Seismic Processing with Deep Convolutional Neural Networks: Opportunities and Challenges

S. Hou[1], H. Hoeber[1]

[1] CGG

Summary

Deep convolutional neural networks (DCNNs) are growing in popularity in seismic data processing and inversion due to their achievements in signal and image processing. In this paper we explore the link between DCNN and seismic processing. We demonstrate the potential of the application of DCNNs to seismic processing by analysing its performance with data deblending as an example. We discuss challenges and issues to solve before deploying DCNNs to production, and suggest some directions of study.

Introduction

Deep convolutional neural networks (DCNNs) have gained significant interest and popularity in seismic data processing and interpretation community due to their achievements in signal and image processing. However, recent research has also highlighted failures (they can be easily fooled, see Nguyen et al., 2015) and weaknesses (their uncertainty is not easily quantifiable, see e.g. Ghahramani, 2015) associated with such machine learning (ML) algorithms. In this paper, we demonstrate the use of DCNNs to perform seismic processing where the input data, at each step, are mapped into physics-based signal and noise components. We also discuss the challenges, algorithmic performance and accuracy as well as associated risks in using machine learning algorithms such as DCNNs for seismic processing.

Deep convolutional neural networks for seismic processing

Neural networks (NNs) non-linearly approximate an unknown function f which maps an input seismic dataset $\mathcal{X}$ to a given output (or target) $\mathcal{X}'$:

$$\mathcal{X}' = \mathrm{f}(\mathcal{X};\Phi) \qquad \Phi = \underset{\Phi}{\operatorname{argmin}}\, \mathcal{L}\,(\mathcal{X},\mathcal{X}')$$

The parameters Φ are updated iteratively by minimizing a user-defined loss function $\mathcal{L}$, such as an L1- or L2-norm, possibly with additional regularization, of the difference between $\mathcal{X}$ and $\mathcal{X}'$. NNs involve two phases: 1. Training: identifying the mapping f from known inputs and outputs (targets) via the supervised learning; 2. Inference: mapping the input to an unknown output with the pre-trained model. On the other hand, our usual way of seismic processing only involves the second phase: physics-based mapping of the input to an unknown output but with a known mapping.

Convolutional neural networks (CNNs) are considered deep if they contain at least one hidden layer. Each hidden layer is calculated from the previous layer with a linear operator (a convolutional kernel) and a non-linear operator (an activation function). The convolutional kernel consists of one or more convolutional filters. Whereas a classic multilayer perceptron consists of fully connected layers that can perform arbitrary matrix multiplications, allowing for convolutions reduces the number of parameters to be learned, acting in effect as a regularizing term. The non-linearity of the activation function enables DCNNs to describe complex non-linear relations. The activation step works similarly to functions such as muting or sparseness promotion in seismic processing and inversion.

Encoder-decoder types of DCNN architectures, such as U-Net (Ronneberger et al., 2015), contain two parts: the encoder transforms the input $\mathcal{X}$ to a feature map which is essentially a sparse representation of $\mathcal{X}$; the decoder reverse-transforms the feature map back to the target $\mathcal{X}'$. Sparsity-promoting (or domain) transforms, have been widely used in seismic processing algorithms. The domain transforms in conventional seismic processing are user chosen based on particular features of the signal and/or noise, whilst the encoder-decoder in a DCNN learns the sparse representation from the data. The DCNN may lack of direct interpretability ("black box") but has the potential to discover more underlying relations in the data.

Seismic deblending via DCNN

To illustrate these points, we apply the U-Net architectures on a seismic deblending problem. Modern seismic data involves multiple sources firing at high shooting rate, which causes the reflected wavefields to overlap (blend) in the recorded data. Conventional physics-based deblending, the process of isolating the individual source wavefields, takes into account the source firing times, sequences, and the physical signal behaviour in order to separate and obtain cleaned shot records for each individual source. We validate the application of the DCNN processing result by comparing to results of the physics-based deblending on validation data, i.e. data that was not used in the training phase.

