatial and depth-wise effective sampling rates. The resulting synthetic seismic cubes are used as forward models of recorded, fully processed seismic for the simulated subsurface.

Novel deep learning architecture: “WiSeNet”

For attribute computations and segmentation tasks, the 3D nature of the underlying geology dictates that a 3D DCN should offer superior performance compared to the 2D approach that is applied in image recognition tasks. A good illustration for retaining higher dimensions is to study the intersection of faults on seismic cross sections. The intersection of a fault plane oriented nearly parallel to a 2D seismic cross section will be nearly horizontal (see fault A in Figure 3), making it difficult to distinguish from reflecting layers. By contrast, a dip-oriented fault will be nearly vertical and easy to recognize (see the faults labeled B and C in Figure 3). It is more natural to detect (3D) faults by analyzing them based on their 3D expression in seismic subcubes rather than on 2D cross sections that might not be optimally oriented.

To overcome common scenarios that negatively impact 2D DCN performance such as in Figure 3, we desire to implement a 3D DCN. It consists of convolutional and pooling layers, each with 3D kernels. Ideally, such a network will be able to ingest an entire 3D seismic survey as a single input, or at least a 3D array large enough to capture the geologic information relevant to the prediction within the receptive field of the DCN. We want our DCN implementation to output predictions on a voxel-wise basis for the input 3D subcube rather than perform a single inference that is understood to apply only at the center voxel. To do this, we use a fully convolutional architecture (Long et al., 2015; Ronneberger et al., 2015; Cicek et al., 2016). Such segmentation architectures are far more efficient than patch-based, center-voxel classification approaches (Waldeland et al., 2018) and provide spatially more consistent predictions (Long et al., 2015).

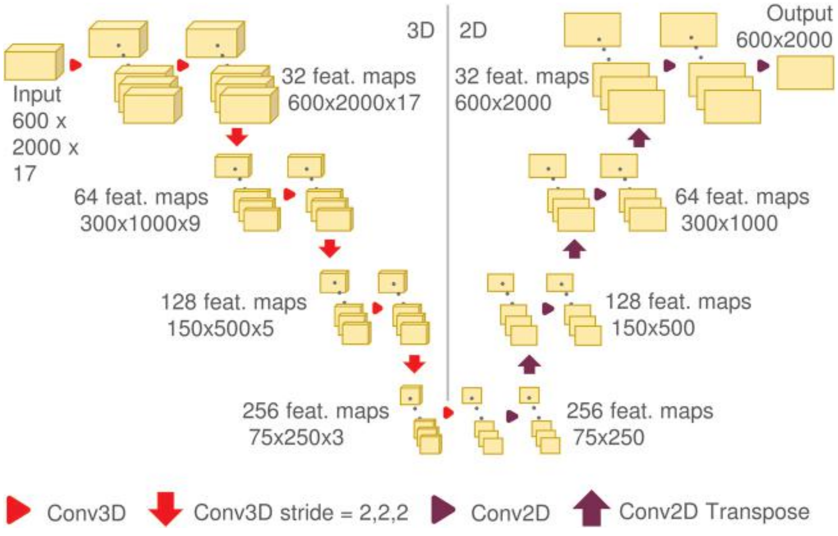

Figure 4. WiSeNet 3D-to-2D fully convolutional architecture comprised of a slab input (with one dimension being relatively small in size); 3D encoder, which collapses the small dimension down to a singleton dimension; and a 2D decoder, whose output corresponds spatially to the central slice of the input.

In practice, GPU memory limitations significantly restrict the size of 3D input that can be ingested by a DCN. At the time of writing, modern GPUs have up to 32 GB of memory (such as in the NVIDIA V100). Under such limitations, we were able to ingest a subcube as large as 128×128×128 voxels in a 3D fully convolutional architecture. Earlier attempts to apply DL to seismic interpretation (Gramstad and Nickel, 2018; Guo et al., 2018; Ma et al., 2018; Maniar et al., 2018; Pham et al., 2018; Shi et al., 2018; Waldeland et al., 2018; Wang et al., 2018; Wu et al., 2018; Zhao and Mukhopadhyay, 2018; Xiong et al., 2018) either chose 2D segmentation architectures or small 3D patch-based classification architectures due to GPU hardware limitations.

To cope with GPU memory constraints, we use a novel, memory-efficient 3D-to-2D fully convolution network similar to the U-Net architecture (Ronneberger et al., 2015; Cicek et al., 2016). Our encoder-decoder architecture uses a 3D encoder to capture 3D information in a large enough receptive field to convey the pertinent geologic context, followed by a 2D decoder, which provides pixel-wise segmentation on the central slice of the initial 3D input. The input to our DCN is a 3D slab with relatively large inline (or crossline) and vertical extents and a small crossline (or inline) extent; a typical input extent is 600 × 2000 voxels in the inline (or crossline) and vertical directions and 17 voxels in the crossline (or inline) direction. Importantly, the DCN architecture requires the small dimension to have an odd number of slices so that the central slice is explicitly defined.

