Automatic detection of diffraction-apex using fully convolutional networks

Thamiris Coellho[1,2]*, Lucas de M. Araújo[1,2], Tiago A. Coimbra[1], Martin Tygel[1], Sandra Avila[1,2], and Edson Borin[1,2]

[1]Center for Petroleum Studies (CEPETRO), University of Campinas (Unicamp), Brazil

[2]Institute of Computing (IC), University of Campinas (Unicamp), Brazil

This paper was prepared for presentation at the 16th International Congress of the Brazilian Geophysical Society, held in Rio de Janeiro, Brazil, August 19-22, 2019.

Abstract

Diffractions play a significant role in seismic processing and imaging since they can image structures smaller than the seismic wavelength, such as discontinuities, faults, and pinch-outs. The traveltime of a non-migrated stacked diffraction event typically has a hyperbolic shape around its apex, which collapses after a migration procedure. We can interpret such apex as the time image of a point diffractor. An essential problem in diffraction analysis is the detection of those apexes in the generally noisy data environment because their position and processing parameters (such as migration velocity) play an essential role in obtaining more reliable and accurate imaging results. In this work, we introduce a Fully Convolutional Network (namely, LeNet-5 FCN) to automatic detect diffraction apexes on real seismic data. To deal with the low amount of annotated data, we propose to use data augmentation (e.g., polarity inversion, automatic gain control, zoom) and ensemble strategies. By combining our LeNet-5 FCN with those strategies, we reached 91.2% average accuracy on three land seismic datasets.

Introduction

In physics literature, diffractions are events observed on measured data related to the scattering of the wavefront energy by a point or sharp edge, which may be small-scale heterogeneities or geometrical discontinuities (see, Goodman, 2004). In seismic wave propagation, diffraction events typically result from scattering on geological discontinuities, such as faults and pinch-outs. As well recognized in the seismic literature (e.g., Klem-Musatov, 1994), diffraction events can convey useful additional details to the ones obtained by reflection-based conventional processing. The information that is conveyed by diffraction events is a topic of active research, both in academic and industry For instance, Reshef and Landa (2009) proposes the use of diffraction events for local velocity analysis. In the context of velocity analysis. In the same way, Coimbra et al. (2013) introduce an approach for migration-velocity analysis based residual diffraction migration. Diffraction imaging is also employed for better understanding of natural fault distributions (Burnett et al., 2015), as well as for small-fault interpretations (Sturzu et al., 2014), Moreover, Santos et al. (2012) proposed a tomography based on diffraction traveltime. A comprehensive account of the theoretical and applied seismic diffration literature can be found in Klem-Musatov et al. (2016).

Due to their scattered nature, diffraction events are in general much weaker than reflection events. As a consequence, suitable seismic processing becomes necessary to produce so-called diffraction-only datasets (or D-section), in which, simultaneously, diffractions are enhanced and reflections attenuated. For our purposes, we consider a D-section, which predominantly contains diffraction information, constructed by a double-square-root (DSR) stacking operator (Faccipieri et al., 2016). In fact, we follow Coimbra et al. (2018) to obtain a zero-offset (ZO), stacked D-section by means of a simplified DSR operator with a suitable aperture for separation and enhancing diffractions. The method provides, besides the stacked diffraction events (in the shape of approximate hyperbolas), also the stacking velocity section. As well known that stacking velocity at the apex of the diffraction hyperbola coincides with the time-migrated velocity that point. Because of this fact, it is of value to find the apex position of diffraction events in the D-section, these being obtained by manual picking. However, manual picking turns out to be a time-consuming and error-prone task. In fact, even for 2D-acquisition datasets, there are hundreds of diffraction events with ambiguous regions, thus making picking a difficult endeavor.

Nowadays, convolutional networks are finding substantial advances in pattern recognition, in particular, prediction inference on a single, individual pixel (or sample). New methods are appearing for classification and semantic segmentation of objects in bi-dimensional images. A new technology, called fully convolutional networks (FCNs) (see, Long et al., 2015), is built from locally connected layers only. Moreover, such networks can work regardless of their original image size, without requiring any fixed number of units at any stage. In summary, there are two types of inference for an FCN. First, the down-sampling layer (pooling) that is used to extract and interpret the context, i.e., what we are seeing. Secondly, the up-sampling layer (unspooling) that is used to enable precise localization, i.e., where is the object to be seen. FCNs are capable of detecting local patterns in images due to its nature, while convolutional neural networks (CNNs), due to their fully connected layers, are more suited for recognizing global patterns. Finally, FCNs also have a computational advantage, since they can process images of arbitrary size, exploiting the locality of the convolutional operators.

