SAND2024-05826J

Denoising Seismic Waveforms Using a Wavelet-Transform-Based Machine-Learning Method

Louis Quinones[*1] and Rigobert Tibi[1]

ABSTRACT

Seismic waveform data recorded at stations can be thought of as a superposition of the signal from a source of interest and noise from other sources. Frequency-based filtering methods for waveform denoising do not result in desired outcomes when the targeted signal and noise occupy similar frequency bands. Recently, denoising techniques based on deep-learning convolutional neural networks (CNNs), in which a recorded waveform is decomposed into signal and noise components, have led to improved results. These CNN methods, which use short-time Fourier transform representations of the time series, provide signal and noise masks for the input waveform. These masks are used to create denoised signal and designaled noise waveforms, respectively. However, advancements in the field of image denoising have shown the benefits of incorporating discrete wavelet transforms (DWTs) into CNN architectures to create multilevel wavelet CNN (MWCNN) models. The MWCNN model preserves the details of the input due to the good time–frequency localization of the DWT. Here, we use a data set of over 382,000 constructed seismograms recorded by the University of Utah Seismograph Stations network to compare the performance of CNN and MWCNN-based denoising models. Evaluation of both models on constructed test data shows that the MWCNN model outperforms the CNN model in the ability to recover the ground-truth signal component in terms of both waveform similarity and preservation of amplitude information. Model evaluation of real-world data shows that both the CNN and MWCNN models outperform standard band-pass filtering (BPF; average improvement in signal-to-noise ratio of 9.6 and 19.7 dB, respectively, with respect to BPF). Evaluation of continuous data suggests the MWCNN denoiser can improve both signal detection capabilities and phase arrival time estimates.

KEY POINTS

- We assess whether wavelet transforms can improve denoising performance in seismic denoising algorithms.
- We find that wavelet transforms do improve denoising performance by improving signal information retention.
- The improved seismic denoising model may allow for more signal detections at sites with high noise intensity.

INTRODUCTION

A recorded seismic waveform can be viewed as a superposition of energy radiated by a defined source of interest, referred to as the "signal" component, and energy produced by all other sources, referred to as the "noise" component. The contamination of recorded data by noise from unwanted sources can lead to issues in subsequent analyses, caused by the deterioration of the quality (low signal-to-noise ratio [SNR]) and the distortion of the amplitudes of the target signals. Thus, the suppression of noise in the waveform data is an important preprocessing step of the seismic analysis workflow. Noise suppression methods include frequency-based filtering techniques, approaches based on component analysis (Bharadwaj *et al.*, 2017), continuous wavelet transform (CWT) thresholding (Ruikar and Doye, 2011; Mousavi and Langston, 2016, 2017; Tibi *et al.*, 2023), deep-learning-based denoising (Zhu *et al.*, 2019; Tibi *et al.*, 2021), and many other techniques (Hennenfent and Herrmann, 2006; Oropeza and Sacchi, 2011; Han and van der Baan, 2015; Liu *et al.*, 2015; Chen *et al.*, 2017; Huang *et al.*, 2017; Yuan *et al.*, 2018). Typically, the goal of noise suppression is the improvement of SNR values to better detect low-amplitude signals of interest. However, standard noise suppression techniques such as frequency filtering do not lead to the desired outcomes when the frequency bands of the signal of interest and noise overlap. In addition, noise suppression techniques can cause distortion of amplitudes and increased uncertainty in signal polarities, leading

1. Sandia National Laboratories, Albuquerque, New Mexico, U.S.A., https://orcid.org/0000-0002-4784-3940 (RT)

*Corresponding author: laquino@sandia.gov

Cite this article as Quinones, L., and R. Tibi (2024). Denoising Seismic Waveforms Using a Wavelet-Transform-Based Machine-Learning Method, *Bull. Seismol. Soc. Am.* **XX**, 1–12, doi: 10.1785/0120230304

to issues further downstream in the processing pipeline (Herrmann, 1973).

Convolutional neural network (CNN)-based deep-learning denoisers have shown some success in resolving these issues (e.g., Tibi *et al.*, 2021). Previously developed CNN-based seismic denoisers used short-time Fourier transforms (STFTs) of seismic waveform data (Zhu *et al.*, 2019; Tibi *et al.*, 2021) in a network with a U-style architecture that consists of a series of fully convolutional layers (Ronneberger *et al.*, 2015). In this approach, inputs are transformed into compact but deep representations, which are subsequently deconvolved into the desired model outputs. Recent advancements in the subfield of computer vision focused on image restoration have shown that incorporating discrete wavelet transform (DWT) functions into CNNs can lead to improvements in performance for tasks such as image denoising (Liu *et al.*, 2019; Tian *et al.*, 2023). There is, however, a trade-off between the improvement in performance and computational efficiency following the incorporation of DWT. Nevertheless, with DWT and the corresponding inverse wavelet transform (IWT), the information retention across the network layers far exceeds that of CNN involving Fourier transforms (Liu *et al.*, 2019).

Here, we construct a CNN that uses wavelet-transform functions, which will hereafter be referred to as the multilevel wavelet CNN (MWCNN). Like the STFT-based denoiser network (Tibi *et al.*, 2021), the MWCNN decomposes an input waveform into its signal and noise components. To train and evaluate the MWCNN denoising model, we use a curated data set of signal and noise waveforms recorded by the University of Utah Seismograph Stations (UUSSs) monitoring network (The University of Utah, 1962). The performance of the MWCNN denoising model is compared to that of a fully CNN (FCNN) denoising model, which uses only convolution-based model layers and uses the same sets of constructed test and real-world seismic waveform data in the appropriate input formats.

