Nonhyperbolic normal moveout stretch correction with deep learning automation*

Mohammad Mahdi Abedi[1], David Pardo[2]

[1]*Basque Center for Applied Mathematics, Bilbao, Spain. E-mail: mabedi@bcamath.org (Corresponding author).*
[2]*University of the Basque Country, Department of Mathematics, Spain; and Basque Center for Applied Mathematics, Bilbao, Spain; and Ikerbasque, Bilbao, Spain. E-mail: david.pardo@ehu.es*

Abstract
Normal moveout (NMO) correction is a fundamental step in seismic data processing. It consists of mapping seismic data from recorded traveltimes to corresponding zero-offset times. This process produces wavelet stretching as an undesired byproduct. We address the NMO stretching problem with two methods: 1) an exact stretch-free NMO correction that prevents the stretching of primary reflections, and 2) an approximate post-NMO stretch correction. Our stretch-free NMO produces parallel moveout trajectories for primary reflections. Our post-NMO stretch correction calculates the moveout of stretched wavelets as a function of offset. Both methods are based on the generalized moveout approximation and are suitable for application in complex anisotropic or heterogeneous environments. We use new moveout equations and modify the original parameter functions to be constant over the primary reflections, and then interpolate the seismogram amplitudes at the calculated traveltimes. For fast and automatic modification of the parameter functions, we use deep learning. We design a deep neural network (DNN) using convolutional layers and residual blocks. To train the DNN, we generate a set of 40,000 synthetic NMO corrected common midpoint gathers and the corresponding desired outputs of the DNN. The data set is generated using different velocity profiles, wavelets, and offset vectors, and includes multiples, ground roll, and band-limited random noise. The simplicity of the DNN task –a 1D identification of primary reflections– improves the generalization in practice. We use the trained DNN and show successful applications of our stretch-correction method on synthetic and different real data sets.

Keywords: NMO, Moveout, Neural network, VTI, Machine learning

INTRODUCTION

Normal moveout (NMO) correction consists of removing the kinematic effect of the source-receiver separation on reflections traveltime from a seismogram. It is a fundamental part of routine seismic data processing workflows and involves interpolating the amplitudes of each trace from offset-dependent traveltimes to zero-offset traveltime. To use NMO corrected data in subsequent data processing and interpretation steps, all traces in a seismogram should ideally be convered to their zero-offset version. However, the NMO correction process leads to a stretching of recorded wavelets. NMO stretching distorts the wavelets' shape and frequency content so that the highly stretched parts of the data are traditionally muted before further processing of the recorded data. The large-offset distortions are intensified in absence of a suitable nonhyperbolic moveout approximation.

* This paper is published in GEOPHYSICS (2022),87(2):U57. http://dx.doi.org/10.1190/geo2021-0489.1

NMO stretching prevention (or compensation) has been the subject of many research studies. One of the first published remedies to the problem consists of correcting the normal moveout in overlapping blocks of data (Rupert and Chun, 1975). This method and its improved versions (Brouwer, 2002; Shatilo and Aminzadeh, 2000; Strong and Hearn, 2012) avoid the stretching within each block of data but create artifacts such as discontinuities and wavelet duplication where the blocks overlap. More recent approaches use more advanced –but computationally intense– algorithms such as matching pursuit wavelet decomposition (Zhang et al., 2013; Yang et al., 2014), Wiener-shaping filter (Biondi et al., 2014), dynamic time warping (Chen et al., 2018), and regularized inversion (Ma et al., 2019; Yuan el at., 2019; Wang et al., 2021) to address the stretching issue in NMO correction. In a different method, Perroud and Tygel (2004) keep the simple NMO algorithm intact but change the input velocity function over each reflection to reproduce the zero-offset time difference of NMO trajectories at the largest offset. This method and its enhanced versions (Masoomzadeh et al., 2010; Abedi and Riahi 2016; Abedi et al., 2019; Faccipieri et al., 2019) are computationally fast but require a high operator cost because they require manual identification of primary reflections. In the above, the method of Faccipieri et al. (2019) is also applicable in common-reflection surface domain, and the methods of Abedi and Riahi (2016), Biondi et al. (2014), and Yuan et al. (2019) are suitable for NMO correction of crossing events but require special data quality such as the absence of coherent noise.

To decrease the operator cost of manual seismic data processing and interpretation steps, machine learning algorithms can be employed. Specifically, convolutional neural networks (mainly used in image-oriented deep learning) have been in use for applications that include the identification of different features in seismic data such as first arrival picking (e.g., Hollander et al., 2018; Yuan et al., 2020), fault detection, (e.g., Li et al. 2019; Wu et al., 2019), salt body detection, (e.g., Shi et al., 2019; Yang, and Ma, 2019), bright spot detection (e.g., El Zini et al., 2019), and channel detection (e.g., Pham et al., 2019; Gao et al., 2021). Therefore, deep learning is a candidate method to automatically detect the objective primary reflections for stretch-free NMO correction.

We first propose an enhanced version of Abedi and Riahi's (2016) stretch-free NMO correction using generalized moveout approximation (GMA; Fomel and Stovas, 2010). Based on that, we propose a new method for direct attenuation of NMO stretching on conventionally NMO corrected data. Next, we design a deep convolutional neural network, generate synthetic training data, and train the network for identification of primary reflections location and wavelength on conventionally NMO corrected data. We obtain a fast and automatic stretch-correction algorithm, which we test on synthetic and real data examples.

WAVELET STRETCHING CORRECTION

Normal moveout (NMO) correction consists of mapping data from the $t-x$ coordinate system into the $\tau-x$ space using a moveout equation; where t is traveltime, x is offset, and τ is traveltime at zero offset. Generalized moveout approximation (GMA; Fomel and Stovas, 2010) can produce highly accurate traveltime approximations in different models (Appendix A) and can reduce to many other traveltime approximations being in use for NMO correction. GMA equation reads (Fomel and Stovas, 2010),

$$t(x) \simeq \sqrt{\tau^2+\frac{x^2}{V_n^2}+\frac{Ax^4}{V_n^4\left(\tau^2+B\frac{x^2}{V_n^2}+\sqrt{\tau^4+2B\frac{\tau^2x^2}{V_n^2}+C\frac{x^4}{V_n^4}}\right)}}. \quad (1)$$

This equation has five parameters: τ is the zero-offset two-way time, V_n is the NMO velocity, and A, B, and C have different definitions based on specific environment properties (e.g., for transversely isotropic with vertical symmetry axis (VTI) media), or rays traced properties, as presented in appendix

A. For NMO correction, we need these parameters at all data points of the τ axis, which we call parameter functions. Figure 1 shows examples of the parameter functions of equation 1 for a layered VTI model, in red. To apply NMO correction at each x, $t(x)$ is calculated for all τ values, the amplitudes of the input common midpoint (CMP) gather are interpolated at the calculated t values, and then mapped to the corresponding τ values. This mapping process stretches the original events because the NMO trajectories (representation of equation 1 for all data points in the time axis) converge at nonzero offsets. Figure 2a shows these NMO trajectories calculated from the conventionally estimated parameter functions in Figure 1; Figure 2b shows the resultant NMO corrected CMP gather.