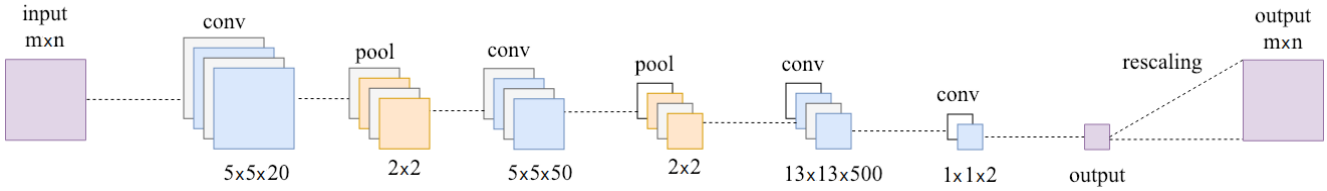


Figure 1: LeNet-5 FCN architecture with its convolutional (conv) and pooling (pool) layers. Given an $m \times n$ input, where $m, n \geq 64$, the output will have $\lceil \frac{m-63}{4} \rceil \times \lceil \frac{n-63}{4} \rceil$ dimension. This output can be re-scaled back to $m \times n$ dimension.

In this work, we propose a Deep-Learning based approach to devise an automatic tool to detect the apex of diffractions in seismic, real D-sections. More specifically, we introduce an FCN for apex detection. Our proposed FCN is implemented on the LeNet-5 architecture (LeCun et al., 1998), giving rise of what we called a LeNet-5 FCN (see, Figure 1). By combining LeNet-5 FCN with data augmentation and ensemble strategies, we reached, for three illustrative land seismic datasets, an average accuracy of 91.2%.

Methodology

For our experiments, an FCN receives as input an $m \times n$ dimension image, where $m, n \geq 64$, and produces as output a prediction window of $\lceil \frac{m-63}{4} \rceil \times \lceil \frac{n-63}{4} \rceil$ dimension. Training is done with 64×64 windows (i.e., regions extracted from the image), which outputs a 1×1 prediction (apex or non-apex), a training process quite similar to that of CNN. After we trained the model, the inference can be done with any window of size ≥ 64, which is also zero-padded. FCNs typically have up-sampling layers, but in our case a simple re-scale of the output to the original size is sufficient. This means that a prediction pixel in the output corresponds to a 4×4 region in the original data dimension. Since we are interested in detecting apex regions and not a single point, this granularity is sufficient. The strategy of combining an FCN architecture with the centrality of the label allows us to reduce the detection task to a classification task. The here employed centrality approach has been described in our previous work (Araújo et al., 2018).

To describe our proposed apex-detection algorithm, we make use of presently available deep-learning technologies, as described in, e.g., Goodfellow et al. (2016, Chapter 11). We trained the network for 100 epochs[1] with Stochastic Gradient Descent with a cross entropy loss function, momentum factor 0.9, starting learning rate 0.1, reduced to 0.01 after each epoch. We initialized the network weights with Glorot Uniform, and we regularized the network with a 0.6 dropout at the last hidden layer and 0.0005 weight penalty. All hidden layers used ReLU and the output layer Softmax activation functions.

The available training, validation and testing sets were annotated by a Geophysicist, picking apex and non-apex coordinates. For our application, we are interested in apexes of single diffractions without any interference from other diffractions. Such apexes constitute the apex or positive class. The non-apex or negative class is composed by apexes with crossings, diffraction tails, or not well-defined (background) apexes. We made an effort to maintain the negative class balanced with these three types of occurrences. Around each point in the D-volume, that is assumed to be a candidate apex, a 64×64 window is extracted, with that candidate point at its center. In this way, each class corresponds to a candidate apex that is located at the *center* of the window. The training, validation and testing sets are composed of these windows, each with its corresponding label (apex or non-apex).

[1]The final model was not necessarily the one from the 100^{th} epoch, but the one which presented the lowest error on the *validation* set.

