Released to public domain under Creative Commons license type BY (https://creativecommons.org/licenses/by/4.0).
Copyright (c) 2018 SLIM group @ Georgia Institute of Technology.

Deep-convolutional neural networks in prestack seismic: Two exploratory examples

Ali Siahkoohi[1], Mathias Louboutin[1], Rajiv Kumar[2], and Felix J. Herrmann[1,2]

1. School of Computational Science and Engineering, Georgia Institute of Technology
2. School of Earth and Atmospheric Sciences, Georgia Institute of Technology

Email: alisk@gatech.edu

Introduction

Motivated by our work on deep convolutional neural networks in seismic data reconstruction [4], we discuss how Generative Adversarial Networks (GANs) [1] can be employed in prestack problems ranging from the relatively mundane removal of the effects of the free surface to dealing with the complex effects of numerical dispersion in time-domain finite differences. Results show the potential of transfer learning in the field of seismic especially for tasks that obtaining suitable training data is hard.

Methodology

Generative Adversarial Networks

We explore the use of GANs i.e., networks that are capable of generating examples drawn from a probability distribution that may include a certain mapping. We will use the architecture based on ResNets [2] proposed by [3] and the following objective [4]:

$$
\begin{aligned}
&\min_{\theta^{(\mathcal{G})}} \ \mathop{\mathbb{E}}_{\mathbf{x}\sim p_X(\mathbf{x})}\left[\left(1-\mathcal{D}_{\theta^{(\mathcal{D})}}\left(\mathcal{G}_{\theta^{(\mathcal{G})}}(\mathbf{x})\right)\right)^2\right] + \mathop{\mathbb{E}}_{\mathbf{x}\sim p_X(\mathbf{x}),\ \mathbf{y}\sim p_Y(\mathbf{y})}\left[\|\mathcal{G}(\mathbf{x})-\mathbf{y}\|_1\right],\\
&\min_{\theta^{(\mathcal{D})}} \ \mathop{\mathbb{E}}_{\mathbf{x}\sim p_X(\mathbf{x})}\left[\left(\mathcal{D}_{\theta^{(\mathcal{D})}}\left(\mathcal{G}_{\theta^{(\mathcal{G})}}(\mathbf{x})\right)\right)^2\right] + \mathop{\mathbb{E}}_{\mathbf{y}\sim p_Y(\mathbf{y})}\left[\left(1-\mathcal{D}_{\theta^{(\mathcal{D})}}(\mathbf{y})\right)^2\right],
\end{aligned} \tag{1}
$$

$\mathcal{G}_{\theta^{(\mathcal{G})}} : X \rightarrow Y$ is a CNN mapping an initial distribution $p_X(\mathbf{x})$ to a target distribution $p_Y(\mathbf{y})$. $\mathcal{D}_{\theta^{(\mathcal{D})}} : \mathbb{R}^{m\times n} \rightarrow [0,1]$ is the discriminator CNN.

Transfer Learning

We propose exploiting transfer learning [5] by using a pre-trained CNN trained on the data obtained from a survey with similar geological features, and then fine-tuning it with a small percentage of data obtained on the new survey area.

Removal of the free surface

Our goal here is to see whether a CNN can learn the mapping from data generated with a free surface to data without a free surface. We train a CNN that maps shot-records with free surface multiple and ghost recorded at all receiver locations to the corresponding shot in training data set without free surface and ghost. The training velocity model has 25 percent different water depth.