To avoid the NMO stretching problem, the NMO trajectories should be made parallel over the objective reflection events (Abedi and Riahi, 2016). For stretch-free NMO correction, we propose to add a shift term to equation 1 and make its parameter functions constant over the wavelength of each reflection:

$$t_{SF}(x) \simeq (\tau - \tau_{PC}) + \sqrt{\tau_{PC}^2 + \frac{x^2}{V_{nPC}^2} + \frac{A_{PC} x^4}{V_{nPC}^4 \left(\tau_{PC}^2 + B_{PC} \frac{x^2}{V_{nPC}^2} + \sqrt{\tau_{PC}^4 + 2B_{PC} \frac{\tau_{PC}^2 x^2}{V_{nPC}^2} + C_{PC} \frac{x^4}{V_{nPC}^4}} \right)}}, \quad (2)$$

where the subscript PC marks the parameter functions that are made constant over the objective reflections and interpolated in between. We call these new parameter functions "partially constant". For any offset (x) of a reflection, τ is the only variable in equation 2 (other parameters are kept constant over the reflection). Therefore, the time difference of neighboring NMO trajectories is kept fixed at all offset values, ensuring true parallelism of the trajectories. Figure 1 shows the partially constant version of the parameter functions in black. Using these parameter functions in equation 2, the resultant NMO trajectories are plotted in Figure 2c. Applying a simple NMO correction using these new trajectories, the objective reflections are mapped stretch-free (Figure 2d). Since the regions between these events are still stretched, a conventional stretch-mute based on the time difference of the subsequent trajectories can be employed. Instead of muting, we divide the magnitude of these highly stretched regions by the stretch factor to avoid discontinuities.

An already NMO corrected CMP gather can also be stretch corrected. We derive a moveout equation for stretched events and then apply a simple moveout correction by this equation on conventionally NMO-corrected gathers. This moveout is the mapping time (τ) of a data point using the conventional NMO correction (from equation 1) as a function of the mapping time of its stretch-free NMO correction (from equation 2). We solve equation 1 for τ^2, and replace t with t_{SF} from equation 2 to find the post-NMO moveout equation

$$t_{SC}^2(x) = \frac{t_{SF} V_n^2 U - (AB + U) x^2 + A \sqrt{(1 - 2A - 2B + C) x^4 + 2(B - 1) x^2 t_{SF} V_n^2 + t_{SF}^2 V_n^4}}{(A + U) V_n^2}, \quad (3)$$

where t_{SC} is the stretch-correction moveout (for a conventionally NMO corrected data using equation 1), t_{SF} comes from equation 2, and $U = A + B^2 - C$. Figure 2e shows the moveout trajectories of equation 3 over the conventionally NMO corrected data in Figure 2b. Applying a post-NMO moveout correction using equation 3 (flattening the trajectories in Figure 2e), the stretching effect is attenuated (Figure 2f). In the calculation of equation 3, we ignore the variation of parameters between τ and t_{SC}. Therefore, the post-NMO stretch corrected data by equation 3 (Figure 2e) have some degree of remaining stretch compared to the stretch-free data resulting from equation 2 (Figure 2d).

Figure 3a compares the second reflection in Figure 2 at the largest offset, after NMO correction by the aforementioned methods, with the original wavelet. Figure 3b shows the amplitude

spectra of different wavelets in Figure 3a. After the conventional nonhyperbolic NMO correction with equation 1, the wavelet stretching and its effect on frequency content are visible. The proposed stretch-free NMO correction using equation 2 produces an accurate result that matches the original wavelet without stretching. The proposed post-NMO stretch correction (equation 3) attenuates the stretching, although some remainder stretching effect is still visible in Figure 3, which is expected due to its approximate nature.

To convert the conventional parameter functions to their PC version that we use in equations 2 and 3 to address the stretching problem, we need to estimate the wavelength of the reflection. Figure 4 shows the effect of errors in the wavelength estimation on the final results. In Figure 4a, the wavelength is half the actual value. As a result, the events are partially stretch-free and partially stretched (note that there is no discontinuity). In Figure 4b the wavelength is 1.8 times the actual value. As a result, the stretch-free window that is formed over the objective reflection interfere with the neighboring reflection. Although the NMO-corrected wavelets in Figure 4b match the corresponding ones in Figure 3a, some artifacts are produced in their vicinity.

The proposed approaches for stretch-free NMO and post-NMO stretch correction are fast and entail the same computational cost as that of a simple NMO correction. They are reversible, and do not require specific conditions in the input data. However, to convert the parameter functions to their partially constant counterparts, we should identify the data points that are located over the wavelength of the objective reflections at zero-offset. Since it is tedious to manually select the onset and end of each reflected wavelet, we leave this task to a machine learning algorithm.

AUTOMATION WITH DEEP LEARNING

To automatize the process of stretch correction, we should automatically obtain the partially constant parameters from the regular parameters. We limit the automation process to the post-NMO stretch correction because the identification of primary reflections in an NMO corrected gather is simpler. We design a deep neural network (DNN) that given a conventionally NMO corrected CMP gather, separates the zero-offset time points located over the wavelength of the objective reflections from the rest. Our desired output is made of ones over the reflection's wavelength at zero offset and zeros elsewhere. For each identified event, we make the regular parameters constant by repeating the parameter values at the central sample and interpolate between the events. The simplicity of the DNN task –a 1D identification of primary reflections– improves the generalization in practice. In the following, we describe the training data, our DNN architecture, and the training process.

Training dataset

We generate a training dataset composed of 40000 input NMO corrected CMPs as images with a fixed size of 501×40 pixels (with values ranging between -1 and 1), and the desired output (ground truth) for each data sample. The size of each CMP is selected so that it is sufficiently large to show the main primary reflections and other events in an NMO corrected CMP and small enough for computational efficiency. With a time sampling interval of 4ms, the input is a 2s long CMP, which includes the part of conventional seismic data where the main stretching problem arises. This dataset is also large enough to include different features of seismic data that we explain in the following. Nonetheless, larger data sets may be needed if more complicated data are considered.

We generate the training data by a fast ray-based algorithm in horizontally layered earth models. First, we generate 20,000 CMP samples using different velocity-depth profiles, wavelets, and offset vectors:

1. The velocities and thicknesses of layers are selected within certain predefined ranges. The velocity ranges from 1.4 to 5 km/s, and the thickness ranges from 110 to 420 m. Velocities are selected so that the velocity-depth profile has a realistic trend (Figure 5a). The number of

layers varies because we model for reflections that have a fixed maximum zero-offset two-way time of 2s.
2. We generate a set of 1,200 different wavelets, including Ricker wavelets with different dominant frequencies between 16 and 32 Hz, different peak-to-trough magnitude ratios, and different phase rotations from -45° to 45° (Figure 5b).
3. The source-to-receiver offsets are uniformly distributed with a maximum of between 3 and 5 km.

The modeled data include amplitude and phase variations with offset. In half of the modeled CMPs, we include free-surface interbed and surface layer intrabed multiple reflections from different layer boundaries.

We expand the original 20,000 samples to 40,000 samples by separately adding two types of noise to each input data sample (each modeled CMP): 1) Bandlimited normally distributed random noise with different signal-to-noise ratios and frequency content. 2) Linear dispersive noise (ground roll).

Finally, we NMO correct each CMP with GMA and normalize each CMP with its maximum absolute value. We use the ray-traced parameterization (Appendix A), which produces a highly accurate NMO correction. We induce few residual moveouts in the NMO correction, which may be seen in realistic seismic data processing, by adding errors to parameters A and B of GMA (equation 1) in a normal distribution with a standard deviation of 10%. We apply a 200% stretch-mute function to remove the highly distorted parts of the training data, but keep the first ten traces unmuted. Figure 6 shows four representative examples of the training data set.

Similarly, we generate another 4,000 data samples (input CMPs and corresponding output) for the validation dataset. This data is generated with different random number generator seeds than the ones used for generating the training data set.

Architecture

We use a convolutional DNN with residual blocks (Figure 7). A residual block (He et al., 2016) has a shortcut connection over few layers and it is designed to mitigate the problems of vanishing gradients and accuracy saturation in deeper networks. After each convolutional layer and before the activation function, we use batch normalization for stabilization (Ioffe and Szegedy, 2015). The network should shrink the input image size (501×40 pixels, determined when generating the training data set) to the output vector size (501×1). Therefore, we use strides in some convolutional layers and pooling layers between the blocks. In each residual block, the shortcut path goes through a convolutional block that has the same size reduction as in the main path. We use the ReLU activation function within each block and the sigmoid activation function at their output to add nonlinearity to the network and to ensure that the desired output (ground truth) of the network is between 0 and 1. The final architecture (Figure 7) was selected after testing several different variations of it. The network has 29,510 trainable parameters, which are determined during training.