The network's 3D encoder collapses the small dimension into a singleton dimension via a series of stride-2 convolutional or max pooling layers; the singleton dimension is then removed, and the 2D feature maps (corresponding to the central slice) are decoded by a series of 2D convolutional and transpose convolutional layers. In the encoding step, the 3D convolutions consist of 3 × 3 × 3 kernels, increasing from 32 to 256 feature maps. Each transposed convolution in the decoding step consists of 3 × 3 kernels, decreasing from 256 to 32 feature maps. The final 1 × 1 convolution provides pixel-wise predictions on the center slice. All convolutional operators (except for the final) in the network are followed by a rectified linear unit (ReLu) activation function, as is conventionally found in most modern DCNs. One implementation of our architecture is depicted in Figure 4. We note that a somewhat different 3D-to-2D architecture (where the vertical direction is collapsed) has been used in the medical imaging field (Rafiei et al., 2018). Details and mathematical derivations for (transposed) convolutions, max pooling, and ReLu can be found in Goodfellow et al. (2016).

To illustrate the performance uplift provided by our WiSeNet (Wide Seismic Network) 3D-to-2D architecture, we compute inference on a single inline of size 600-by-2000 with the architecture in Figure 4 and a 3D DCN center-voxel network with the same architecture as the encoding portion of our network. The 3D DCN takes approximately 100 ms to predict the center voxel of a single subcube using an NVIDIA V100 GPU. For 1.2 million voxels comprising the 600-by-2000 inline, this takes approximately 33 h. Our WiSeNet 3D-to-2D network, by contrast, computes inference in approximately 1–2 s, which is four to five orders of magnitude faster than the 3D DCN center-voxel network.

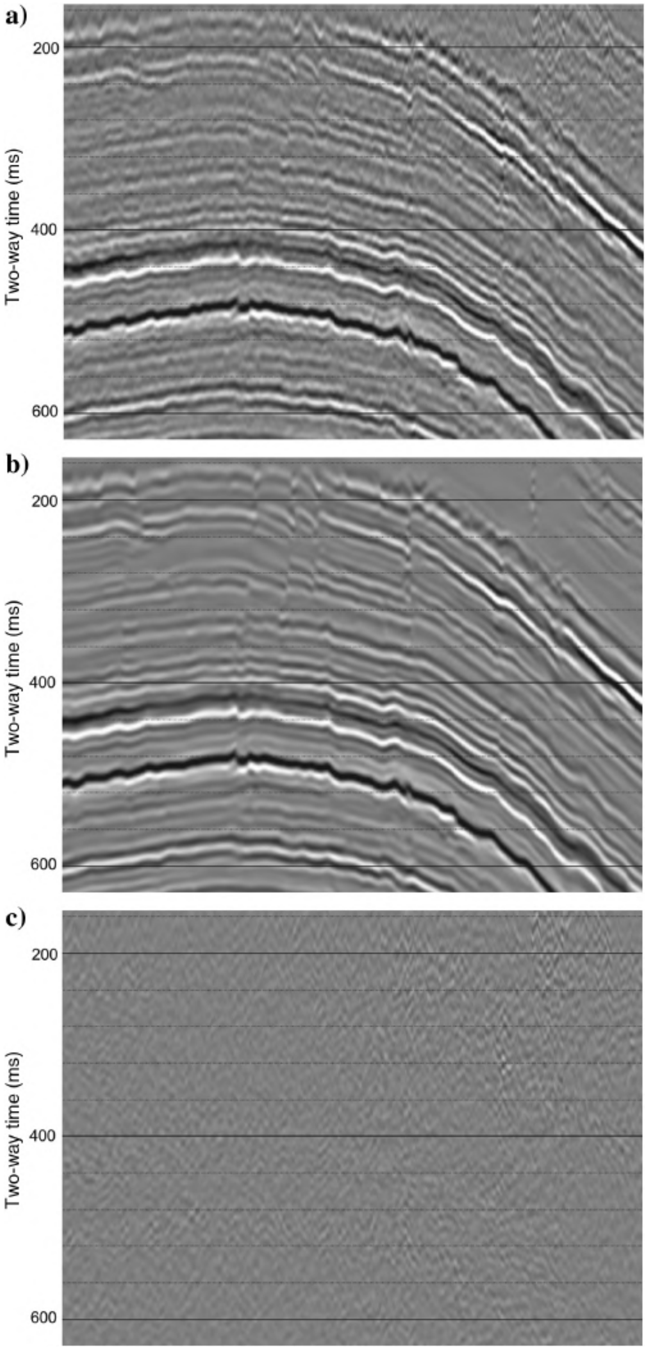

Figure 5. (a) Field-recorded seismic from Southwest Texas, USA. (b) The same data as Figure 5a after applying WiSeNet denoising. (c) Removed noises computed as the difference between (a and b). The WiseNet DCN separates geology from pseudorandom background noises using a 3D learning model. It provides inference throughput almost as fast as for 2D DCN architectures.