METHODS AND DATA

For a given input waveform, the MWCNN provides a pair of signal and noise mask operators that are applied to the input waveform to estimate the signal and noise components, respectively. We follow the same general waveform data processing procedure outlined in Zhu *et al.* (2019) and Tibi *et al.* (2021), with the notable difference being that instead of using STFT representations, we use wavelet-transform representations of the input waveforms constructed using CWT functions. The purpose of using wavelet-transform representations is to achieve better timescale fidelity between the MWCNN denoiser inputs and outputs. All data used to construct the training, validation, and test data sets were collected from a set of selected UUSS network stations from the associated Utah event bulletin covering the period from 2009 to 2018 (Fig. 1). The methodology of combining defined "signal" and "noise" waveform samples to construct "noisy" waveform data is expanded

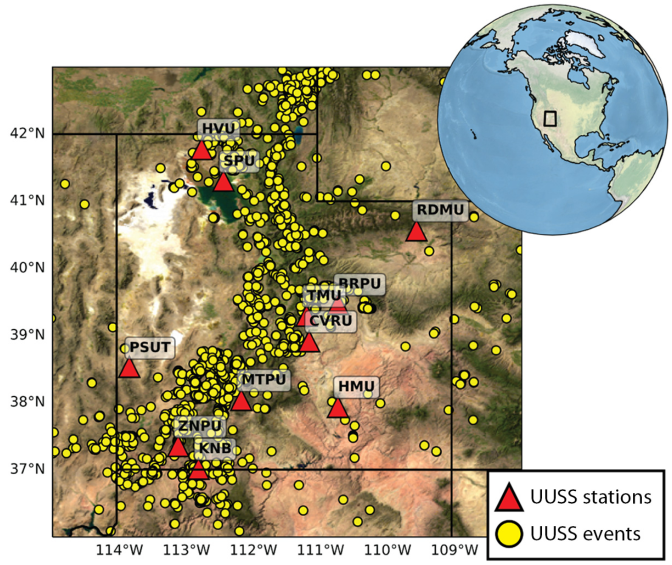

Figure 1. Map showing the locations of the selected University of Utah Seismograph Stations (UUSSs) (triangles) and catalog events (circles, 2009–2018) used to construct the training, validation, and test data sets used in this study. (Inset) Global map with the study region outlined by the black box. The color version of this figure is available only in the electronic edition.

upon in the Data construction section. If we assume that the recorded seismic waveform data can be represented as a superposition of the defined signal and noise components, then the noisy waveform data can be represented in the timescale domain as

$$D(a,\tau) = S(a,\tau) + N(a,\tau), \tag{1}$$

in which $D(a,\tau)$ represents the CWT of the noisy waveform data, with a being the scale and τ the time lag; $S(a,\tau)$ is the CWT of the noise-free signal component; and $N(a,\tau)$ is the CWT of the pure noise component. The CWT for each 60 s waveform component was calculated using a Morlet wavelet (Büssow, 2007) with 10 voices and a sampling rate of 100 Hz. The CWT, $X(a,\tau)$, for a continuous waveform, $x(t)$, using a mother wavelet function, $\psi(t)$, is calculated using the following:

$$X(a,\tau) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\Psi^{*}\left(\frac{t-\tau}{a}\right)dt, \tag{2}$$

in which $\Psi^{*}(t)$ is the complex conjugate of the wavelet function, meaning the wavelet transform is the cross correlation of a continuous waveform with a set of translated versions of the mother wavelet function. To save on computational costs for model training and evaluation, these CWTs are downsampled in time. The MWCNN model uses the CWT representation as input and outputs a signal mask, $M_S(a,\tau)$, and a noise mask,