Training

We deploy the designed DNN (Figure 7) in TensorFlow (Abad et al., 2016). We select the conventional mean squared difference between the DNN output and the ground truth as the loss function. We also tested lower and higher norms of the misfit, and we observed the best performance with the L2-norm. Adam's gradient-based algorithm (Kingma and Ba, 2014) is selected as the optimizer, which trains the DNN in (initially selected) 300 epochs. Adam is computationally efficient and adaptively changes the learning rate during training. In each epoch, all the training data samples pass through the network, and the optimizer updates the layers' weights to decrease the loss value. Figure 8a shows the loss evolution during the training process for both the training and validation datasets. Since our final objective is to assign a zero or a one to each point in the output vector, we calculate binary accuracy as a metric for the assessment of the training process (Figure 8b). This

metric calculates the percentage of the points in the output that correctly match the ground truth, after assigning 1 to all values above 0.5, and 0 to all values below 0.5. A similar binary classification step is added to the DNN output after training. In Figure 8 both the loss and metric values of the training and validation data start to depart, and the validation result stops to improve after epoch 75. Therefore, we select the network after 75 epochs of training as our final trained DNN. For the training data examples shown in Figure 6, we also display the trained DNN outputs (as the red dots), which properly match the ground truth (black dots).

APPLICATIONS

To test the proposed method for NMO stretch correction, we generate a synthetic data set using acoustic finite-difference modeling in an earth model that contains dipping and curved reflectors, which can induce non-hyperbolic moveout (see appendices of Fomel and Stovas, 2010). The modeled data contain multiples. We also add dispersive linear noise to each modeled CMP. Therefore, the data set is obtained using a different modeling algorithm in a different earth model than the ones used for generating the training and validation data sets in the previous section. The testing data set contains 163 CMP gathers. We use a conventional semblance-based parameter estimation on selected CMPs, interpolate the estimated parameters, and apply NMO correction on each CMP gather using a three-parameter version of GMA (Fomel and Stovas, 2010). We used a manual picking of V_n and η, but there are many studies on the automation of this process that can be employed in practice (e.g. using particle swarm optimization (Fortini et al. 2013; Wilson and Gross, 2019), polynomial chaos expansion (Abassi and Gholami, 2018), and simulated annealing (Velis, 2021)). Figure 9a shows one example of the processed CMPs. Introducing each conventionally processed CMP gather to the previously trained DNN, we rapidly calculate the objective output vector (Figure 9a) needed to make the GMA parameters partially constant over the primary reflections. To preserve the continuity, we use the average of three neighboring CMPs as the input. Although some primary reflections have residual moveout (i.e. they are not completely flat), the DNN has recognized them properly in presence of multiples. Using the resultant partially constant parameter functions in equation 3, Figure 9b shows the stretch-corrected version of Figure 9a. Repeating this process for all the CMP gathers in the data set and stacking the results, Figure 9c and 9d show the resultant stacked sections. The stacked section resulting from the conventional NMO correction (Figure 9c) shows severe stretching effects that include changes of wavelet shapes and weakened amplitudes. After applying our post-NMO stretch correction method, the stacked section shows improved clarity and accuracy (Figure 9d). Figure 10 shows the local frequency content (Fomel, 2007) of the stacked sections depicted in Figure 9. The attenuation of higher frequencies in the stacked section is a known problem of NMO stretching, and it is overcome by our proposed stretching-correction method.

Next, we test our method on a real marine CMP gather containing multiples and dispersive noise. Figure 11a shows the CMP gather, NMO corrected by GMA (equation 1) with the parameterization from Abedi and Stovas (2019a) presented in Appendix A. The number of traces in this data is larger than the input size of our trained DNN. Therefore, we use the first 40 of every other trace in the real data as the input of our DNN. The CMP is covered with multiples and vigorous linear noise, and the DNN has recognized the visible flat primary reflections. We use the DNN output (shown in Figure 11) to obtain the partially constant parameter function, then apply our stretch-correction method. Figure 11b shows the result. In Figure 11c and 11d, we calculate the local frequency content (Fomel, 2007) of the CMP gathers. After applying our stretch-correction method, the selected primary reflections by DNN have a higher frequency content.

Finally, we test our method on a set of 106 deep-marine CMP gathers. Figure 12a shows a representative CMP gather NMO corrected with equation 1 (with parameter definition in equation A-1), before and after stretch correction by our method. The conventional NMO-corrected data have

limited stretching and no interfering artifact, therefore, a mute function is not used. We separate the large-angle part of each CMP (indicated with red lines) and present the large-angle stacked data before and after stretch correction in Figure 12b and 12c. The mean amplitude spectra shown in Figure 12d shows that the spectral content of the far angles is improved after our stretch correction.

CONCLUSION

We proposed two methods to address the NMO stretching problem: 1) a stretch-free NMO correction, and 2) a post-NMO stretch correction. The first method prevents the stretching of primary reflections during NMO correction. The second method reduces the stretching using an already NMO-corrected gather. Both methods are based on the GMA that reportedly produces highly accurate traveltime approximations in different environments. Both methods employ a simple NMO correction algorithm with new moveout equations (equations 2 and 3) and new parameter functions. They consist of direct mapping of interpolated amplitudes with no sensitivity to data quality. They correct for the stretching of primary reflections but stretch other events in between the corrected ones to avoid discontinuity.

Critical to the development of the above methods is the use of deep learning to convert the original parameter functions of GMA to the new partially constant parameter functions. We designed a DNN using convolutional layers and residual blocks, generated a training data set of NMO corrected synthetic CMP gathers, and then trained the network to identify the time samples of primary reflections at zero offset. We found 40,000 carefully selected training data samples to be general enough for our application and reached the optimal performance of the designed DNN after 75 epochs.

The presented applications of our method on a set of synthetic modeled data and two real marine data sets show improvements in preserving the wavelets' shape and frequency content after stretch correction. The NMO correction process performed by equation 2 or 3 is as fast as a conventional NMO correction. DNNs are also known to be fast in application. Our DNN works with fixed input size. However, we can have stretch correction of larger CMPs using a downsampled version of the CMP as the DNN input.

ACKNOWLEDGMENTS

We are grateful to the journal reviewers, H. Wilson and anonymous reviewers, and to the associate editor I. Jones, and the editor in chief J. Etgen for their insightful comments and suggestions. The work is supported by the European Union's Horizon 2020 research and innovation program grant agreement no. 777778 (MATHROCKS); the European Regional Development Fund (ERDF) through the Interreg V-A Spain-France-Andorra program POCTEFA 2014-2020 Project PIXIL (EFA362/19); the Spanish Ministry of Science and Innovation with references PID2019-108111RB-I00 (FEDER/AEI), PDC2021-121093-I00, and the "BCAM Severo Ochoa" accreditation of excellence (SEV-2017-0718); and the Basque government through the BERC 2018-2021 program, the three Elkartek projects 3KIA (KK-2020/00049), EXPERTIA (KK-2021/00048), and SIGZE (KK-2021/00095), and the Consolidated Research Group MATHMODE (IT1294-19) given by the Department of Education.