Results and discussion

Deep learning applied to attribute computation

We use our 3D-to-2D WiSeNet architecture for denoising seismic data. Providing 3D input data helps the DCN learn a representation of the subsurface that distinguishes well between geologic patterns and noises. Figure 5 shows an example on field-recorded data. The input data are shown in Figure 5a, WiSeNet output is in Figure 5b, and the removed noises computed as the difference between input and output, which is never used except for human visual inspection in Figure 5c. Note how fault planes are much more visible after denoising and how the removed noises have no visible correlations with the locations of faults or horizons.

We use DL to predict the expected values of attributes that are otherwise computed through a long, computationally intense workflow to produce a proprietary type of spectral decomposition (shell spectral decomposition [SSD]). Several distinguishing features of SSD compared with spectral decomposition techniques typically used in the oil industry (see Sinha et al., 2005) are that the spectral bands are estimated from the data and that it removes background trends in bandwidth versus depth. These qualities make it computationally expensive, and a good candidate for our DCN regression approach. The DL version (shell spectral decomposition using deep learning [SSDDL]) focuses only on providing an uplift to computation speed because it is designed to replace the conventionally computed attributes as closely as possible. Because spectral decomposition is often thought of as a 1D vertical technique, we use 2D DCNs for SSDDL.

For the SSDDL computation, a supervised DCN was trained using synthetically generated, normalized seismic as input with similarly normalized SSD as labels, computed using a conventional workflow. Field-recorded data could have been used, but we chose to use synthetically generated data that were already available in large quantities. The network learns the relative strength of the red, green, and blue components in different frequency bands by examining the example grayscale inputs and their RGB labels. For each training batch, a labeled 2D seismic profile is evaluated. The labels are from the same 2D seismic profile but have an extra dimension to accommodate the red, green, and blue color channels.

Our DL network uses a custom architecture that includes 2D convolutional layers, batch-normalization layers, rectified linear unit layers, with downsampling, and upsampling like that used by the fully convolutional architecture of Long et al. (2015). No softmax layers are included because our aim is regression style matching of floating-point values.

Our conventional SSD computations include background trend removal that is sensitive to changes in bandwidth versus depth. SSDDL does a good job of replicating this process without explicitly being trained to do so (Griffith, 2019). We believe that the size of the 2D network's receptive field is critical to learning the background trend and separating it from geologically important variations in spectral content, for example, to map "temporal bed thickness and geologic discontinuities," as described by Partyka et al. (1999). The receptive field measures the extent of input data that affects a predicted voxel value. In our DCN, it is composed of the lateral extent and the depth extent. Hyperparameter testing emphasized optimization of the receptive field by adjusting convolutional kernel sizes, number of convolutions per layer, plus the number of DCN layers. Hyperparameters were chosen based on validation losses during training plus consistency of error values between denoised results and desired labels. Consistency was measured using the standard deviation of root-mean-