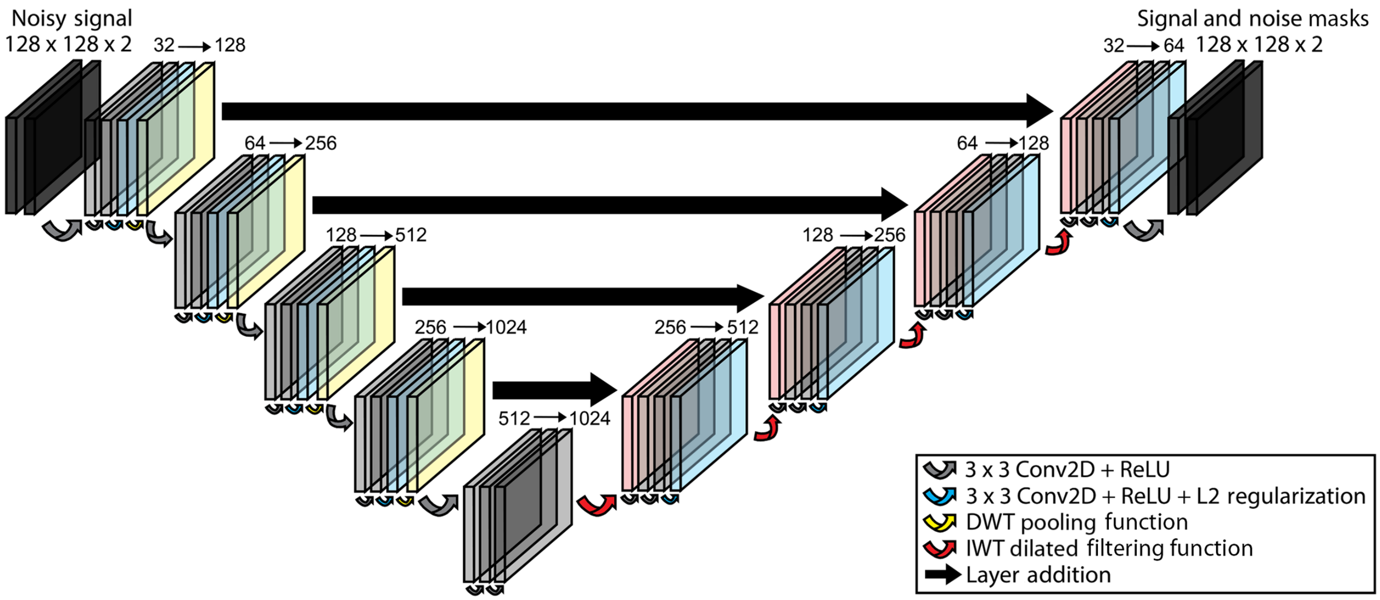

Figure 2. Architecture of the multilevel wavelet convolutional neural network (MWCNN) with each network layer represented by a rectangular prism with the number of channels associated with each block of layers is given. The color of each network layer and associated arrow represents the processes used in the layer, as described in the legend. Processes represented in blue arrows include an L2-norm regularization function. Processes represented by green and red arrows indicate the utilization of discrete (DWT) and inverse (IWT) wavelet transform functions, respectively. The rectified linear unit (ReLU) activation function is used in all 2D convolution layers. When a DWT is involved, the number of channels is halved, and when an IWT is used, the number of channels is decreased by a factor of 4. The color version of this figure is available only in the electronic edition.

$M_N(a,\tau)$. These masks are the training targets for the MWCNN, which, when multiplied elementwise with the CWT of the input waveform, provides the CWT estimates of the signal, $\hat{S}(a,\tau)$, and noise, $\hat{N}(a,\tau)$, components, respectively, as indicated in the following:

$$\hat{S}(a,\tau) = M_S(a,\tau) \times D(a,\tau), \tag{3}$$

$$\hat{N}(a,\tau) = M_N(a,\tau) \times D(a,\tau). \tag{4}$$

The mask operators themselves are matrices of values between 0 and 1. In training the network, the masks as targets are calculated using the training set as follows:

$$M_S(a,\tau) = \frac{|S(a,\tau)|}{|S(a,\tau)| + |N(a,\tau)|}, \tag{5}$$

$$M_N(a,\tau) = \frac{|N(a,\tau)|}{|S(a,\tau)| + |N(a,\tau)|}. \tag{6}$$

The estimated signal, $\hat{S}(a,\tau)$, and noise, $\hat{N}(a,\tau)$, components in the timescale domain are transformed into the time domain using the IWT.

Network architecture

The MWCNN follows a U-net-style architecture made up of blocks of convolutional layers bookended by wavelet-transform layers. The MWCNN architecture shown in Figure 2 is adapted from both the image restoration network architecture presented in Liu *et al.* (2019) and the denoiser network reported in Tibi *et al.* (2021), seeking to combine aspects of both to improve seismic denoising performance. The MWCNN architecture is designed around using convolutional layers to extract features from the input, which are then downsampled using a DWT in place of standard pooling functions. Pooling functions are used in CNN architectures to reduce the dimensions of feature maps, thereby reducing computational costs and helping avoid overfitting issues. However, the use of pooling functions can also lead to information loss due to oversmoothing of the feature maps. The benefit of using DWT functions over standard pooling functions is the ability of the former to retain high-frequency information and their fully invertible nature (Liu *et al.*, 2019). To illustrate this, we examine a 2D DWT function, which can be mathematically represented as a set of four fixed convolutional filters that decomposes the (i,j) values of some initial image, x, into four sub-band images reflecting one low-pass filter, x_{LL}, and three high-pass filters, x_{LH}, x_{HL}, and x_{HH} (Mallat, 1989). Using the Haar wavelet, these four filters can be represented by the following set of equations:

$$\begin{cases} x_{\mathrm{LL}} = x(2i-1,2j-1) + x(2i-1,2j) + x(2i,2j-1) + x(2i,2j) \\ x_{\mathrm{LH}} = -x(2i-1,2j-1) - x(2i-1,2j) + x(2i,2j-1) + x(2i,2j) \\ x_{\mathrm{HL}} = -x(2i-1,2j-1) + x(2i-1,2j) - x(2i,2j-1) + x(2i,2j) \\ x_{\mathrm{HH}} = x(2i-1,2j-1) - x(2i-1,2j) - x(2i,2j-1) + x(2i,2j) \end{cases}. \tag{7}$$

In addition, due to the DWT being biorthogonal, the original image x can then be reconstructed without information loss by simply using the IWT on the four generated sub-band images,

$$x = \mathrm{IWT}(x_{\mathrm{LL}}, x_{\mathrm{LH}}, x_{\mathrm{LH}}, x_{\mathrm{HH}}). \tag{8}$$

Meanwhile, a standard average pooling function with a factor of 2 acting on some feature map, x_l, within the lth layer of the model can be represented using the mathematical function

$$x_l(i,j) = \frac{x_{l-1}(2i-1,2j-1) + x_{l-1}(2i-1,2j) + x_{l-1}(2i,2j-1) + x_{l-1}(2i,2j)}{4}. \tag{9}$$

This pooling function expression is analogous to the low-frequency portion of the DWT function. This means that much of the high-frequency information of the image is lost when standard average pooling functions are implemented within CNN models. As such, the incorporation of the DWT into our network architecture to replace the use of standard pooling functions leads to less information loss across the layers while also leading to potential benefits in feature reconstruction due to the invertible nature of the DWT function.