APPENDIX A
Different parameterizations for generalized moveout approximation

Parameters A, B, and C in equation 1 have flexible definitions based on the type of constraints that are used to fit the traveltime approximation to exact kinematical properties. For VTI media, Fomel and Stovas (2010) find,

$$A = -4\eta,\ B = \frac{1+8\eta+8\eta^2}{(1+2\eta)},\ C = \frac{1}{(1+2\eta)^2}, \tag{A-1}$$

where η is called anellipticity parameter. For VTI media, Abedi and Stovas (2019a) modify the definition of the parameters to:

$$A = -4\eta\frac{\left(\eta+\sqrt{1+2\eta}\right)^2}{(1+2\eta)^2}, B = \frac{1+2\eta\left(2+\eta+2\sqrt{1+2\eta}\right)}{1+2\eta}, C = \frac{1}{(1+2\eta)^2}. \tag{A-2}$$

Using equation A-2 instead of equation A-1, the accuracy of GMA (equation 1) improves in VTI media. In layered earth models, GMA and its extended version (Abedi and Stovas 2019b) produce nearly exact traveltime approximations when defining the auxiliary parameters from the properties of one ray that is traced at a finite large offset (Fomel and Stovas, 2010):

$$\begin{aligned} B &= \frac{\left(\tilde{x}_f - \tilde{p}_f\tilde{t}_f\right)}{\tilde{x}_f\left(1+\tilde{t}_f\left(\tilde{p}_f\tilde{x}_f - \tilde{t}_f\right)\right)} - \frac{\tilde{x}_f{}^2 A}{1-\tilde{t}_f{}^2+\tilde{x}_f{}^2}, \\ C &= \frac{\left(\tilde{x}_f - \tilde{p}_f\tilde{t}_f\right)^2}{\tilde{x}_f{}^2\left(1+\tilde{t}_f\left(\tilde{p}_f\tilde{x}_f - \tilde{t}_f\right)\right)^2} + \frac{2A}{1-\tilde{t}_f{}^2+\tilde{x}_f{}^2}, \end{aligned} \tag{A-3}$$

where $\tilde{x} = x/(tV_n)$, $\tilde{t} = t/\tau$, $\tilde{p} = pV_n$, p is ray-parameter and subscript f denotes the far-offset reference ray.

References

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, 2016, Tensorflow: A system for large-scale machine learning. In 12th symposium on operating systems design and implementation, 265-283.

Abbasi, M., and A. Gholami, 2018, Automatic nonhyperbolic velocity analysis by polynomial chaos expansion: Geophysics, **86**, no. 6, U79–U88.

Abedi, M. M., and M. A. Riahi, 2016, Nonhyperbolic stretch-free normal moveout correction: Geophysics, **81**, no. 6, U87-U95.

Abedi, M.M., M.A. Riahi, and A. Stovas, 2019, Three-parameter normal moveout correction in layered anisotropic media: A stretch-free approach: Geophysics, **84**, no. 3, C129-C142.

Abedi, M.M., and A. Stovas, 2019a, A new parameterization for generalized moveout approximation, based on three rays: Geophysical Prospecting, **67**, no. 5, 1243-1255.

Abedi, M. M., and A. Stovas, 2019b, Extended generalized non-hyperbolic moveout approximation: Geophysical Journal International, **216**, no. 2, 1428-1440.

Biondi, E., E. Stucchi, and A. Mazzotti, 2014, Nonstretch normal moveout through iterative partial correction and deconvolution: Geophysics, **79**, no. 4, V131-V141.

Brouwer, J. H., 2002, Improved NMO correction with a specific application to shallow-seismic data: Geophysical Prospecting, **50**, 225-237.

Chen, S., S. Jin, X. Y. Li, and W. Yang, 2018, Nonstretching normal-moveout correction using a dynamic time warping algorithm: Geophysics, **83**, no. 1, V27-V37.

El Zini, J., Y. Rizk, and M. Awad, 2019, A deep transfer learning framework for seismic data analysis: A case study on bright spot detection: IEEE Transactions on Geoscience and Remote Sensing, **58**, no. 5, 3202-3212.

Faccipieri, J.H., T.A. Coimbra, and R. Bloot, 2019, Stretch-free generalized normal moveout correction: Geophysical Prospecting, **67**, no. 1, 52-68.

Fomel, S., 2007, Local seismic attributes: Geophysics, **72**, no. 9, A29–A33.

Fomel, S., and A. Stovas, 2010, Generalized nonhyperbolic moveout approximation: Geophysics, **75**, no. 2, U9-U18.

Fortini, C., D. Maggi, V. Lipari, and M. Ferla, 2013, Particle swarm optimization for seismic velocity analysis: SEG Technical Program Expanded Abstracts, 4864-4868.

Gao, H., X. Wu, and G. Liu, 2021, ChannelSeg3D: channel simulation and deep learning for channel interpretation in 3D seismic images: Geophysics, **86**, no. 4, IM73-IM83.

He, K., X. Zhang, Sh. Ren, and J. Sun, 2016, Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, 770-778.

Hollander, Y., A. Merouane, and O. Yilmaz, 2018, Using a deep convolutional neural network to enhance the accuracy of first-break picking: 88th Annual International Meeting, SEG, 4628-4632.

Ioffe S., and C. Szegedy, 2015, Batch normalization: Accelerating deep network training by reducing internal covariate shift: Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:448-456

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

Li, S., C. Yang, H. Sun, and H. Zhang, 2019, Seismic fault detection using an encoder–decoder convolutional neural network with a small training set: Journal of Geophysics and Engineering, **16**, no. 1, 175-189.

Ma, Y., Y. Luo, and P. Kelamis, 2019, Superresolution stacking based on sparse Radon transform: Geophysics, **84**, no. 1, V45-V54.

Masoomzadeh, H., P.J. Barton, and S.C. Singh, 2010, Nonstretch moveout correction of long-offset multichannel seismic data for subbasalt imaging, Example from the North Atlantic: Geophysics, **75**, no. 4, R83-R91.

Perroud, H., and M. Tygel, 2004, Nonstretch NMO: Geophysics, **69**, 599-607.

Pham, N., S. Fomel, and D. Dunlap, 2019, Automatic channel detection using deep learning: Interpretation, **7**, no. 3, SE43-SE50.

Rupert, G.B., and J.H. Chun, 1975, The block move sum normal moveout correction: Geophysics, **40**, 17-24.

Shatilo, A., and F. Aminzadeh, 2000, Constant normal-moveout (CNMO) correction: a technique and test results: Geophysical Prospecting, **48**, 473-488.

Shi, Y., X. Wu, and S. Fomel, 2019, SaltSeg: Automatic 3D salt segmentation using a deep convolutional neural network: Interpretation, **7**, no. 3, SE113-SE122.

Strong, S., and S. Hearn, 2012, The importance of non-hyperbolic and stretch effects in far-offset P and PS NMO processing: ASEG Extended Abstracts, 1-4.

Velis, D., 2021, Simulated annealing velocity analysis: Automating the picking process: Geophysics, **86**, no. 2, V119-V130.

Wang, D., S. Yuan, T. Liu, S. Li, and S. Wang, 2021, Inversion-based non-stationary normal moveout correction along with prestack high-resolution processing: Journal of Applied Geophysics, **191**, 104379.

Wilson, H. and L. Gross, 2019, Reflection-constrained 2D and 3D non-hyperbolic moveout analysis using particle swarm optimization: Geophysical Prospecting, **67**, no.3, 550-571.

Wu, X., L. Liang, Y. Shi, and S. Fomel, 2019, FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation. Geophysics, **84**, no. 3, IM35-IM45.

Yang, F. and J. Ma, 2019, Deep-learning inversion: A next-generation seismic velocity model building method: Geophysics, **84**, no. 4, R583-R599.

Yuan, P., S. Wang, W. Hu, X. Wu, J. Chen, and H. Van Nguyen, 2020, A robust first-arrival picking workflow using convolutional and recurrent neural networks: Geophysics, **85**, no. 5, U109-U119.

Yang, X., S. Cao, Q. Liu, Y. Wang, and D. Yuan, 2014, A Modified Nonstretching NMO Correction Using Matching-Pursuit Algorithm: SEG, 84th Annual International Meeting, expanded Abstracts, 517-521.