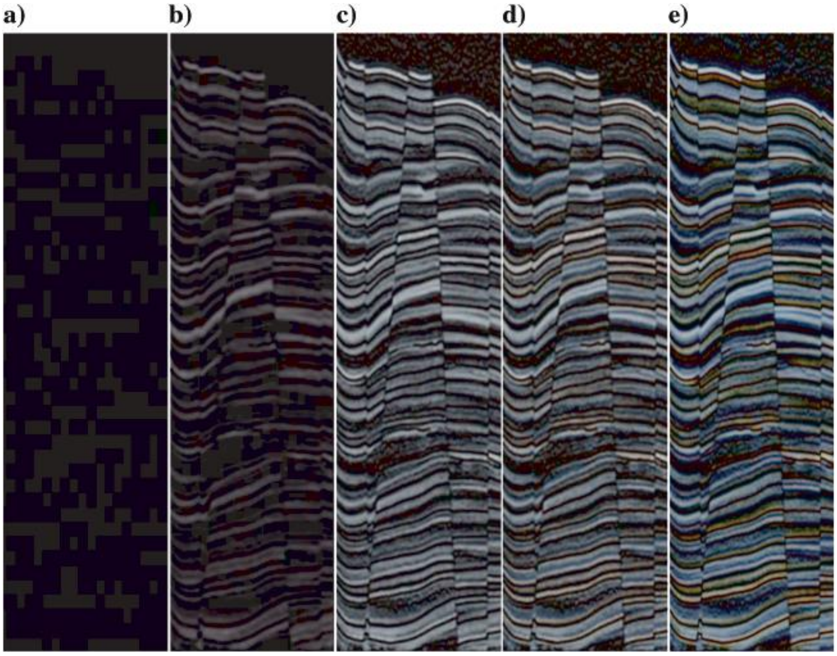

Figure 6. Example of SSDDL predicted for partially trained DCN after a different number of training epochs. Panel (a) is from the start of training, whereas (e) is from a network that is almost fully trained, plus three intermediate results. Note how early results focus on luminance and later training steps learn to associate colors with variations in seismic spectral content.

square errors across sliding 2D windows, measured on inference results for each set of hyperparameters. Good results are obtained using convolutional kernel sizes and numbers of layers that result in receptive fields on the order of 1 × 1 km (lateral distance by depth range).

It is instructive to observe intermediate results while the DCN trains. Figure 6 shows SSDDL predictions after selected partial training epochs. At the start of training, the DCN has not learned anything and a black image is predicted (Figure 6a). Then, the DCN learns a grayscale, low-resolution version of the input, focused on the brightest reflectors. It progresses to learn a more detailed grayscale version of the desired SSD estimate (Figure 6c), then to discriminate the thickest layers (blue), then finally to include thinner red and green layers (Figure 6e). Volumetrically, there are more blue voxels because they represent the thickest layers, with fewer green and red voxels, respectively. The progression for accurately associating colors to layers during training mirrors the volumetric proportions of colors in the labeled input examples, at least in a qualitative manner. This observation leads us to conclude that it takes a lot of data and many training epochs to accurately predict attribute values that are rarely observed. This is not unlike class imbalance problems in semantic segmentation DCNs for predicting features that occur infrequently compared with other features being predicted.

It is important to assess the quality of the output attribute computation quantitatively via DL loss functions and qualitatively based on expert human judgement. Mean-squared error is typically used in DL loss functions for regression tasks. We note that mismatches between labels and predictions were most often observed where the synthetic example data undersamples specific geometries of seismic layering or unusual layer thicknesses.

An attractive extension is to make spectral decomposition respond to stratigraphic layer thicknesses instead of vertical bed thickness. With conventional code, it would be tedious and computationally expensive to measure bed dips and apply filters perpendicular to dip. It could be enabled with DL by augmenting input examples, so that similar spectral decomposition values exist for a wide range of layer dip angles. Rotating the input 2D labeled examples accomplishes this in a few lines of code.

Once we selected a trained DCN, it was used to perform inference on many field-acquired seismic data sets that were not part of the training set. Numerous expert interpreters were asked to participate in blind-test comparisons. They found it nearly impossible to distinguish between attribute results for DCN and conventionally computed versions. Figures 7 and 8 illustrate the close comparison between SSD and SSDDL for different portions of a seismic traverse and for an example stratal slice from the Maverick Basin in Southwest Texas, USA. Figures 7a and 8a show results from SSDDL, whereas Figures 7b and 8b show results from traditional computation. SSD and SSDDL stratal slices in Figure 8a and 8b have almost identical blue coloring for patch reefs in the upper-left corner of the maps and cyan and white coloring for larger reefs in the bottom of the maps, for example. For a more complete description of the geology, see Aconcha et al. (2008).

DCNs can provide a very significant speedup for computing attributes when they involve complex workflows. The times vary widely based on hardware platform, with best results for platforms that include several recent vintage GPUs. Figure 9 shows relative performance for conventional computations on a Linux blade using multiprocessing on 10 CPUs versus DL on a similar Linux blade with four NVIDIA K80 GPUs. Approximate compute times for an 8 bit 1 Gb seismic cube are 1 h conventionally versus 3 min for our DCN approach, most of