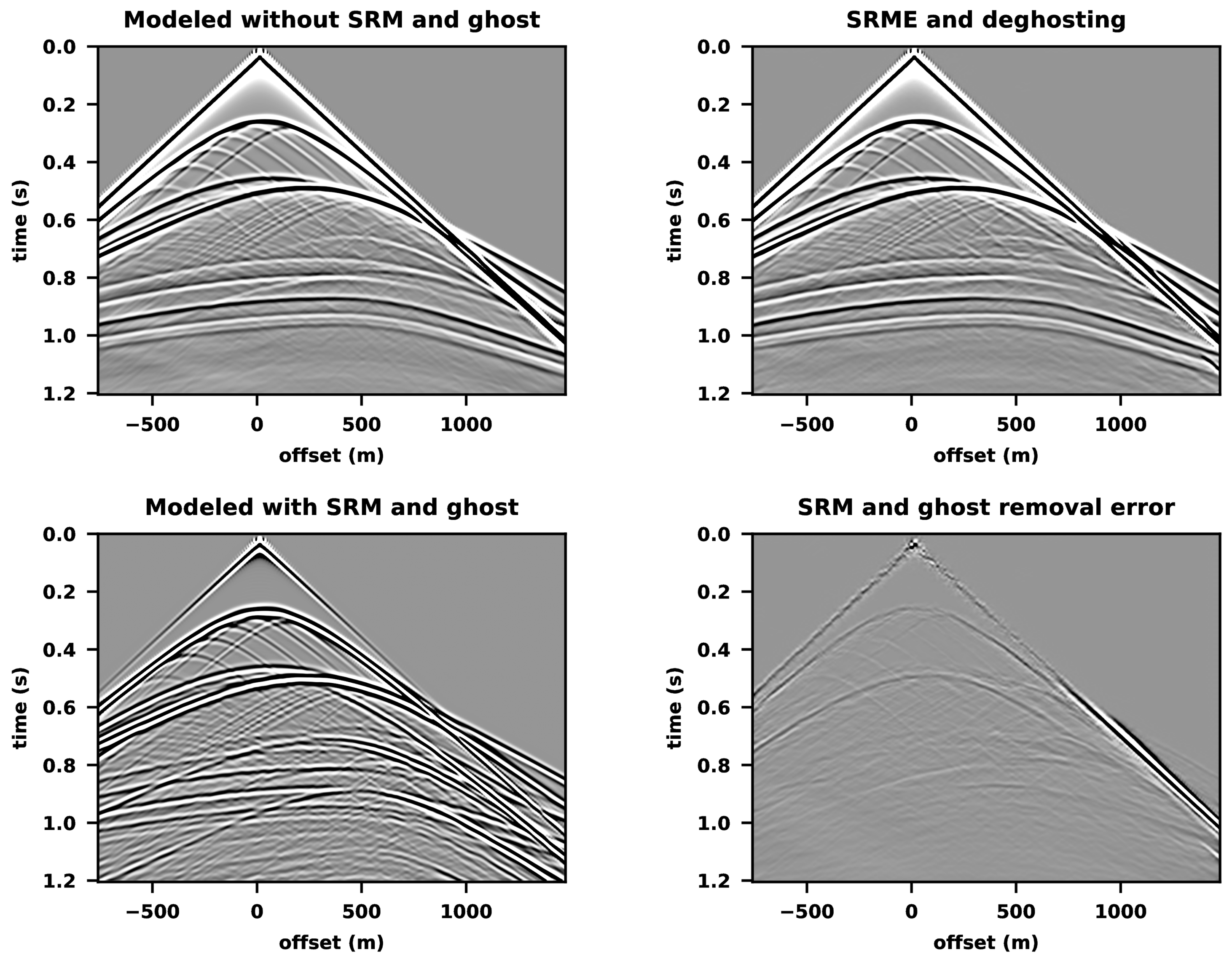

Figure 1. SRME and deghosting result via a CNN

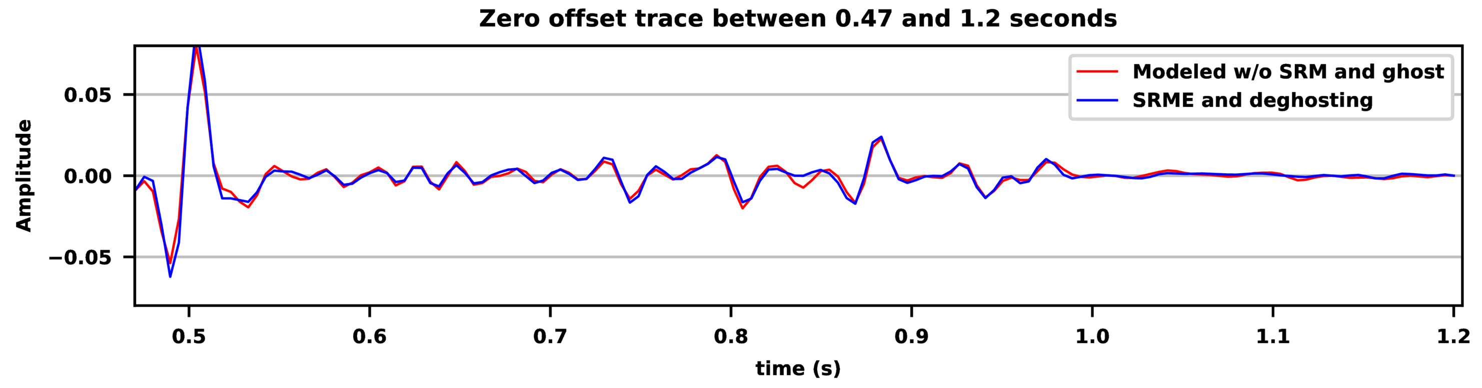

Figure 2. Comparison of Zero-offset traces

Removal of numerical dispersion

Our goal now is to train a CNN to map numerically dispersed wavefields to non-dispersed wavelfields. We do the mentioned objective by training a GAN on training velcicty pathces in Figure and later exploit transfer learning by fine-tuning the network with 5 percent of shots on testing velocity, $\mathbf{m}^*$.

We demonstrate that CNNs generalize well, as opposed to wave equation that is too specific, by comparing the correction obtained by the following equation using a slightly perturbed velocity model, $\hat{\mathbf{m}}$.

$$
\begin{aligned}
\mathbf{u} &= A^{-1}(\mathbf{m}^*)\mathbf{q},\ \underline{\mathbf{u}} = \underline{A}^{-1}(\mathbf{m}^*)\mathbf{q},\\
\hat{\mathbf{u}} &= A^{-1}(\hat{\mathbf{m}})\underline{A}(\hat{\mathbf{m}})\underline{\mathbf{u}}.
\end{aligned} \tag{2}
$$