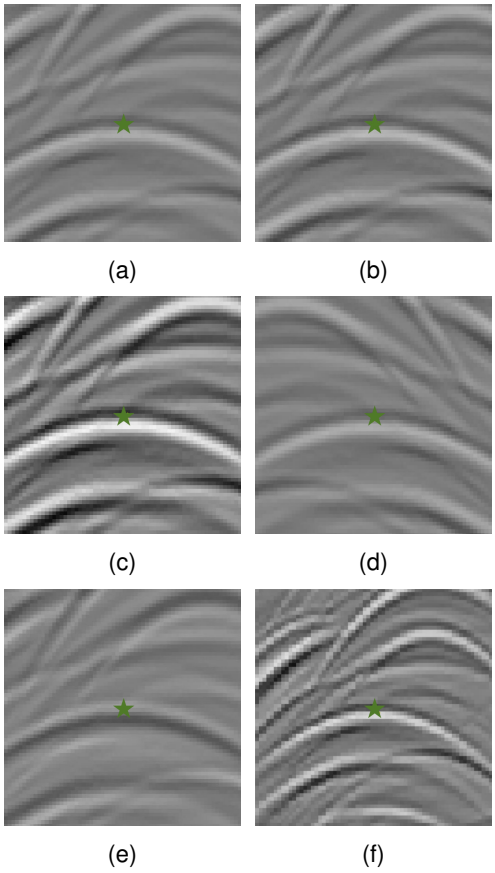


Figure 2: All augmentation techniques applied to a single sample. (a) original sample; (b) AGC; (c) clipping; (d) horizontal flip; (e) polarity inversion; (e) zoom from a 96×96 window re-scaled to 64×64. The centrality of the event is not lost in any transformation.

Figure 3 shows a seismic data after diffraction enhancement by the DSR technique. Green stars refer to D-volume data points annotated as apex (Region A) and red dots as non-apex. Non-apexes are shown in

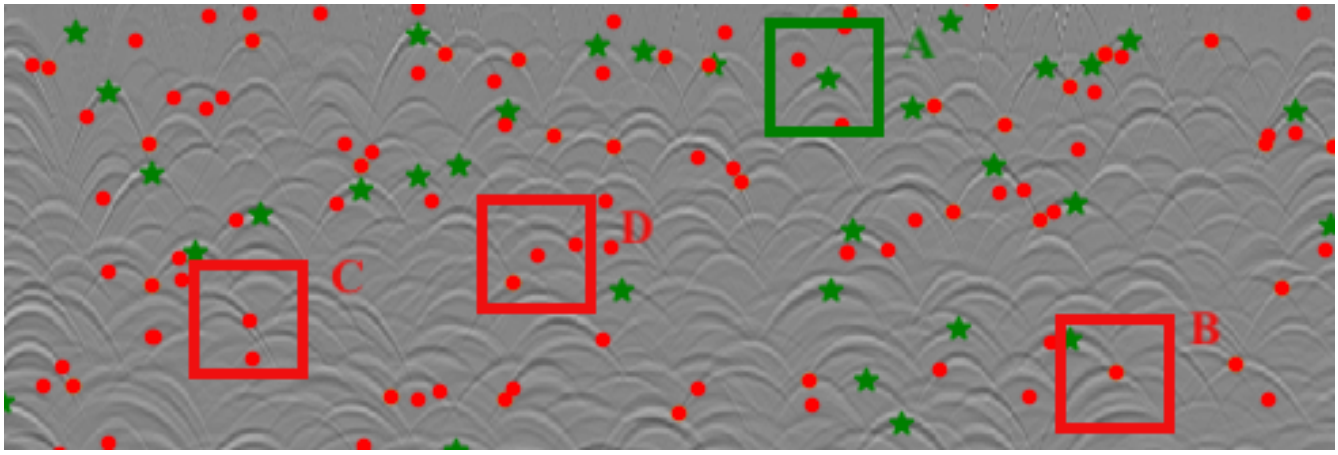


Figure 3: Diffraction panel with annotated coordinates by the Geophysicist. ★ marks apex coordinates (Region A) and ● non-apex coordinates (Regions B, C, and D). Particularly, non-apexes are shown in Region B (crossing), C (diffraction tail) and D (background). Windows of size 64×64 are illustrated.

Region B (crossing), C (diffraction tail) and D (background).

The Geophysicist annotated 1206 coordinates in 2D data from three real land datasets, acquired on Brazilian basins (Tacutu, Solimões, and Paraná). We used two of the three panels (D-sections) for training and validation (80%/20% split), and the remaining one for testing. We normalized each panel (i.e., seismic signal) to the $[-1; 1]$ interval.