The MWCNN denoiser inputs are the normalized real and imaginary components of the CWT of the noisy waveform, which have been downsampled to a size of 128 × 128 × 2, in which the last number indicates the number of feature map channels. Through each downsampling block of convolutional layers, the number of feature map channels is expanded as the overall feature map size is compacted to attain deeper representations of the inputs. Each block of convolutional model layers uses 2D convolution with a 3 × 3 weight kernel passed through a rectified linear unit activation function. To help counter possible overfitting, the final convolution layer in each block has an L2-regularization function applied. Finally, each downsampling block is ended by a DWT layer. In total, there are four downsampling blocks in the MWCNN denoiser, making up a total of 16 layers (Fig. 2). Between the downsampling and upsampling portions of the network, we include one final set of three convolutional layers. The upsampling blocks follow a similar structure, with the difference being that the IWT function layer, which upsamples the feature map, is the first layer applied in each of the four upsampling blocks, making up a further 16 layers. The change in the number of feature map channels is denoted above each of the blocks. In contrast to a typical U-net architecture, skip connections are created by performing elementwise addition rather than concatenation, following the structure given in Liu *et al.* (2019). Skip connections are denoted as black arrows in Figure 2.

The MWCNN denoiser was built using the Keras library (Chollet, 2015) within the larger TensorFlow module (Abadi *et al.*, 2016). The model was compiled to minimize the categorical cross-entropy loss function using the Adam optimization algorithm (Kingma and Ba, 2014). The MWCNN denoiser contains a total of ∼18 million trainable parameters. The total number of layers was chosen to minimize overfitting while providing sufficient capacity for adequate performance of the resulting denoiser model. The performance of the MWCNN denoiser is compared to that of an FCNN denoising model that was trained using the network architecture described in Tibi *et al.* (2021) and the training and validation sets used here.

Data construction

The MWCNN model is trained to be able to denoise seismic waveform data recorded on any component of any UUSS network station. Thus, to adequately train the denoising model, we compiled a set of high-SNR local and regional waveforms, which serve as our noise-free target signal waveforms, and characteristic noise waveform samples from all three components of a set of 11 stations from across the UUSS network. These stations, shown in Figure 1, are representative of the UUSS network monitoring region over the examined period of 2009–2018. The compiled data cover a wide range of signal and noise environments from which the MWCNN network may learn. To compile the set of high-SNR waveforms, we used the UUSS event bulletin (The University of Utah, 1962), which contained over 30,000 events over the examined period. These regional waveforms make up our potential signal data set and are composed of detections made from events with $0.0 \le M_c \le 5.3$ and epicentral distances of ≤670 km. Each signal waveform was detrended, demeaned, and, to improve the SNR, band-pass filtered between 1 and 20 Hz. A waveform was considered for potential use as a high-SNR signal waveform if its SNR exceeded the threshold value of 4.0. Waveform recordings from all three channels of each of the 11 selected stations were examined, yielding a total of 8284 potential high-SNR signal waveforms across the three station channels. These waveforms, which were cut into 60 s segments with the predicted *P*-wave arrival time occurring at the 10 s mark for each segment, were then manually reviewed to determine whether a clear signal arrival could be seen. This review process eliminated a portion of the high-SNR waveforms, leaving a total of 6376 waveforms, with 1913, 1594, and 2869 coming from the HHE, HHN, and HHZ channels, respectively. Meanwhile, the data set of noise waveforms (i.e., no obvious signal present) is composed of a set of 15,246 waveforms, which were manually selected by an expert analyst from a broader spectrum of UUSS network stations to represent the range of noise sources that would be observed in the Utah region (Tibi *et al.*, 2021). Each of the noise waveforms is also a 60 s segment of data. The signal and noise waveforms are all sampled at 100 Hz.

The first preprocessing performed is the splitting of the signal and noise waveform data sets into the training, validation, and test sets following a 70–15–15 partitioning. This partitioning is performed first to ensure that no signal or noise