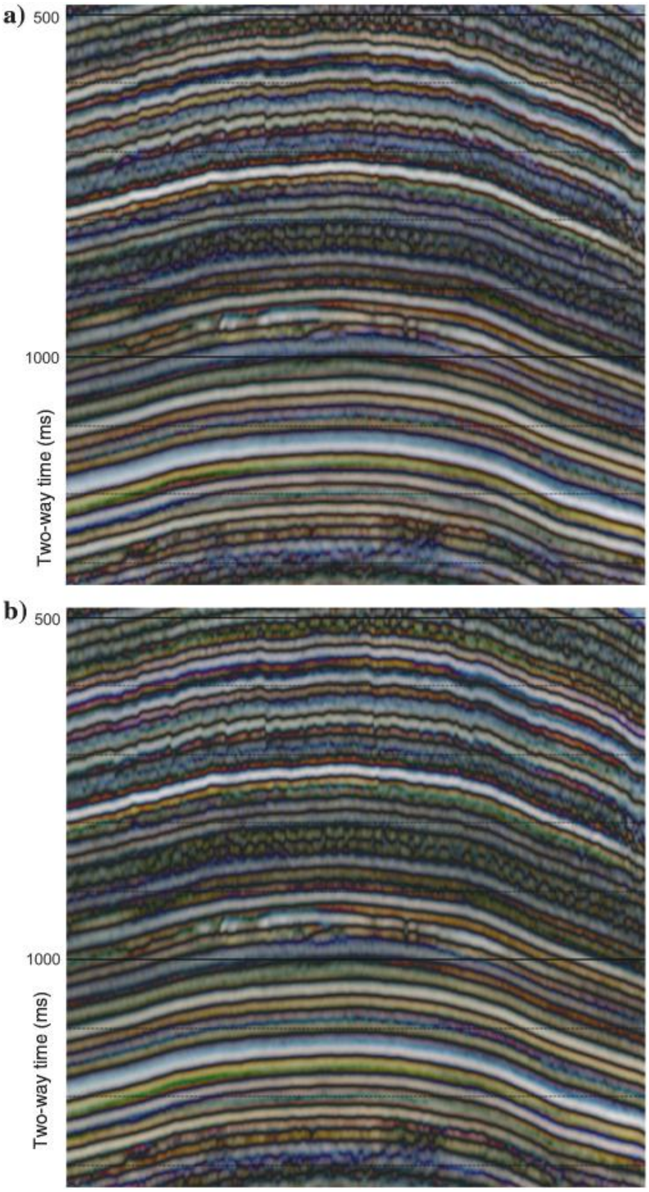

Figure 7. (a) A traverse from Southwest Texas, USA, in which a compute-intensive proprietary type of spectral decomposition (SSDDL) is computed using a trained DCN. (b) SSD computed for comparison using the traditional workflow. The similarity between (a and b) demonstrates the good performance of the trained DCN.

which is for reading and normalizing the input data in preparation for the DCN inference step. This speedup allows contemplation of real-time attribute computation, which can be a key enabler for interpreters to use the attributes to better understand geology.

An additional advantage of using DL for attributes, whose impact is easy to overlook, is that the code is short and simple compared with conventionally coded attribute computations. The bar chart in Figure 9 shows that DCNs provide a large compute time reduction, and that when built using open-source Python packages, such as Keras and TensorFlow, there is a much smaller code base to maintain compared with traditional computation of attributes. Figure 9 does not include code to generate synthetic labeled seismic because DCN training is performed once, but the trained DCN can be applied to many field-recorded seismic data sets.

This discussion has focused on DL applied to denoising and SSD, but we have built DL networks for several other seismic attributes and data-cleaning applications. Our vision includes replacing more conventional seismic attributes and processing methods with DL in the future.

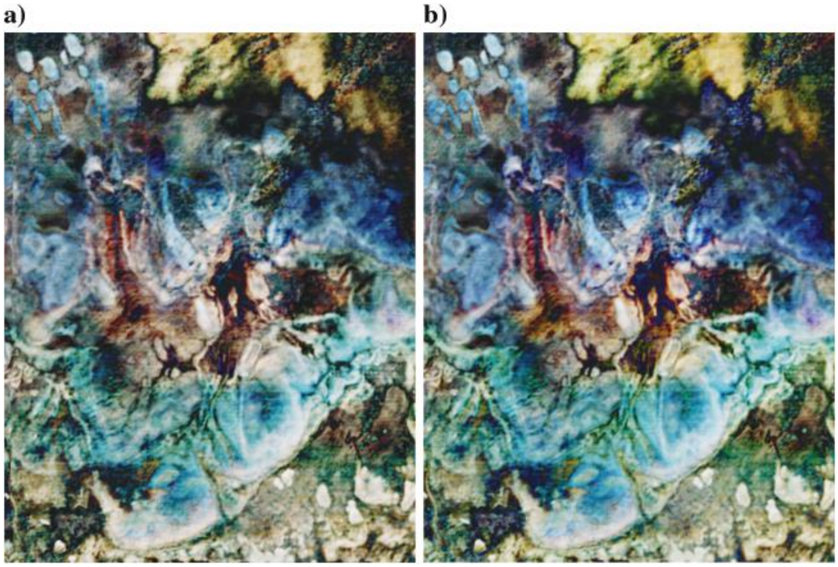

Figure 8. Color stratal slices from Southwest Texas showing shoals, patch reefs, and carbonate buildups. Panel (a) is computed using a trained DCN (SSDDL). (b) The same stratal slice computed using the traditional workflow (SSD). The similarity between (a and b) demonstrates the good performance by the trained DCN.