Given that there were not much available annotated data for training, we used data augmentation (Paulin et al., 2014) to enhance variability during training:

(1) **Automatic Gain Control (AGC)**: enhances the signal within a sliding time window;

(2) **Clipping**: clips the signal in minimum and maximum values, also enhancing the signal;

(3) **Horizontal flipping**: flips the window around a center axis. This would be equivalent to inverting the acquisition axis. Rotation is a typical augmentation technique, but in the seismic context it would generate samples that have no physical meaning;

(4) **Polarity inversion**: inverts the signal phase, multiplying it by -1;

(5) **Zoom**: generates windows of different sizes and rescales them to 64×64. In this work, we used 32×32, 96×96 and 128×128, rescaled to 64×64 with nearest pixel interpolation. The latter technique was the one that provided the most improved variability in the training set. With data augmentation, we expanded the training data seven times. Figure 2 illustrates each augmentation technique.

As a final step to have a most accurate outcome, we used the so-called *ensemble* technique, which consists in voting the predicted class of a sample through the prediction of an ensemble of trained models. Often the ensemble is more precise than a single model, yielding more accurate results (Dietterich, 2000). Specifically, we constructed our ensemble with the *bagging* strategy (Breiman, 1996), training nine models with the same LeNet-5 FCN architecture, each from a set of 80% of randomly selected samples from the original training set. Although all models have the same architecture, each will have its unique set of weights due to randomness in the initialization, optimization process and training set.

Results

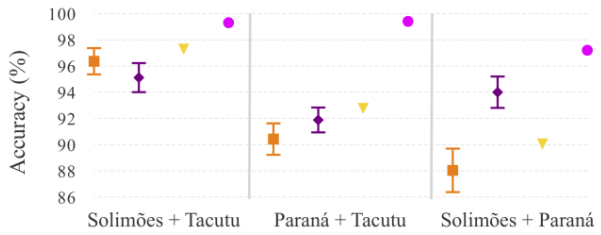


Figure 4: Validation accuracy for all experiments: ■ pure data, ♦ data augmentation, ▼ ensemble, and ● data augmentation and ensemble.

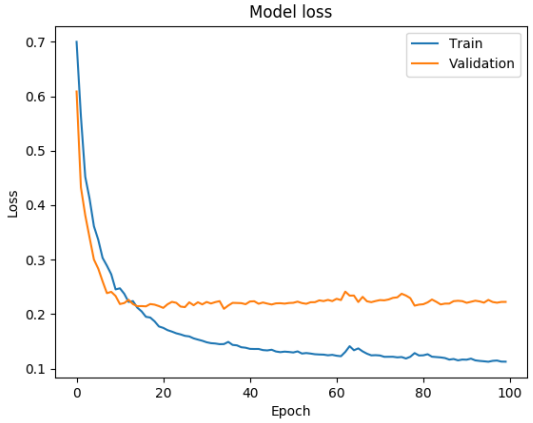
We performed four experiments using: 1) pure training set; 2) augmented training set; 3) models trained on pure training set; 4) models trained on the augmented training set. We run the experiments 1 and 2 nine times to reduce the effects of randomness. Experiments 3 and 4 were executed only one time since variability is already built into the ensemble.

Due to the increase in volume of data obtained with augmentation, we got a training set seven times larger. This contributed to a faster model converging and presenting greater stability on training and validation (see Figure 5).

Figure 4 shows the results for all experiments. We obtained an average accuracy of 91.6% for experiment 1, 93.7% for experiment 2, 93.4% for experiment 3 and 98.6% for experiment 4. The ensemble with augmentation strategy yielded the best result, with an error reduction of over 83%.

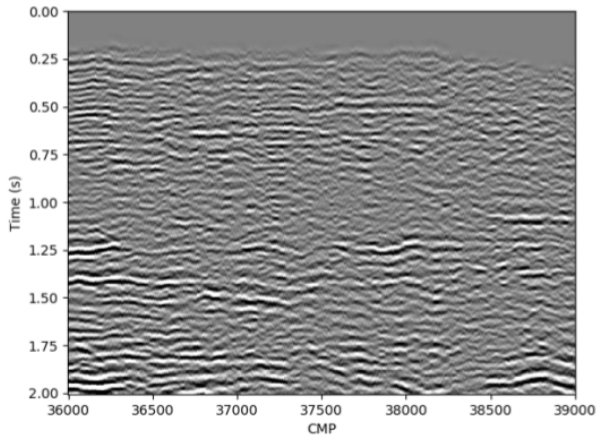
For our final analysis of this automatic apex detection tool, we evaluated the performance of the best model, from

(a)



(b)

Figure 5: Loss $\times$ Epoch for training and validation. (a) The graph for experiment 1: pure training set; (b) The graph for experiment 2: augmented training set.