Figure 1 shows a comparison of physics-based deblending and DCNN deblending results. The standard DCNN U-Net model with the convolutional kernel size of 3 by 3 was first trained on a subset of physics-based deblended data, and then applied to the validation data (figure 1(a) and (c)). Note that the physics-based deblending algorithm (figure 1(b)) had been parameterized to leave some mild cross-talk as a trade-off in order to preserve primaries as much as possible. In general, the learned DCNN operator generates comparable results to the physics-based results. Careful examination reveals DCNN is able to remove more high frequency noises and spikes as they are not statistically consistent; this is highlighted in Fig. 1 by the black boxes and their zooms. To further assess the primary preservation, we compare the stacks between the physics-based deblending (figure 2(b)) and the DCNN result (figure 2(c)). The amplitudes of deeper events at low frequency in standard DCNN result are not well-preserved as shown in figure 2(e). Figure 2(d) and (f) show that we are able to reduce the primary leakage, and bring the performance closer to physics based deblending on this data by applying a tailored U-Net architecture, with 3x7 asymmetric convolutional kernel size.

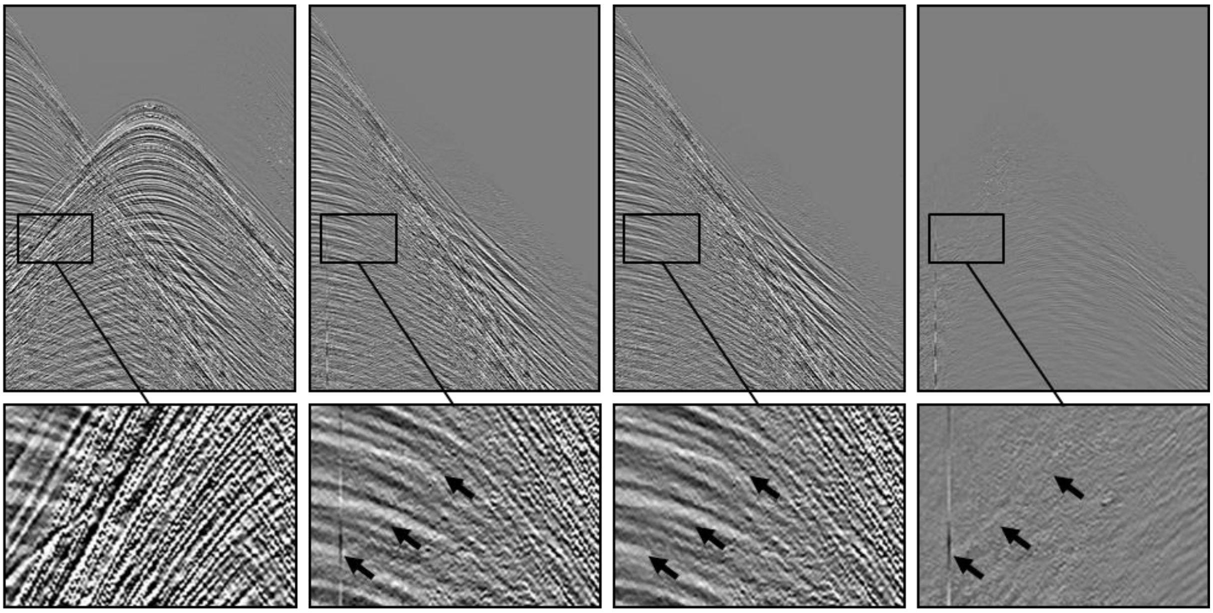

***Figure 1** shot records from field data (a) inputs to deblending; (b) physics-based deblending; (c) DCNN deblending of (a); (d) difference (b) and (c).*

Discussion: Challenges for using DCNNs for seismic processing

Although DCNNs demonstrate great potential in various applications of seismic processing, its signal fidelity at weak seismic events and the low frequency content did not yet fully match the quality of physics based (state-of-art) processing algorithms. We now discuss challenges and issues to solve before deploying DCNNs to production, and suggest some directions of study.

Theoretical challenges, metrics of success and optimal architectures. Most DNNs were originally proposed for computer vision tasks on natural images. Such images are different in a number of ways to seismic images. Both axes of natural images are spatial and usually at the same scale, whereas seismic data has axis at different scales and resolutions. Seismic bandwidth, spectrum and types of seismic noise are significantly different to that in image processing. In addition, the metrics of success for computer vision and seismic processing are different. DNNs for computer vision find the locations (for object detection) and object boundaries (for the semantic segmentation) and determine classes mainly based on shapes and colours. The results have relatively large tolerance to the pixel level details from the inputs. On the other hand, seismic processing requires extremely high signal fidelity, as the business decision could be up to billions of dollars. This means the outputs are sensitive to not only the general shapes but also the signal amplitude and phase. Weak events in seismic data (e.g. diffractions) are as important as strong events. Non-stationarity of the amplitudes (geological, divergence and attenuation) occurs particularly early on in the processing.