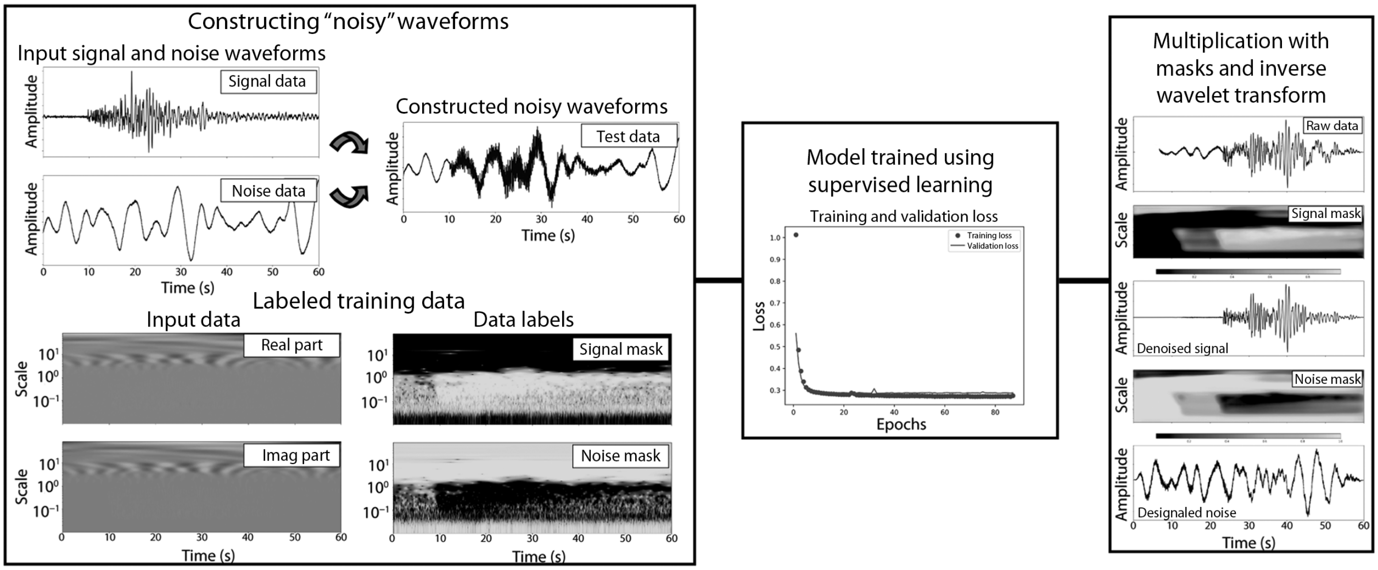

Figure 3. Schematic view of the processing steps taken to train the MWCNN denoising model and denoise waveforms. The preprocessing steps (left panel) consist of constructing noisy waveforms from selected signal and noise waveforms and generating normalized scalograms of the noisy waveforms and signal and noise masks to be used as training inputs and labels, respectively. The MWCNN denoising model is trained using supervised learning, and the best model was obtained after 87 epochs (middle panel). The post-processing steps (right panel) involve multiplying the signal and noise masks provided by the network with the noisy waveform scalograms and then using the inverse wavelet transform to generate the final denoised and de-signaled waveforms.

waveforms that were used to train the denoising model would also be used to validate or test the model's performance. The process for preparing the collected waveform data for use in training and evaluating the MWCNN denoising model includes randomly varying the timing of the signal arrival with respect to the beginning of the waveform, modifying the noise amplitudes, as discussed in the following, and summing the signal and noise waveforms to construct the noisy waveforms. In varying the timing of the signal arrival, the time-shifted precut waveforms were padded with values of 0 at either the beginning or end of the 60 s segment, depending on the among and the direction of the shift. To construct the noisy waveforms, we select a signal waveform and sum it with a randomly selected noise waveform from the same set in a process illustrated in the preprocessing step in Figure 3. This process is conducted 60 times for each signal waveform in each set, resulting in 267,780 noisy waveforms for the training, 57,420 for the validation, and 57,360 for the test data sets. To ensure that the timing of the signal arrival is not "backed" into the MWCNN model, as mentioned earlier, we randomly vary the signal arrival time for each summation run. Overall, the signal arrival times were uniformly distributed throughout the 60 s window of the waveform segments, with some allowed to occur before or after the waveform segments began or ended, respectively, to simulate scenarios in which there would be little or no signal energy present. To increase the potential SNR range of waveforms the MWCNN denoiser can successfully denoise, we also modify the amplitude of each noise waveform when they are selected. This is accomplished by randomly selecting some scale value, v, which ranges from 0.5 to 5.0, and solving for the noise scaling factor, k, using the following:

$$k = \frac{s_{\max}}{v \times n_{\max}}, \tag{10}$$

in which $s_{\max}$ and $n_{\max}$ are the maximum signal and noise amplitudes (in absolute sense), respectively. The noise waveform is then scaled by k before it is used to construct the noisy waveform.

Model training

The MWCNN denoising model was trained in a supervised learning manner, using the CWTs of the constructed noisy waveforms as inputs and the calculated signal and noise masks as labels. For each noisy waveform, we generate a CWT and then split it into its real and imaginary components. The calculated and min–max normalized real and imaginary components for the training and validation sets constitute the inputs for model training. An example of these inputs and their associated labels is shown in the preprocessing panel in Figure 3.

The MWCNN model was trained using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001, a batch size of 256, and a maximum number of training epochs of 400. The model was trained to minimize the categorical cross-entropy loss function while setting a patience of 30 epochs for early stopping, monitored by the validation loss value. The model hyperparameters were tuned to minimize overfitting. The best-performing hyperparameter values from a set of limited data set training tests were selected for use in the full model training. The L2-regularization functions were incorporated into several layers of the network (Fig. 2), and early stopping was added to prevent overfitting. In total, the model training was completed after 87 epochs, which took approximately 18 hr using four 32 GB GPU cores on a Linux GPU-accelerated machine.

The product of the training process is an MWCNN denoising model, which can be applied to denoise sets of waveform data. To denoise a waveform, the trained MWCNN model is used to calculate the signal and noise masks corresponding to the noisy input waveform. The signal and noise mask operators are then applied to the wavelet transform of the input waveform; the resulting products are transformed back into the time domain to obtain the denoised and de-signaled waveforms, respectively (Fig. 3). It is these resulting signal and noise waveforms, which we use to evaluate the overall performance of the MWCNN denoising model when compared with the prior developed FCNN model and traditional band-pass filtering techniques.

RESULTS AND DISCUSSION

To evaluate the denoising ability of the model, we examine its performance on three separate data sets. The first data set to be evaluated is the test data set of constructed noisy waveforms, which were created using the same methodology as the training and validation data sets but contain different noise and signal components than were used for either of those. As we know what the desired model output signal and noise waveforms should be, we focus on the denoising model's ability to reconstruct both the shape and amplitude of the signal and noise waveforms. The second data set to be evaluated is a set of real-world event detections collected from the three channels of the 11 selected UUSS stations (Fig. 1) from events in the Utah event bulletin covering the period of 2018–2020 and not included in the SNR > 4.0 signal set described earlier that were used to construct the noisy waveforms. For this data set, we focus solely on the ability of the denoising model to increase the detection SNR value because we lack ground-truth measurements of what the true pure signal waveforms should appear as. The final data sets examined are a continuous set of 24 hr data collected from station BRPU, where we examine the denoiser's ability to increase signal detection capabilities. In addition, we examine the ability of the denoiser to aid in the refinement of phase arrival time picking for previously detected events.