Yuan, S., W. Wei, D. Wang, P. Shi, and S. Wang, 2019, Goal-oriented inversion-based NMO correction using a convex $L_{2,1}$-Norm: IEEE Geoscience and Remote Sensing Letters, **17**, no. 1, 162-166.

Zhang, B., K. Zhang, Sh. Guo, and K. J. Marfurt, 2013, Nonstretching NMO correction of prestack time-migrated gathers using a matching-pursuit algorithm: Geophysics, **78**, no. 1, U9-U18.

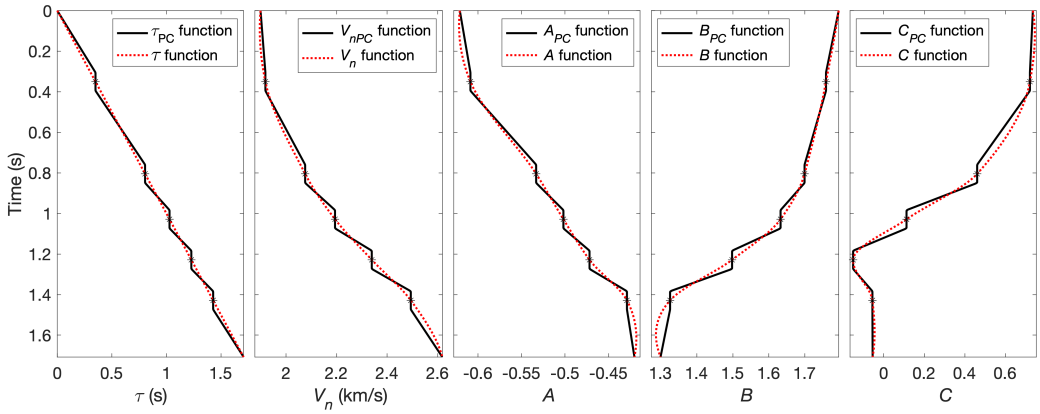

Figure 1. Parameters of equation 1 as a function of zero-offset two-way time. The red dotted lines show the conventional parameter functions, and the black lines show the partially constant version that we use in equation 2 or 3 to address the stretching problem. Note that the τ function is identity. The zero-offset times of the reflections are marked with a star.

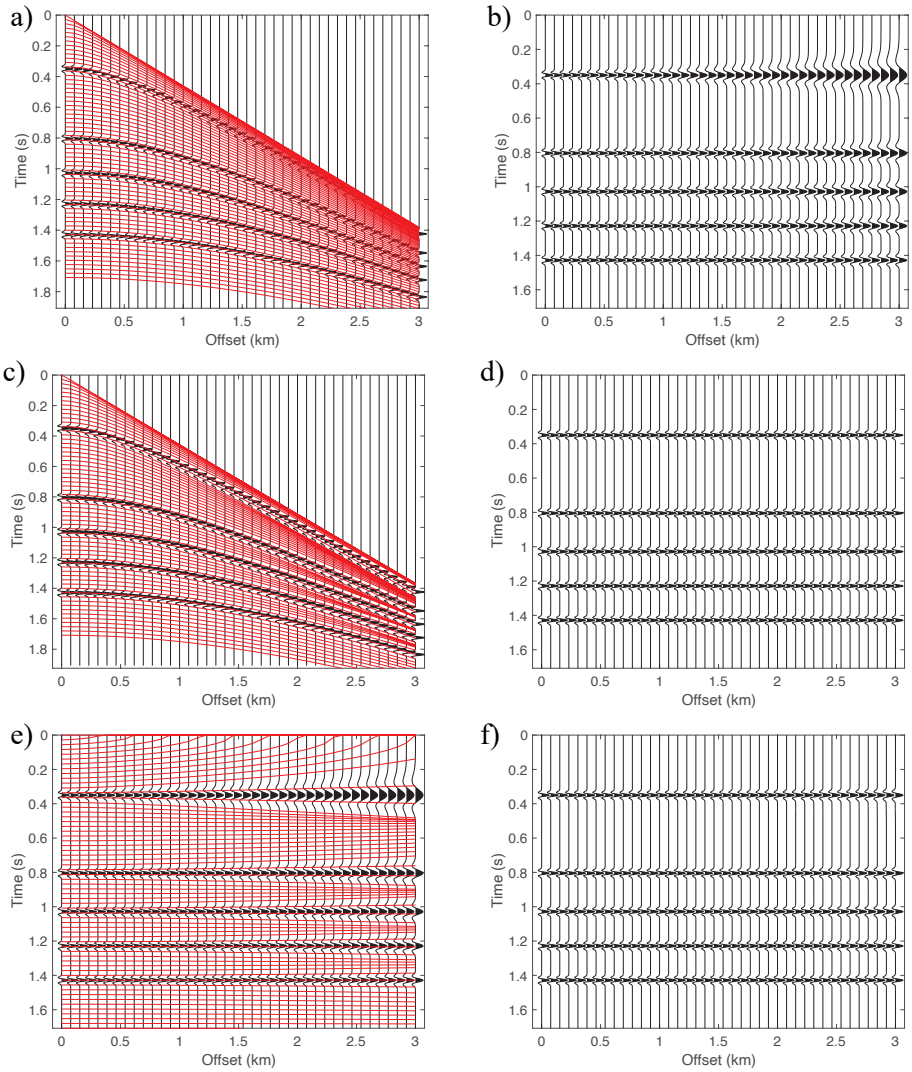

Figure 2. a) A synthetic CMP gather (in wiggle plot), and the NMO trajectories (in red) resulting from inserting the conventional parameter functions (Figure 1) into equation 1. b) NMO correction result of the panel (a). c) The same CMP gather overlaid with the stretch-free NMO trajectories resulting from inserting the PC parameter functions (Figure 1) into equation 2. d) NMO correction result of the panel (c). e) Conventionally NMO corrected CMP (panel b), overlaid with the stretch-correction trajectories resulting from equation 3. f) Moveout correction result of the panel (e).

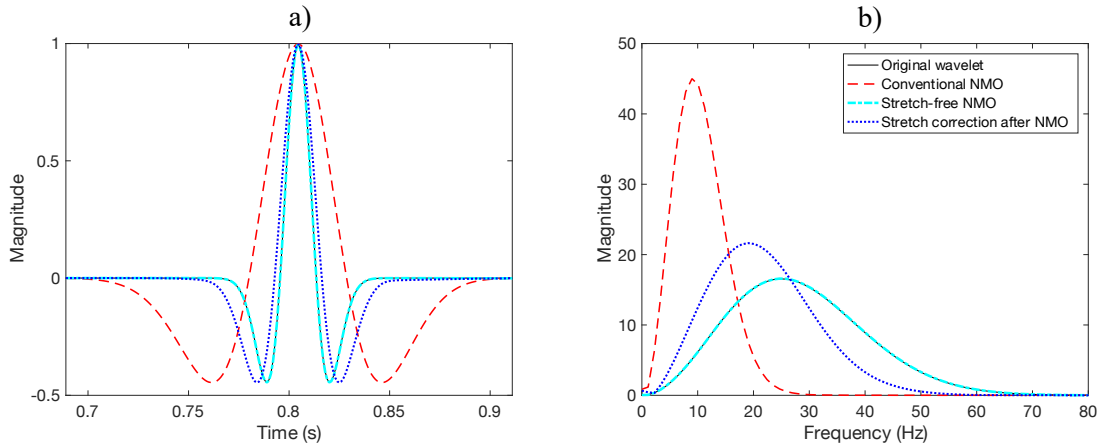

Figure 3. NMO correction effect on wavelet shape (a), and frequency content (b), presented for the second reflection in Figure 2 at its largest offset. The proposed stretch-free NMO correction by equation 2 is exact, but the proposed post-NMO stretch correction by equation 3 is an approximation.