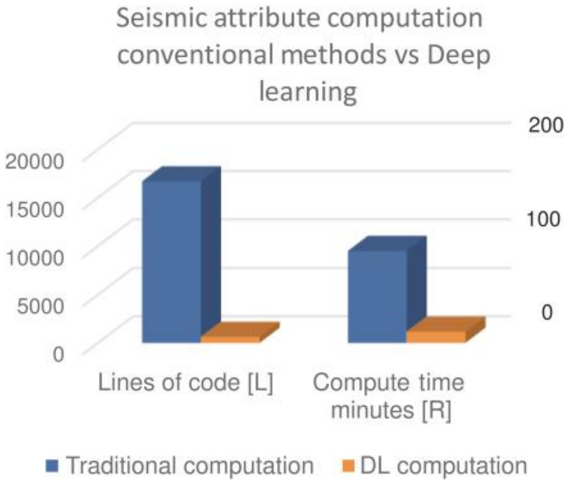

Figure 9. The DL inference compute times are reduced, and the code is much smaller compared with traditional computation methods.

Conclusion

Our work demonstrates that DCNs can be trained for subsurface geoscience applications to achieve superior results with respect to accuracy and compute performance. It shows that including synthetically generated seismic data helps a trained DCN to generalize to many different seismic surveys if attention is given to simulating the subsurface with accurate labels and sufficiently realistic depositional and structural models as well as inclusion of simulated seismic acquisition and processing artifacts.

We showed a novel 3D-to-2D DCN architecture called WiSeNet that extends 2D image-based fully convolutional network approaches to 3D segmentation. It can learn 3D patterns in the input data as illustrated by a denoising DCN, yet it can still deliver inference throughput at rates that are orders of magnitude faster than center-voxel 3D classifiers. Further, we shared a successful example of regression style inference for replicating multistep SSD workflows using a fraction of the compute time.

The authors are committed to the use of DL for solving complex subsurface geoscience problems. We believe that more innovative uses for DL lie ahead than have already been solved. We gratefully acknowledge Shell International Exploration and Production BV for sponsoring our work, and for allowing us to publish it.

Acknowledgments

We owe a debt of gratitude to each of the following people (listed alphabetically), who have made significant contributions to the development of our DL tools and the ideas that drive their development and implementation: R. Ameerali, M. Araya-Polo, D. Cha, A. Chandran, J. Cook, P. Devarakota, O. Falivene, J. Hou, D. Knott, G. Lortzer, K. Sisco, N. Suurmeyer, J. Week, and K. Wright. We thank the three anonymous reviewers whose comments helped to improve and clarify this paper.

Data and materials availability

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

References

Aconcha, E. S., C. Kerans, and H. Zeng, 2008, Seismic geomorphology applied to Lower Glen Rose patch reefs in the Maverick Basin, southwest Texas: Gulf Coast Association of Geological Societies Transactions, **58**, 3–23.

Araya-Polo, M., T. Dahlke, C. Frogner, C. Zhang, T. Poggio, and D. Hohl, 2017, Automated fault detection without seismic processing: The Leading Edge, **36**, 208–214, doi: 10.1190/tle36030208.1.

Bittel, T., V. Kaiser, M. Teichmann, and M. Thoma, 2015, Pixel-wise segmentation of street with neural networks: ArXiv preprint 1511 00513, https://arxiv.org/abs/1511.00513.

Casamitjana, A., S. Puch, A. Aduriz, and V. Vilaplana, 2017, 3D convolutional neural networks for brain tumor segmentation: A comparison of multi-resolution architectures: ArXiv preprint 1705 08236, https://arxiv.org/abs/1705.08236.

Cicek, Ö., A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, 2016, 3D U-net: Learning dense volumetric segmentation from sparse annotation: ArXiv preprint 1606 06650, https://arxiv.org/abs/1606.06650.

De Brebisson, A., and G. Montana, 2015, Deep neural networks for anatomical brain segmentation: ArXiv preprint 1502 02445, https://www.researchgate.net/publication/272195333_Deep_Neural_Networks_for_Anatomical_Brain_Segmentation.

Erden, B., N. Gamboa, and S. Wood, 2017, 3D convolutional neural network for brain tumor segmentation, Stanford CS 231N, http://cs231n.stanford.edu/reports/2017/pdfs/526.pdf.

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

Gramstad, O., and M. Nickel, 2018, Automated interpretation of top and base salt using deep-convolutional networks: 88th Annual International Meeting, SEG, Expanded Abstracts, 1956–1960, doi: 10.1190/segam2018-2996306.1.

Griffith, D. P., 2019, Method for multi-dimensional geophysical data visualization: U.S. Patent Application 20190080507A1.