(a)

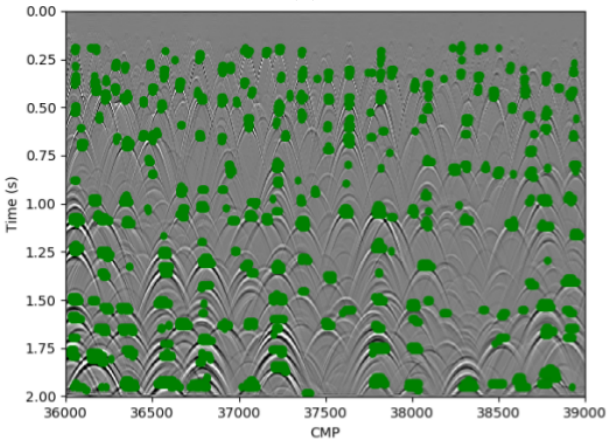


Figure 6: (a) Paraná zero offset CMP stack; (b) Inference over Paraná diffraction stack by the model from experiment 4: models trained on the augmented training set. Detected apex regions are highlighted in green.

experiment 4, over the test set. The average test set accuracy is 91.2%, varying from 87.3% (Paraná) to 94.5% (Solimões). This drop between validation and test accuracy is expected since the final model never had contact with this data during training or validation. We also did a qualitative evaluation of apex detection in the test set. We can observe Figure 6 that the model was able to consistently detect apexes that had no interference.

We used the coordinates of the detected diffraction apexes to obtain the velocity parameters at each point. According to Yilmaz (2001), these velocities can be interpolated to generate a smooth velocity model for time migration. Figure 7a shows the diffraction panel with the detected apexes, from these points we obtained the velocity parameters. By interpolating the velocities as described by Araújo et al. (2018) we obtain the panel in Figure 7b. With the velocity model as the foundation, we migrated the data in time (see Figure 7c).

Conclusions

We presented a LeNet-5 FCN to automatic detect diffraction apexes on D-sections. We trained and tested the network using only real data from land basins. To deal with the relative scarcity of annotated data, we proposed to use data augmentation and ensemble strategies. Results indicated that augmentation and ensemble contributed independently to the final improvement. Our experiments show the viability of employing an FCN architecture to the diffraction apex detection problem, yielding an attractive accuracy even from a modest amount of annotated data. A natural further improvement to the proposed algorithm is to allow for a less restrictive apex class, which would include a larger variety of apexes present in real data.

Acknowledgements

We gratefully acknowledge Petrobras, CNPq, CAPES, and FAPESP (CEPID #2013/08293-7) for partially funding this research. S. Avila is partially funded by Google Research Awards for Latin America 2018 and FAPESP (#2017/16246-0). The authors also thank the High-Performance Geophysics (HPG) team for technical support.