Evaluation of constructed test data

We evaluate the performance of the MWCNN denoising model using two specific metrics: the normalized cross correlation coefficient (CC) and signal-to-distortion ratio (SDR; equation 11) values calculated from the recovered and ground-truth waveforms. The CC value between the recovered waveform and ground-truth waveforms is used to measure the similarity of the two waveforms to determine whether there are any changes in waveform shape associated with the denoising process. The SDR value is used to measure the degree of amplitude distortion caused by the denoising process using the following equation taken from Nakajima *et al.* (2018):

$$\mathrm{SDR} = 10\log_{10}\frac{||W_{\mathrm{GT}}^2||}{||W - W_{\mathrm{GT}}^2||}, \tag{11}$$

in which the array of amplitudes of the recovered waveform is represented as W, and that of the ground-truth waveform as W_{GT}. Using this expression, the SDR value goes to infinity as the amplitudes of the recovered and ground-truth waveforms become more similar. An example of the denoising evaluation is presented in Figure 4, wherein the resulting denoised signal and de-signaled noise waveforms were both able to achieve CC values exceeding 0.98 and SDR values exceeding 12.9.

To assess the benefits of incorporating wavelet transforms into the MWCNN architecture, we compare the resulting denoising and de-signaling performance of the MWCNN model to that of a prior developed FCNN denoising model. Both the MWCNN and FCNN models learned from the same training and validation data sets and were evaluated on the same set of 57,360 constructed test waveforms. Figure 5 presents the denoising results of both models using the described metrics. Here, we are interested in comparing the degree of improvement in both CC and SDR metrics that each model was able to achieve when compared with the original noisy test data set waveforms. We can observe based on the evaluation metric values calculated on the constructed test waveforms (labeled as "Signal + noise" in Fig. 5) that they were more dominated by the noise component rather than the signal due to having higher average noise CC (0.75) and SDR (6.1) values than average signal CC (0.46) and SDR (−6.1) values. Both the MWCNN and FCNN denoising models were able to greatly improve upon both evaluation metric values for both the signal and noise components. The MWCNN model was able to slightly outperform the FCNN model in both the average signal CC (0.83 vs. 0.73) and SDR (9.7 vs. 8.8) evaluation metrics over the whole test data set by primarily showing improvement among the most initially noise-dominated test waveforms. Likewise, the MWCNN model was able to increase the similarity of the noise waveforms slightly when compared with the FCNN model (noise CC of 0.90 vs. 0.87) but underperformed slightly in amplitude retention ability (noise SDR of 10.3 vs. 11.4). The reason for this underperformance is most likely the extreme outlier test data set waveforms, which were almost entirely dominated by noise; those being test waveforms, which, even in their original noisy waveform state, had noise CC > 0.90 and SDR > 20.0 values. A fundamental assumption that accompanies the data generation method we used is that there should always be some mixture of signal and noise energy within a noisy waveform. This assumption leads to issues in cases in which a noisy waveform is almost entirely dominated by either the signal or noise component