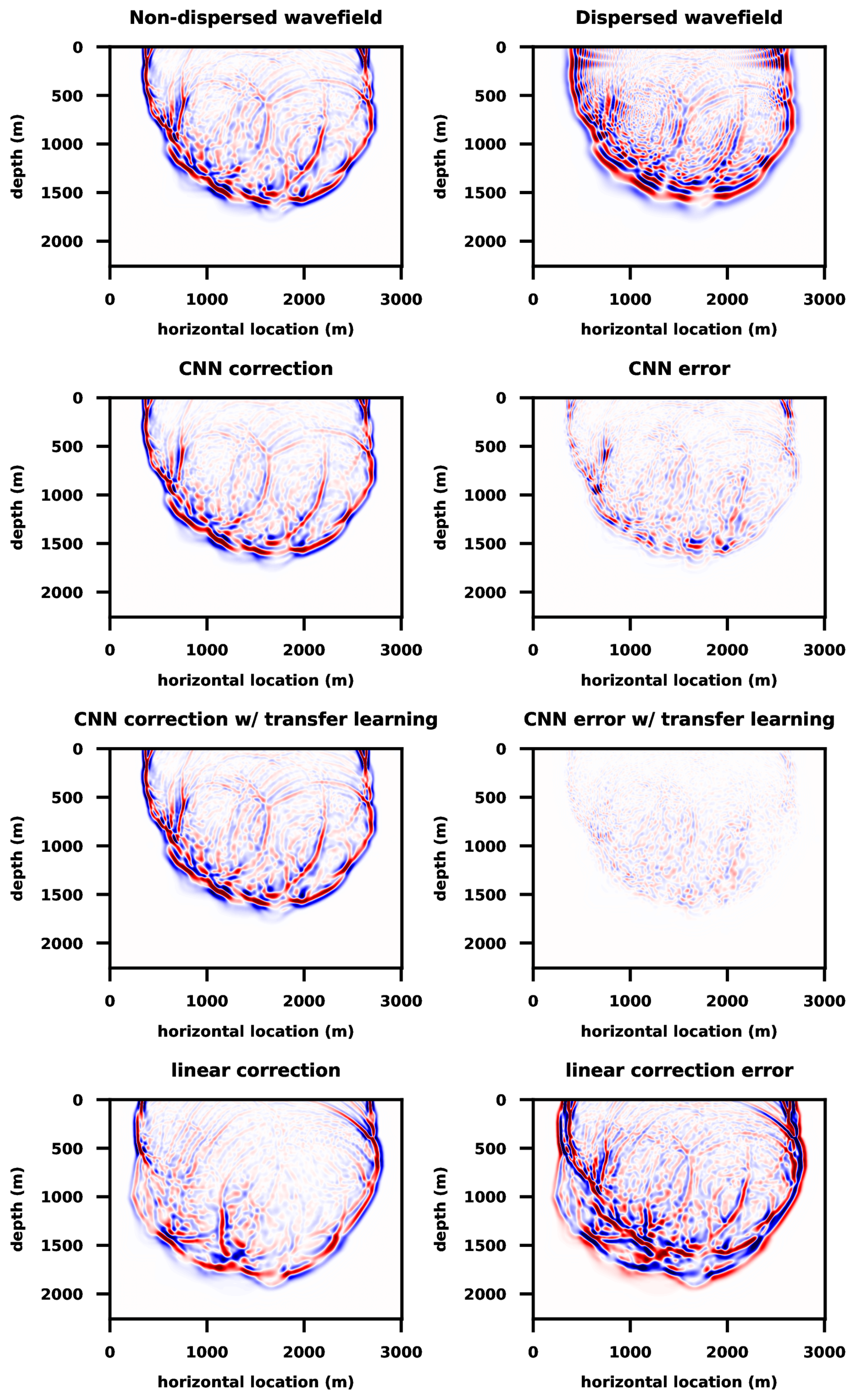

Figure 3. Numerical dispersion removal with a CNN and linear correction

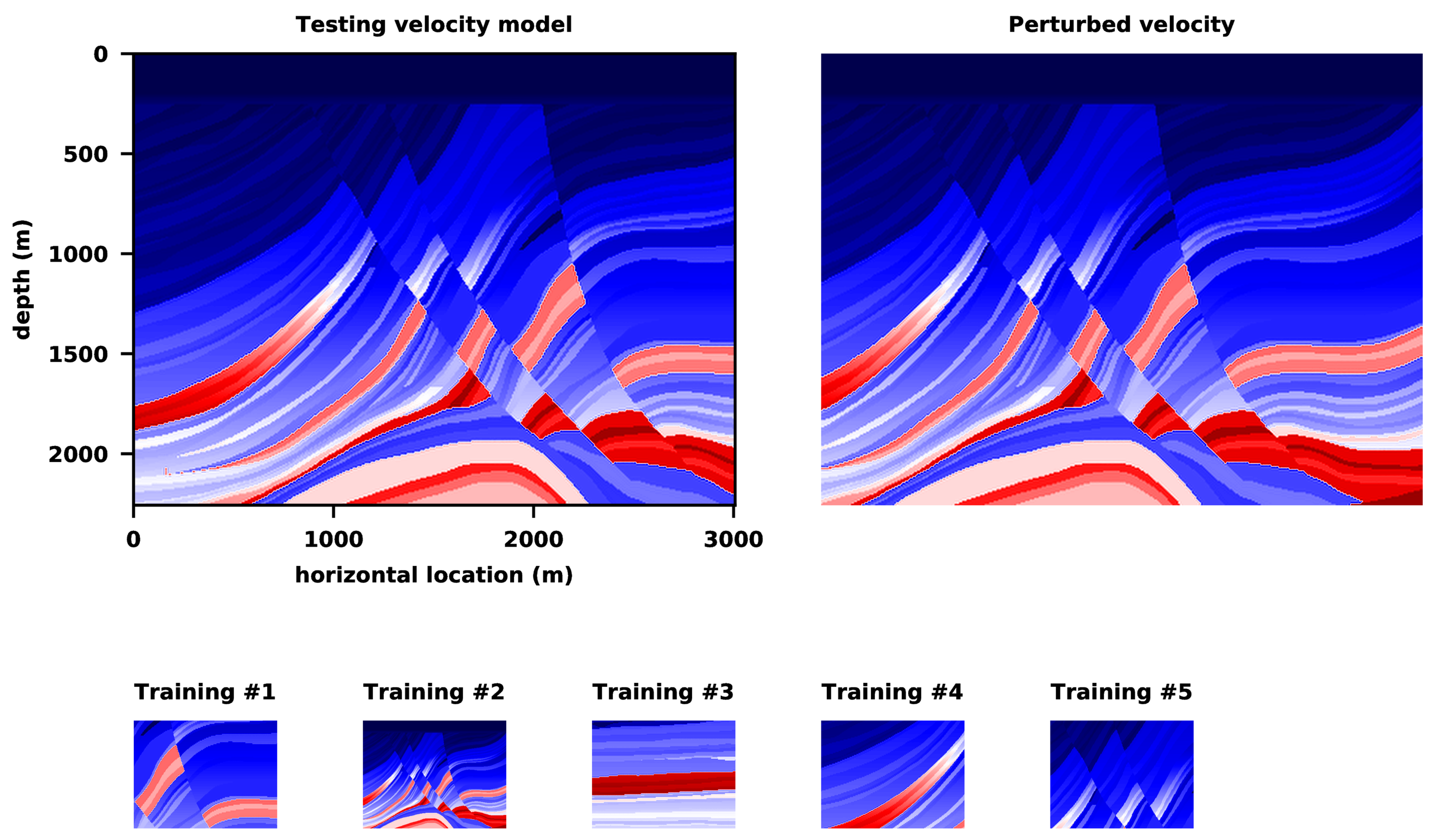

Figure 4. Training and testing velocity models for dispersion removal

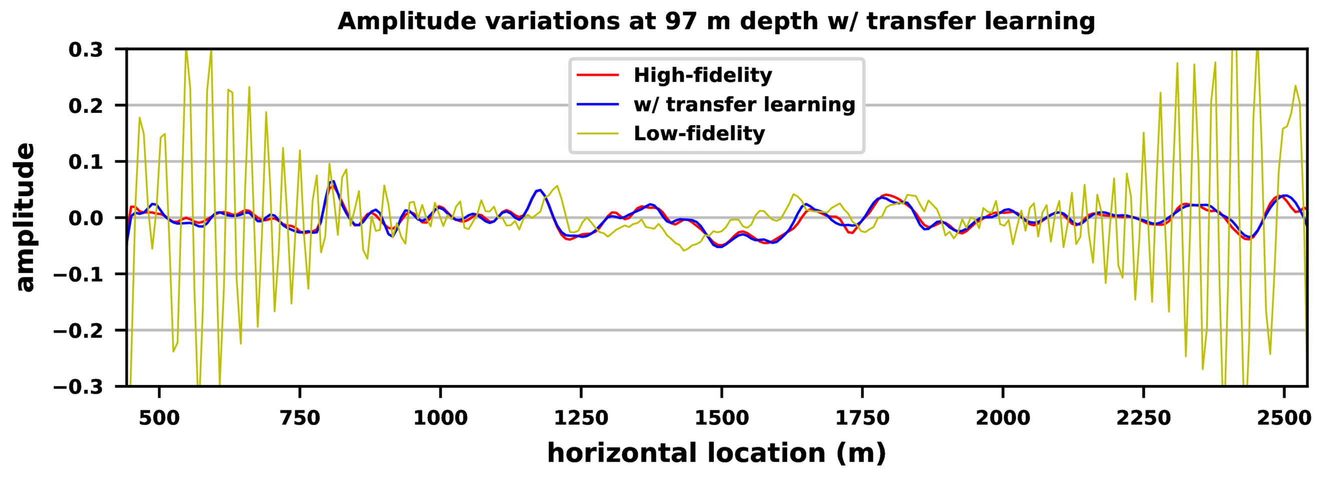

Figure 5. Numerical dispersion removal with a CNN and linear correction

Conclusion

While pedagogical, the numerical dispersion example has clearly demonstrated to us that neural networks can compensate for unmodeled intricate physics. There are strong indications that generative CNNs can have a major impact on complex tasks in prestack seismic data processing and modeling for inversion. We feel incorporating the idea of transfer learning in the field of seismic can play a key role in the success of machine learning techniques.

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. *Advances in neural information processing systems*, pages 2672–2680, 2014.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[4] Ali Siahkoohi, Rajiv Kumar, and Felix J. Herrmann. Seismic data reconstruction with generative adversarial networks. In *80th EAGE Conference and Exhibition 2018*, 2018.

[5] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.