Gross, M. R., G. Guiterrez-Alonso, T. Bai, M. A. Wacker, K. B. Collinsworth, and R. J. Behl, 1997, Influence of mechanical stratigraphy and kinematics on fault scaling relations: Journal of Structural Geology, **19**, 171–183, doi: 10.1016/S0191-8141(96)00085-5.

Guo, B., L. Liu, and Y. Luo, 2018, A new method for automatic seismic fault detection using convolutional neural network: 88th Annual International Meeting, SEG, Expanded Abstracts, 1951–1955, doi: 10.1190/segam2018-2995894.1.

Hajabdollahi, M., R. Esfandiarpoor, S. M. R. Soroushmehr, N. Karimi, S. Samavi, and K. Najarian, 2018, Low complexity convolutional neural network for vessel segmentation in portable retinal diagnostic devices: ArXiv preprint 1802 07804, https://arxiv.org/abs/1802.07804.

He, K., X. Zhang, S. Ren, and J. Sun, 2015, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification: ArXiv preprint 1502 01852, https://arxiv.org/pdf/1502.01852.pdf.

Huang, L., X. Dong, and T. E. Clee, 2017, A scalable deep learning platform for identifying geologic features from seismic attributes: The Leading Edge, **36**, 249–256, doi: 10.1190/tle36030249.1.

Kamnitsas, K., C. Ledig, V. F. J. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, 2017, Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation: ArXiv preprint 1603 05959v3, https://arxiv.org/abs/1603.05959.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012, ImageNet classification with deep convolutional networks: Advances in Neural Information Processing Systems, **60**, 1097–1105.

Long, J., E. Shelhamer, and T. Darrell, 2015, Fully convolutional networks for semantic segmentation: ArXiv preprint 1411 4038, https://arxiv.org/abs/1411.4038.

Ma, M., X. Ji, M. BenHassan, and Y. Luo, 2018, A deep-learning method for automatic fault detection: 88th Annual International Meeting, SEG, Expanded Abstracts, 1941–1945, doi: 10.1190/segam2018-2984932.1.

Maniar, H., S. Ryali, M. S. Kulkarni, and A. Abubakar, 2018, Machine-learning methods in geosciences: 88th Annual International Meeting, SEG, Expanded Abstracts, 4638–4642, doi: 10.1190/segam2018-2997218.1.

Marrett, R., and R. W. Allmendinger, 1991, Estimates of strain due to brittle faulting: Sampling of fault populations: Journal of Structural Geology, **13**, 735–738, doi: 10.1016/0191-8141(91)90034-G.

Mernagh, M., and M. Pendse, 2017, Automatic brain tumor segmentation, Stanford CS 231N, http://cs231n.stanford.edu/reports/2017/pdfs/502.pdf.

Nixon, C. W., 2013, Analysis of fault networks and conjugate systems: Ph.D. dissertation, University of Southampton.

Partyka, G., J. Gridley, and J. Lopez , 1999, Interpretational applications of spectral decomposition in reservoir characterization: The Leading Edge, **18**, 353–360, doi: 10.1190/1.1438295.

Pham, N., S. Fomel, and D. Dunlap, 2018, Automatic channel detection using deep learning: 88th Annual International Meeting, SEG, Expanded Abstracts, 2026–2030, doi: 10.1190/segam2018-2991756.1.

Poudel, R. P. K., P. Lamata, and G. Montana, 2016, Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation: ArXiv preprint 1608 03974, https://arxiv.org/abs/1608.03974.

Rafiei, S., E. Nasr-Esfahani, S. M. R. Soroushmehr, N. Karimi, S. Samavi, and K. Najarian, 2018, Liver segmentation in CT images using three dimensional to two dimensional fully convolutional networks: ArXiv preprint 1802 07800, https://arxiv.org/abs/1802.07800.

Ronneberger, O., P. Fischer, and T. Brox, 2015, U-net: Convolutional networks for biomedical image segmentation: ArXiv preprint 1505 04597, https://arxiv.org/abs/1505.04597.

Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.

C. Berg, and L. Fei-Fei, 2015, Imagenet large scale visual recognition challenge: ArXiv preprint 1409 057, https://arxiv.org/pdf/1409.0575.pdf.

Salehi, S. S. M., D. Erdogmus, and A. Gholipour, 2017, Auto-context convolutional neural network for geometry-independent brain extraction in magnetic resonance imaging: ArXiv preprint 1703 02083, https://arxiv.org/abs/1703.02083.

Schlische, R. W., S. S. Young, R. V. Ackerman, and A. Gupta, 1996, Geometry and scaling relations of a population of very small rift-related normal faults: Geology, **24**, 683–686, doi: 10.1130/0091-7613(1996)024<0683:GASROA>2.3.CO;2.