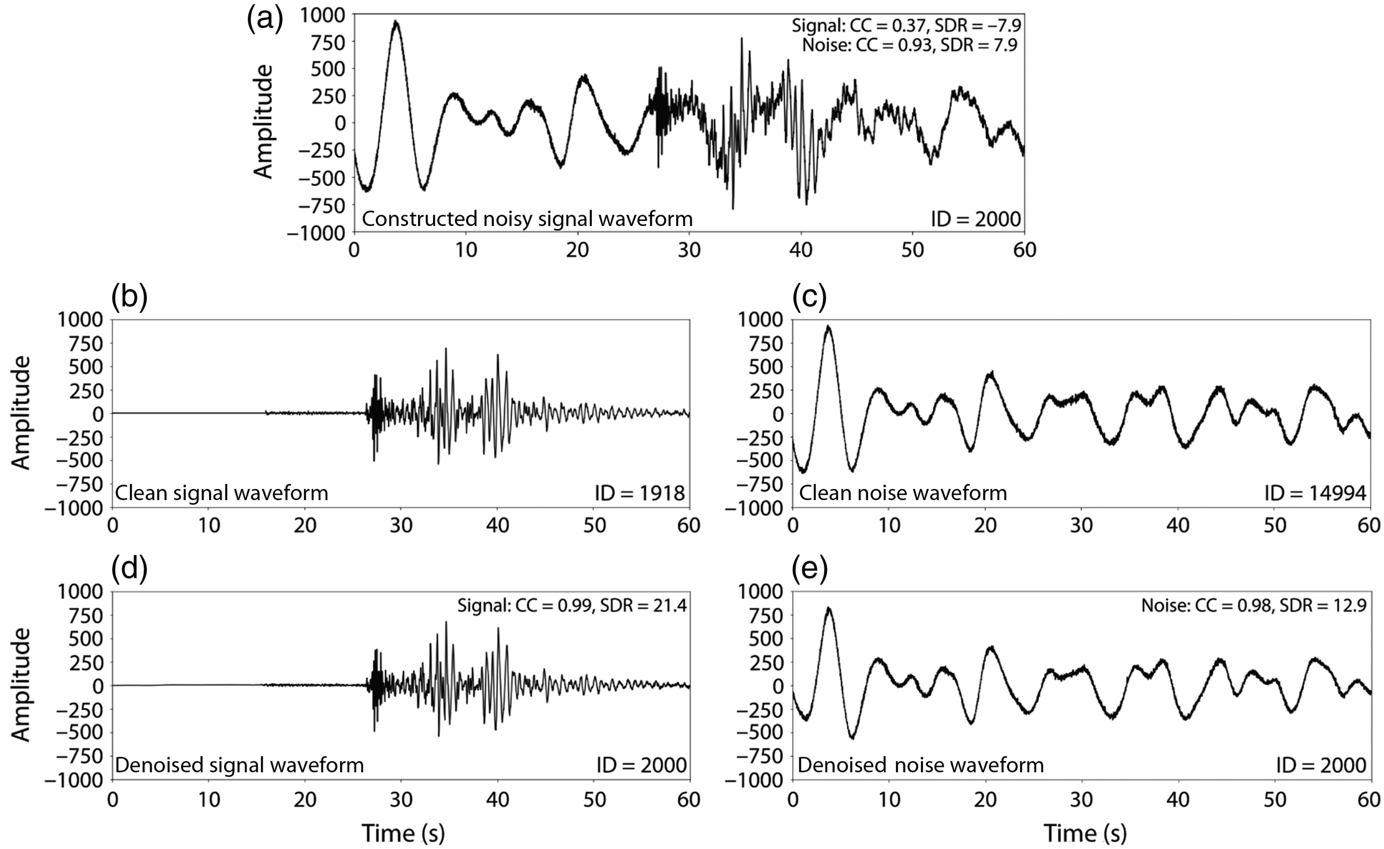

Figure 4. Example of test data set evaluation. (a) An example noisy waveform from the test data set with its ID number, signal cross-correlation coefficient (CC), signal signal-to-distortion ratio (SDR), noise CC, and noise SDR values is recorded. Ground-truth (b) signal and (c) noise waveforms used to construct the example noisy waveform. (d) A recovered (denoised) signal waveform with its CC and SDR values with respect to the ground-truth signal waveform is recorded. (e) A recovered (de-signaled) noise waveform with its noise CC and SDR values with respect to the ground-truth noise waveform is recorded.

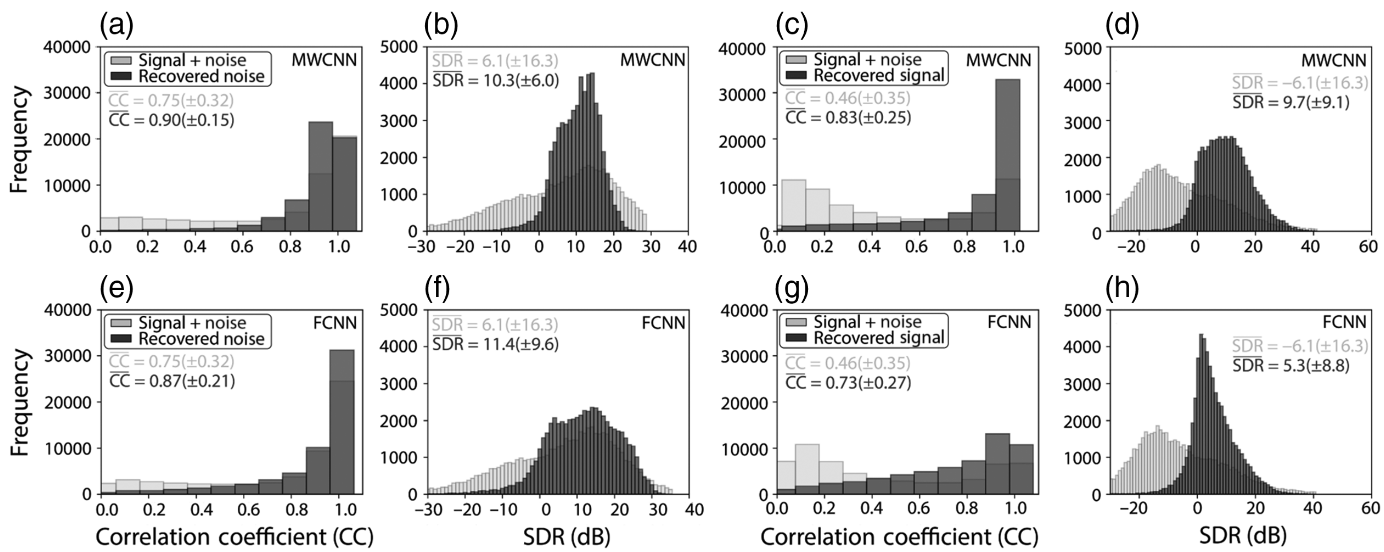

Figure 5. Histogram plots showing the distributions of (a,c,e,g) cross correlation coefficient (CC) and (b,d,f,h) SDR values calculated from the test data set using the (a–d) MWCNN and (e–f) fully convolutional neural network (FCNN) denoising models compared to the initial evaluation metric values calculated from the raw test data set. The average evaluation metric values calculated from the raw and processed test datasets are recorded.