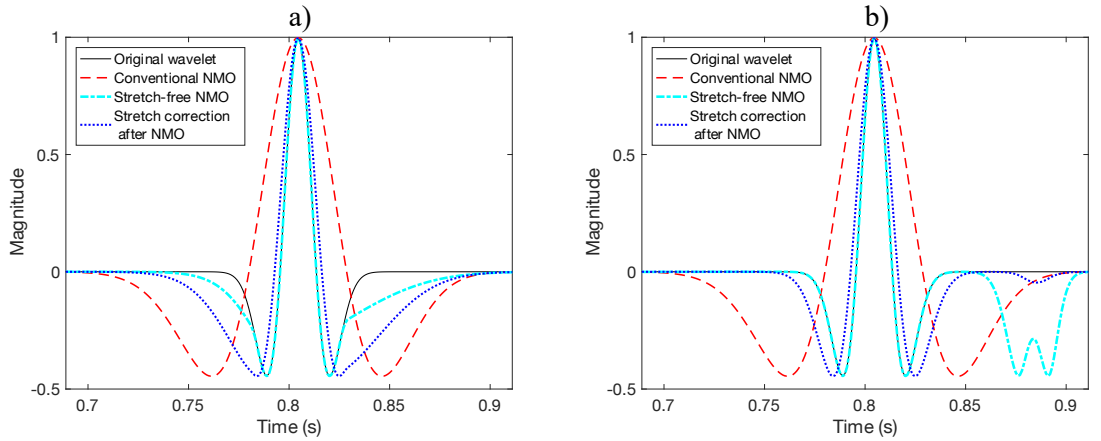

Figure 4. The effect of errors in selecting the wavelength for generating the partially constant parameter functions. a) The wavelength is selected as half the actual value. b) The wavelength is 1.8 times its actual value. To be compared to the correct wavelength selection presented in Figure 3a.

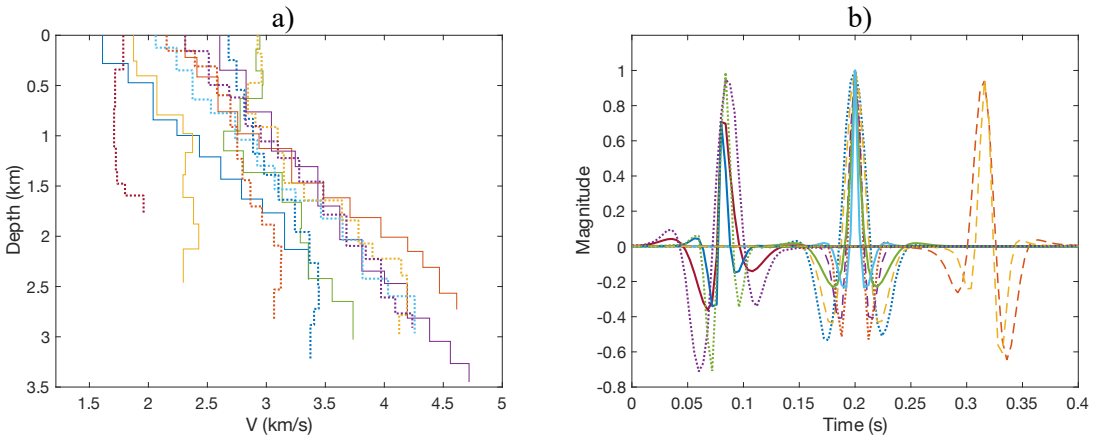

Figure 5. Representative examples of different velocity profiles (a), and different wavelets (b), used for generating the training data set.

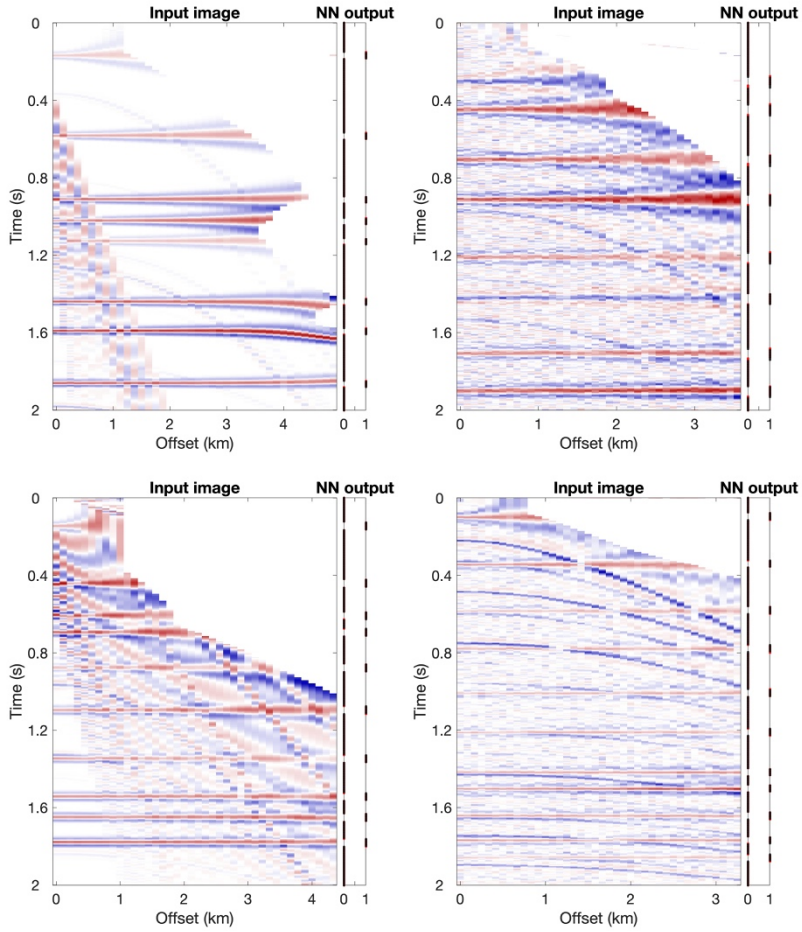

Figure 6. Four representative examples of the training data samples including the input images, and the neural network ground truth (black dots). The trained DNN predictions are also shown (red dots that are mostly covered by the black dots).