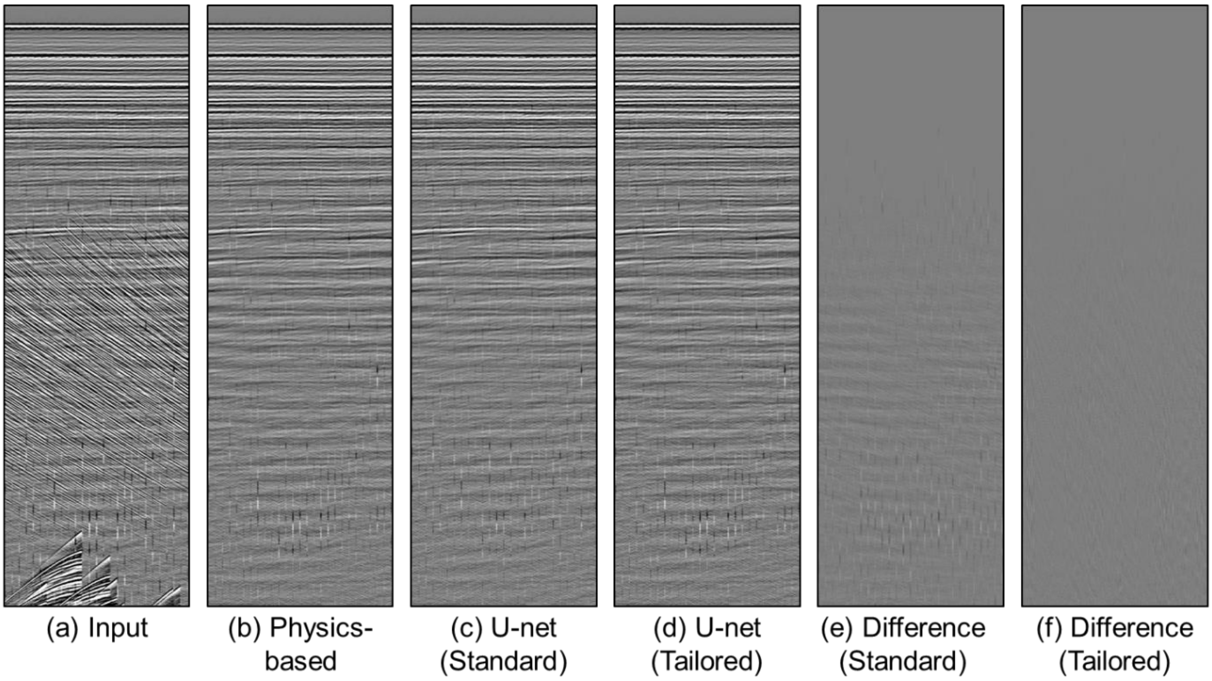

Figure 2 *Inline stacks:(a) input to deblending; (b) physics-based deblending; (c) DCNN deblending via standard U-Net; (d) DCNN deblending via tailored U-Net; (e) difference (a) and (b); (f) difference (a) and (c).*

All of this means that even the best off-the-shelf DNNs need to be further optimized for seismic data processing: 1. A small kernel size (e.g. 3x3) works well in detecting edges but is not sufficient for seismic processing where wavelet shaping occurs and where low frequencies cannot be captured with such few samples, as demonstrated in figures 2(c) and 2(e). In contrast, the tailored kernel size of 3x7 with longer filter length in time achieves higher signal fidelity at the low frequencies (figure 2(d) and 2(f)); 2. L1 and L2 loss functions are biased to strong events and as a consequence the trained models generally perform best at the dominant frequency bandwidth of the seismic. Although the validation loss only improves marginally (figure 3), the uplift on the seismic is substantial. This also shows that the loss function by itself is not sufficient as a success metric; 3. Some operators in DNNs, such as a ReLU activation function and batch normalization, can change the primary amplitudes, and need to be applied with care, whilst some model architectures, such as ResNet (He et al., 2016) and U-Net, are better amplitude preserving.