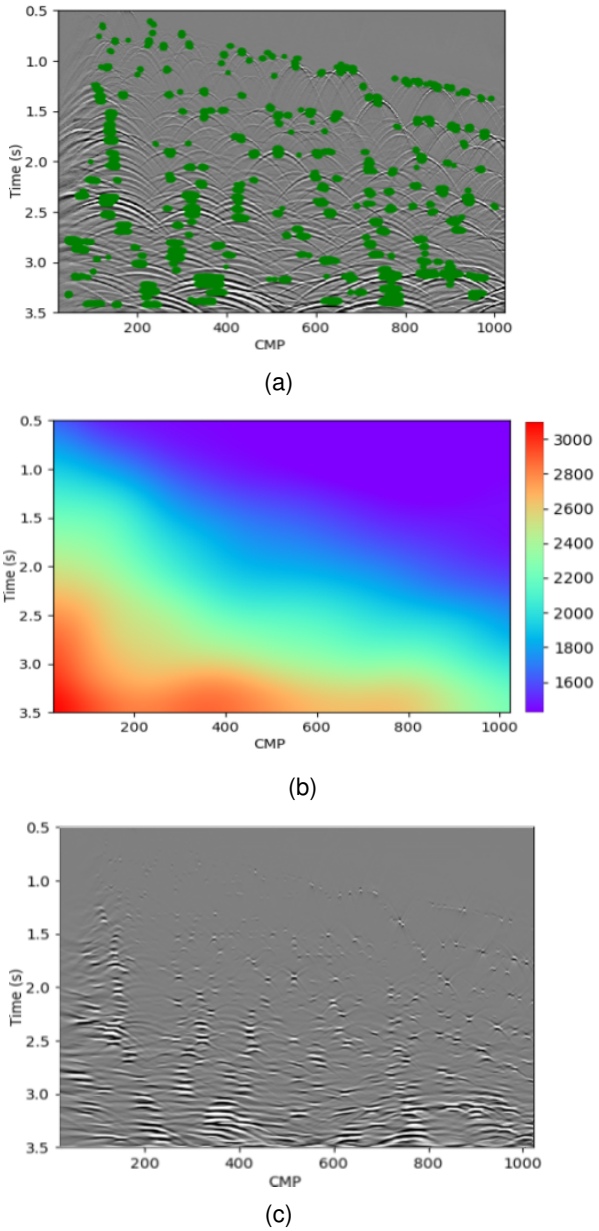


Figure 7: (a) Diffraction stack - automatic picking; (b) Interpolated velocity model; (c) Migrated diffractions.

References

Araújo, L. M., F. M. C. Oliveira, J. H. Faccipieri, T. A. Coimbra, S. Avila, M. Tygel, and E. Borin, 2018, Detecção de estruturas em dados sísmicos com Deep Learning: Boletim SBGf. Publicação da Sociedade Brasileira de Geofísica, 18–21.

Breiman, L., 1996, Bagging predictors: Machine Learning, **24**, 123–140.

Burnett, W. A., A. Klokov, S. Fomel, R. Bansal, E. Liu, and T. Jenkinson, 2015, Seismic diffraction interpretation at piceance creek: Interpretation, **3**, SF1–SF14.

Coimbra, T. A., J. J. S. de Figueiredo, J. Schleicher, A. Novais, and J. C. Costa, 2013, Migration velocity analysis using residual diffraction moveout in the poststack depth domain: GEOPHYSICS, **78**, S125–S135.

Coimbra, T. A., J. H. Faccipieri, J. H. Speglich, L.-J. Gelius, and M. Tygel, 2018, Enhancement of diffractions in prestack domain by means of a finite-offset double-square-root traveltime: GEOPHYSICS, **84**, V81–V96.

Dietterich, T. G., 2000, Ensemble methods in machine learning: International Workshop on Multiple Classifier Systems, 1–15.

Faccipieri, J. H., T. A. Coimbra, L.-J. Gelius, and M. Tygel, 2016, Stacking apertures and estimation strategies for reflection and diffraction enhancement: Geophysics, **81**, V271–V282.

Goodfellow, I., Y. Bengio, and A. Courville, 2016, Deep learning: MIT Press. (`http://www.deeplearningbook.org`).

Goodman, J. W., 2004, Introduction to fourier optics: W. H. Freeman.

Klem-Musatov, K., 1994, Theory of seismic diffractions: Society of Exploration Geophysicists.

Klem-Musatov, K., H. Hoeber, M. Pelissier, and T. J. Moser, 2016, Seismic diffraction: Society of exploration geophysicists.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner, 1998, Gradient-based learning applied to document recognition: Proceedings of the IEEE, 2278–2324.

Long, J., E. Shelhamer, and T. Darrell, 2015, Fully convolutional networks for semantic segmentation: IEEE Conference on Computer Vision and Pattern Recognition, 3431–3440.

Paulin, M., J. Revaud, Z. Harchaoui, F. Perronnin, and C. Schmid, 2014, Transformation pursuit for image classification: IEEE Conference on Computer Vision and Pattern Recognition, 3646–3653.

Reshef, M., and E. Landa, 2009, Post-stack velocity analysis in the dip-angle domain using diffractions: Geophysical Prospecting, **57**, 811–821.

Santos, L. A., W. J. Mansur, and G. A. McMechan, 2012, Tomography of diffraction-based focusing operators: GEOPHYSICS, **77**, R217–R225.

Sturzu, I., A. Popovici, M. Pelissier, J. Wolak, and T. Moser, 2014, Diffraction imaging of the eagle ford shale: First Break, **32**, 49–59.

Yilmaz, Ö., 2001, Seismic data analysis: Processing, inversion, and interpretation of seismic data: Society of exploration geophysicists.