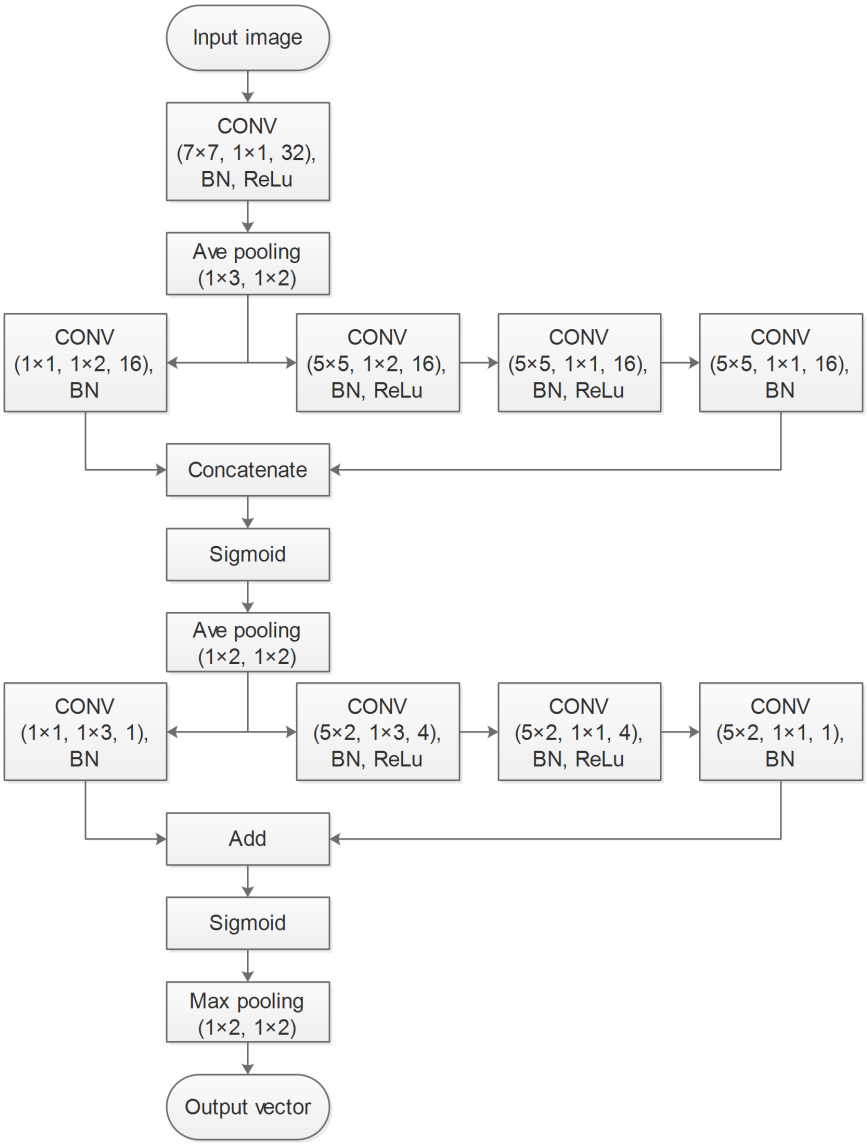

Figure 7. The architecture of our neural network, built from 2D convolutional (CONV), Batch normalization (BN), Average pooling, and Maximum pooling layers. We show in parenthesis the kernel size, stride size, and the number of filters, respectively. The input image size is 501×40, and the output size is 501×1.

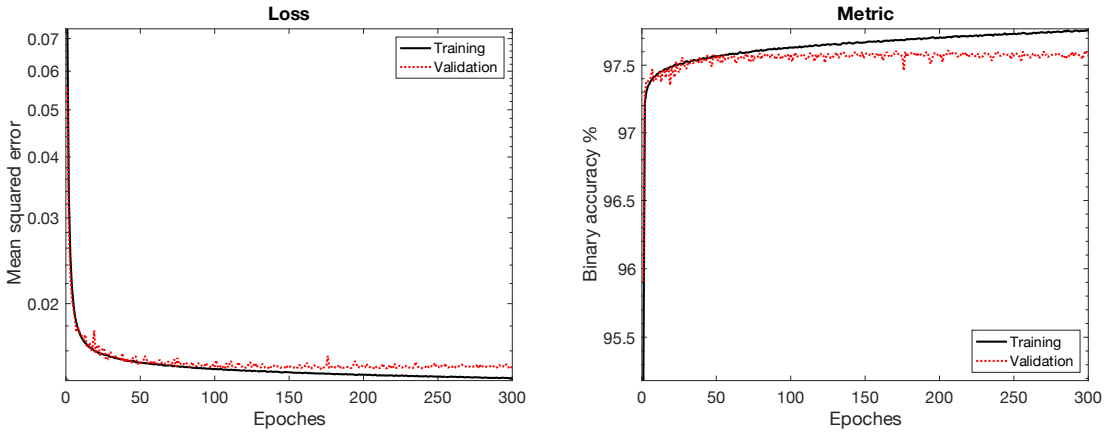

Figure 8. Evolution of loss and metric values during the training process. Since both the loss and metric values for the training data start to depart from the validation data set around epoch 75, we use the trained network after 75 epochs for our tests.

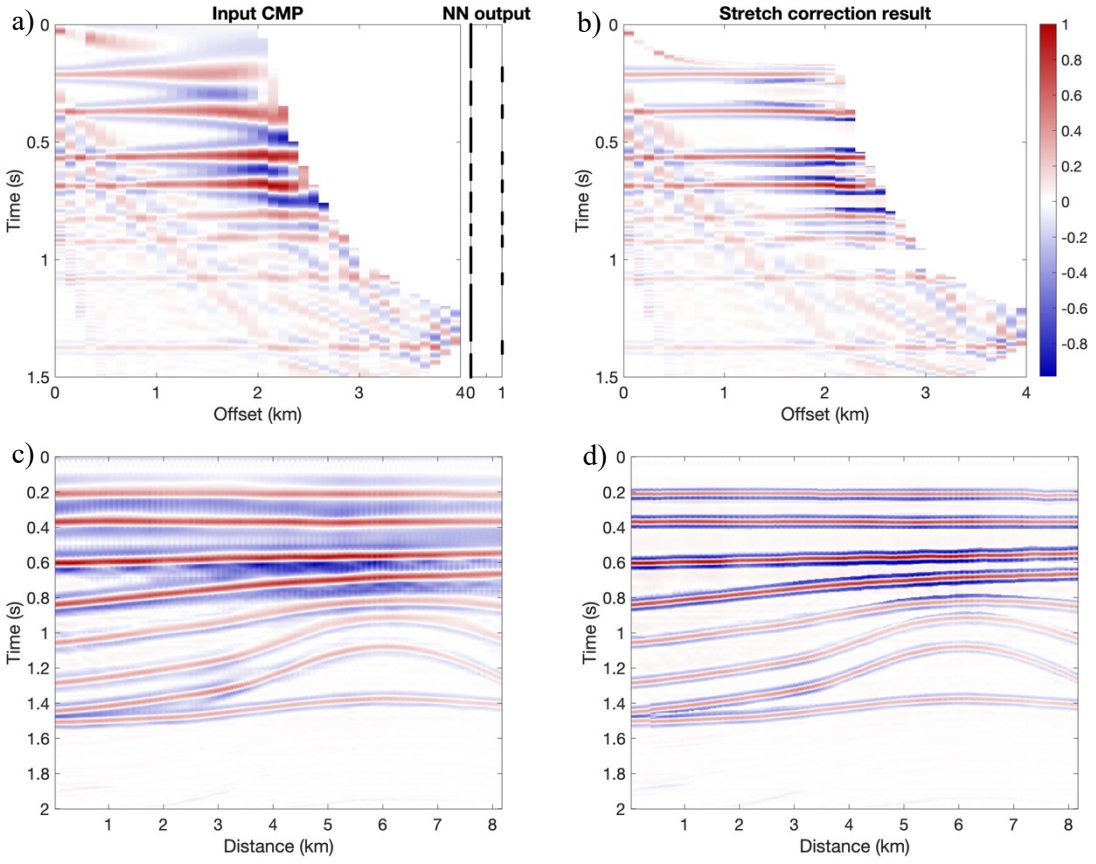

Figure 9. Nonhyperbolic NMO stretch correction on a finite-difference generated synthetic dataset. (a) A representative CMP gather after conventional NMO correction, which serves as the input of our trained neural network, and the neural network prediction. (b) The stretch-corrected version of the CMP in the panel (a), using the partially constant NMO parameters obtained by mean of the neural network. (c) Stacked section of conventional NMO corrected CMPs. (d) Stacked section of NMO corrected CMPs after stretch correction. Reflections are sharper and the wavelet shapes are better preserved.