Shi, Y., X. Wu, and S. Fomel, 2018, Automatic salt-body classification using deep-convolutional neural network: 88th Annual International Meeting, SEG, Expanded Abstracts, 1971–1975, doi: 10.1190/segam2018-2997304.1.

Sinha, S., P. S. Routh, P. D. Anno, and J. P. Castagna, 2005, Spectral decomposition of seismic data with continuous-wavelet transform: Geophysics, **70**, no. 6, P19–P25, doi: 10.1190/1.2127113.

Taylor, L., and G. Nitschke, 2017, Improving deep learning using generic data augmentation: ArXiv preprint arXiv:1708.06020.

Waldeland, A. U., A. C. Jensen, L.-J. Gelius, and A. H. S. Scholberg, 2018, Convolutional neural networks for automated seismic interpretation: The Leading Edge, **37**, 529–537, doi: 10.1190/tle37070529.1.

Wang, Z., H. Di, M. A. Shafiq, Y. Alaudah, and G. AlRegib, 2018, Successful leveraging of image processing and machine learning in seismic structural interpretation: A review: The Leading Edge, **37**, 451–461, doi: 10.1190/tle37060451.1.

Wu, X., Y. Shi, S. Fomel, and L. Liang, 2018, Convolutional neural networks for fault interpretation in seismic images: 88th Annual International Meeting, SEG, Expanded Abstracts, 1946–1950, doi: 10.1190/segam2018-2995341.1.

Xiong, W., X. Ji, Y. Ma, Y. Wang, N. M. AlBinHassan, M. N. Ali, and Y. Luo, 2018, Seismic fault detection with convolutional neural network: Geophysics, **83**, no. 5, O97–O103, doi: 10.1190/geo2017-0666.1.

Zhao, T., and P. Mukhopadhyay, 2018, A fault-detection workflow using deep learning and image processing: 88th Annual International Meeting, SEG, Expanded Abstracts, 1966–1970, doi: 10.1190/segam2018-2997005.1.

Donald P. Griffith is a principal research geophysicist and data scientist at Shell. He has held a variety of positions in seismic processing, acquisition, and interpretation in his 38 year tenure. He is a global SME for seismic attributes. He served as a seismic interpreter for 20+ years developing numerous oil and gas fields in the Gulf of Mexico and offshore Canada, and he has drilled more than 40 wells. His current passion is augmenting seismic interpreters' skills by developing and deploying intelligent covisualization and DL tools.

S. Ahmad Zamanian received a B.S. (2005) in biomedical engineering from Johns Hopkins University, and S.M. (2007), E.E. (2014), and Ph.D. (2014) degrees from MIT, all in electrical engineering and computer science. From 2007 to 2010, he worked as a modeling and simulation engineer for Schlumberger, developing algorithms for drilling automation and downhole telemetry. In 2010, he joined the earth resources laboratory at MIT to conduct his doctoral research, where he investigated hierarchical Bayesian and graph theoretic approaches and algorithms for seismic imaging and other geophysical inverse problems. Since 2014, he has been working in R&D for Shell. He is a senior geophysics research data scientist at Shell International Exploration and Production Inc., where he works on applications of DL to subsurface interpretation. His research interests include the intersection (or perhaps union) of machine learning (particularly DL), seismic interpretation, geophysical inverse problems, and statistical inference.

Jeremy Vila received a Ph.D. (2015) in electrical and computer engineering in signal processing and machine learning from Ohio State University. He is a data science researcher at Shell, specializing in DL-based algorithms on geophysical and petrophysical data. He previously worked at MIT Lincoln Laboratory, applying machine learning for cybersecurity problems. His primary research interests include DL, compressed sensing, and Bayesian graphical networks.

John Solum received a B.S. from Utah State University and an M.S. and a Ph.D. from the University of Michigan. He is a senior structural geologist at Shell Technology Center Houston. Prior to joining Shell, he worked at Sam Houston State University and the U.S. Geological Survey.

R. David Potter received an honors geophysics B.S. from Mcgill University and a geophysics Ph.D. from UCLA. He joined Shell in 2006 and is currently the manager of Shell's hydrocarbon search engine research team in Houston.

Antoine Vial-Aussavy received an engineering degree from the ENSTA Bretagne in France (2012) and an M.Sc. in petroleum geophysics from Imperial College London (2013). He is currently working as an exploration geoscientist in Shell Houston; prior to that he occupied various positions within Shell in the UK and in the Netherlands.

Francesco Menapace received an M.Sc. (2004) in geology from Alma Mater Studiorum — University of Bologna. He joined Shell as exploration geoscientist in 2007, after having worked as a logging while drilling engineer for one of the Oil&Gas service providers. He is currently a business opportunity manager for Exploration Digitalization, based in Houston.