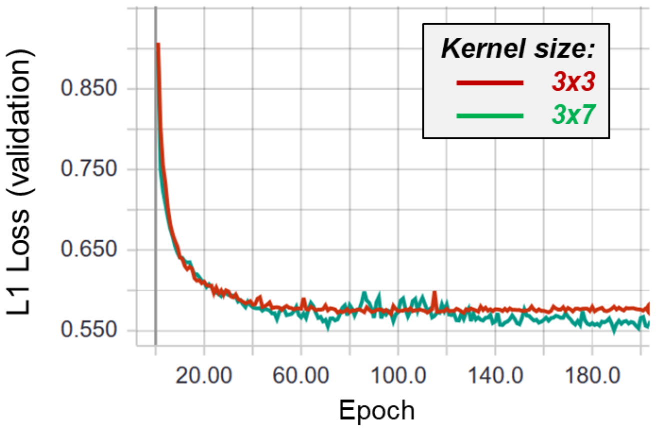

Figure 3 *L1 validation loss curves of the deblending task of figure 2 with standard U-Net (red) and tailored U-Net (green).*

Training datasets and model uncertainty: The choice of training dataset is as important as the choice of the DNN architecture. The accuracy in the training dataset affects the general accuracy of the model whilst the distribution similarity between the training dataset and the production data affects the performance of the DNN model for the production data. In other words, we have uncertainty in the noise, as well as uncertainty in the model that generated the data.

The training data $\mathcal{X}'$ can be provided via physics-based modelling or via physics-based processing of a sufficiently large subset of the data $\mathcal{X}$. Clearly, any errors in the modelling or processing are propagated to the DCNN forward operator and bias the outcome. With synthetics we can model inputs and targets, giving potentially high accuracy in the training data (our modelling may still be flawed). However, it is clearly a challenging task to generate full waveform synthetics that cover the true complexity of the distribution of the field dataset, and to do this for different processing steps. Using a subset of physics-based production data we face the same challenges. The accuracy of the data preparation defines the performance ceiling. While DCNNs can eliminate artefacts introduced by the physics-based algorithm (see Figure 1 and Hou et al., 2019), they may also adversely affect, or even miss, critical features such as subtle structures, amplitude/AVO, attenuation, and 4D effects. In this case, data augmentation is crucial in order to train a generalized model. More data and deeper models nevertheless lead to issues with compute speed and memory, which in turn impact the choice of optimization algorithm.

Lastly, we face the issue of assessing and validating the training quality prior to deploying the model to production. More specifically, we need to quantify the remaining bias and variance (the model uncertainty) without performing a large suite of additional analysis to understand if the residual is small enough and no signal leakage (e.g. correlation with geological features) is visible. Further studies on uncertainty measurement either during the training phase (e.g. with a Bayesian network) or at test time (e.g. dropout as a Bayesian approximation) will be necessary before deploying DCNNs in processing production. Furthermore, as the model uncertainty captures our ignorance about the physics behind the process, we may improve DCNNs by adding physical constraints into the model in order to further reduce the uncertainty.

Conclusions

Using the example of deblending, we have discussed various uncertainties and risks associated with the application of DCNNs to seismic processing. The emerging story in large part also mimics the journey reflected in general ML applications and literatures: After the first rush of excitement, we are now in a phase of realizing the limitations and inherent uncertainties with ML algorithms in seismic processing. This shows clearly the need for further algorithmic developments and analysis bringing into play domain expertise from the seismic industry so that ML can become part of the essential toolbox of seismic processing workflows.

Acknowledgements

The authors thank CGG for permission to publish. Special thanks to Alex Clowes, Igor Mikhalev, Ewa Kaszycka, Gordon Poole, Jeremie Messud, Sharon Howe, Chu-Ong Ting and our colleagues in CGG processing and imaging for the discussions and suggestions.

References

Ghahramani, Z. [2015] Probabilistic machine learning and artificial intelligence. *Nature*, **521**, (7553).

He, K., Zhang, X., Ren, S., and Sun, J. [2016] Deep residual learning for image recognition. *Computer Vision and Pattern Recognition (CVPR).*

Hou, S., Angio, S., Clowes, A., Mikhalev, I., Hoeber, H. and Hagedorn, S. [2019] Learn to invert: surface wave inversion with deep neural network. *81st EAGE Conference & Exhibition*, Extended Abstracts.

Nguyen A, Yosinski J, Clune J. [2015] Deep neural networks are easily fooled: high confidence predictions for unrecognizable images. *Computer Vision and Pattern Recognition (CVPR).*

Ronneberger, O., Fischer, P., and Brox, T. [2015] U-Net: Convolutional networks for biomedical image segmentation, *International Conference on Medical image computing and computer-assisted intervention*, 234-241