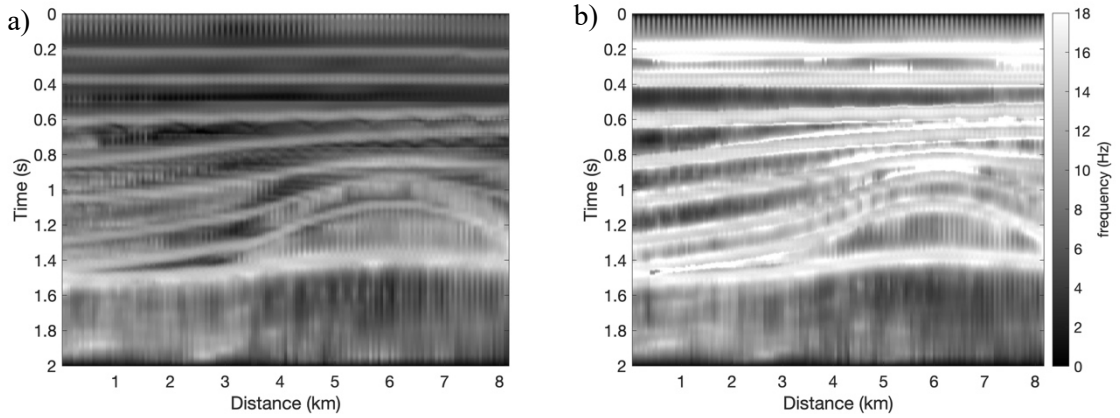

Figure 10. The local frequency content of the stacked sections (a) before stretch correction (Figure 9c), and (b) after stretch correction (Figure 9d). After our stretch correction, the events have higher frequency contents.

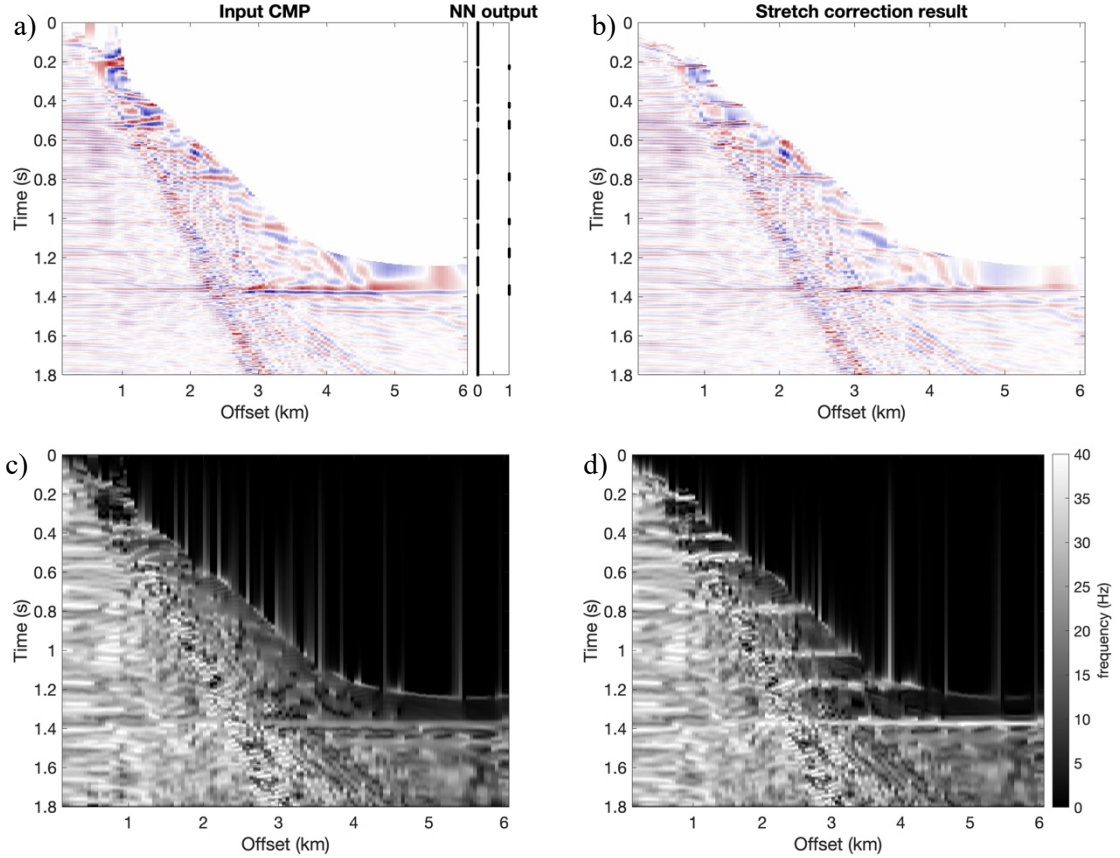

Figure 11. Application of the proposed method on a real marine CMP gather. (a) The input CMP is NMO corrected by GMA (equation 1). Our trained DNN identifies the zero-offset time samples over the primary reflections. (b) Stretch correction, performed by equation 3. (c) The local frequency content of part (a). (d) The local frequency content of part (b). The stretch-corrected primary reflections have higher frequencies.

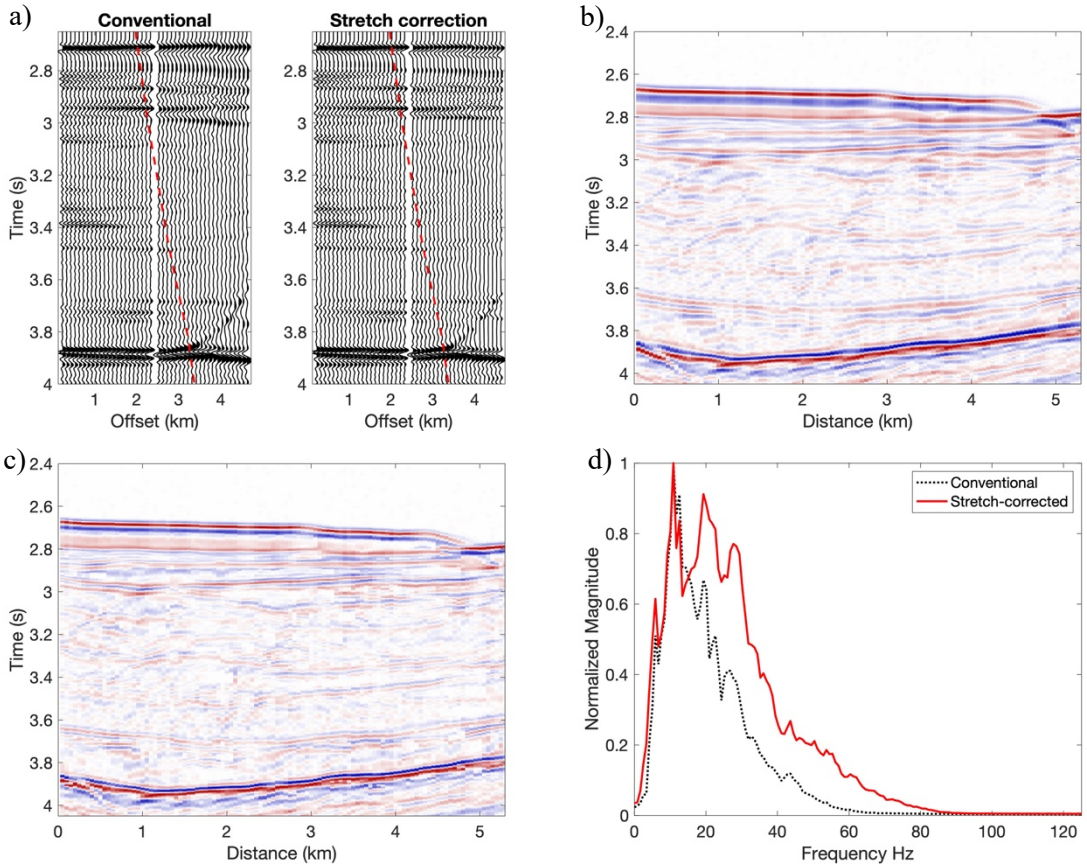

Figure 12. Application of our proposed stretch-correction method on a set of deep marine data. (a) Representative CMP gathers before and after stretch correction. (b) Large-angle stacked section of conventionally NMO-corrected data with GMA. (c) Large-angle stacked section of the data after stretch correction. (d) Mean amplitude spectra of parts (b) and (c). The dashed red lines show the stacked region of each CMP.