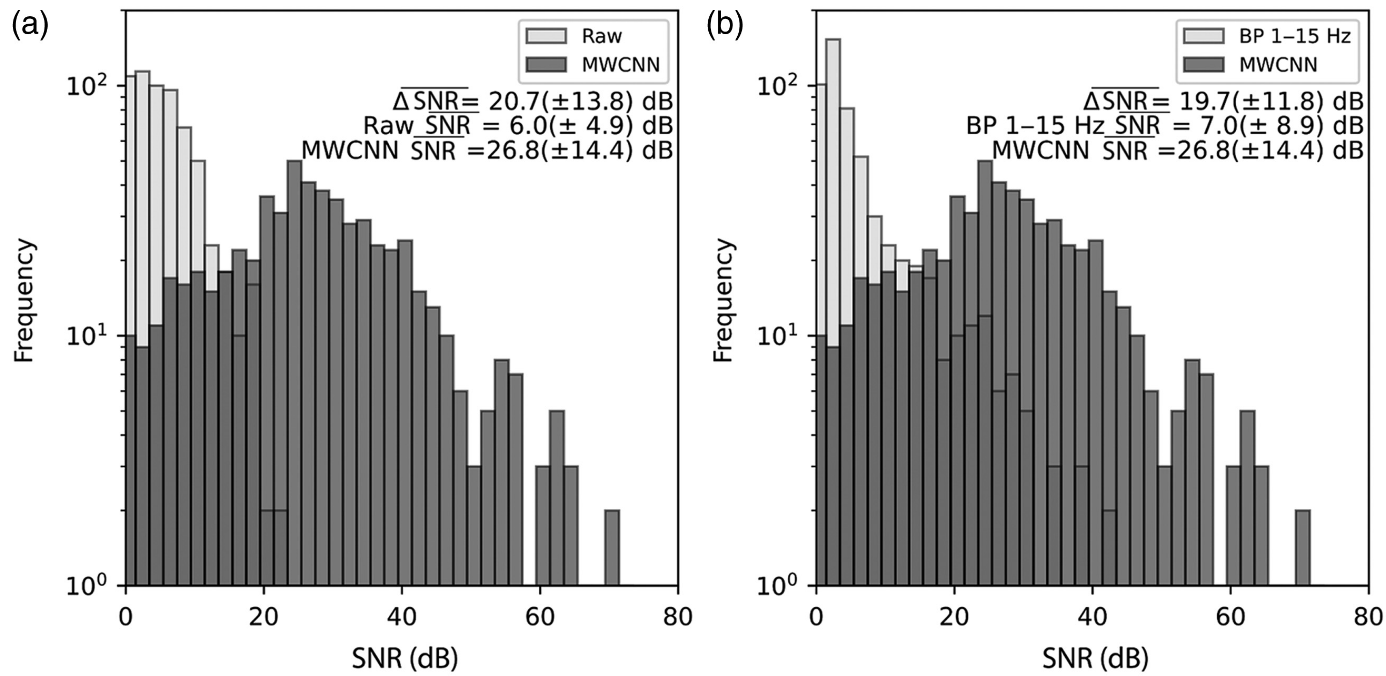

Figure 6. Distribution of the signal-to-noise ratio (SNR) values calculated from a set of 611 signals detected on the HHZ channel of station BRPU. Comparison of the SNR values calculated from (a) the raw and denoised waveforms and (b) the band-pass filtered (1–15 Hz) and denoised waveforms. The average difference in SNR values between the denoised and filtered waveform data, along with the average SNR values of each individual set of waveform data, is recorded.

due to the denoising model attempting to shift some of that energy to the less dominant component. We attempted to account for these types of situations by varying the noise amplitudes to be in a much wider range; however, most of the training data set still contained a more mixed amount of signal and noise energy that the denoising models then learned from. This issue can be seen in the histogram plots of the SDR evaluation results for the MWCNN model, wherein (more so with the noise component) the original noisy test data set contained a far wider range of SDR values than the resulting data set of MWCNN-denoised waveforms (Fig. 5b,d). The issue can also be observed in the FCNN denoising results, albeit to a lesser extent in the case of the noise component (Fig. 5f,h). This could potentially be alleviated by expanding upon the variability of the training data set in the future to allow for more one-component (noise or signal) dominated waveform data.

Evaluation of segmented real-world data with known detections

The second evaluation data sets comprise recorded signal detections collected from selected UUSS stations. Here, we evaluate the ability of the MWCNN denoising model to outperform traditional frequency filtering techniques using a set of waveform data collected from station BRPU. Then, we evaluate the denoising performance of the MWCNN denoising model when compared with the prior developed FCNN model using waveform data collected from all three components of 11 selected UUSS stations. Both denoising models were trained using the same waveform data sets converted into CWT and STFT representations for the MWCNN and FCNN denoising models, respectively. The first set of data consists of 611 waveforms with known detections recorded on the HHZ channel of station BRPU from the Utah event bulletin (2018–2020), for which we evaluate the ability of the MWCNN denoising model to improve SNR values. Here, we examine the ability of the MWCNN model to increase the detectability of signals when compared with other denoising techniques such as band-pass filtering. In all cases, the SNR value (in decibels) for a waveform is calculated using the following:

$$\mathrm{SNR} = 20 \log_{10} \frac{S_A}{N_A}, \tag{12}$$

in which S_A represents the root-mean-square (rms) value of the waveform amplitudes over a 9 s window beginning at the predicted phase arrival time and N_A represents the rms value of the waveform amplitudes over a 9 s window ending 1 s prior to the phase arrival time. We calculated the SNR values for all 611 signals detected over the designated period using the raw waveforms, 1–15 Hz band-pass filtered waveforms, and MWCNN-denoised waveforms. The 1–15 Hz frequency range was selected due to it being the best performing band-pass filter on previously examined UUSS network stations (Tibi *et al.*, 2021). Overall, we found that the MWCNN denoising model was able to greatly improve upon the SNR values of the selected signal detections when compared with the raw and band-pass filtered waveform data sets by improving the average SNR values by 20.7 and 19.7 dB, respectively (Fig. 6). These results suggest that a trained

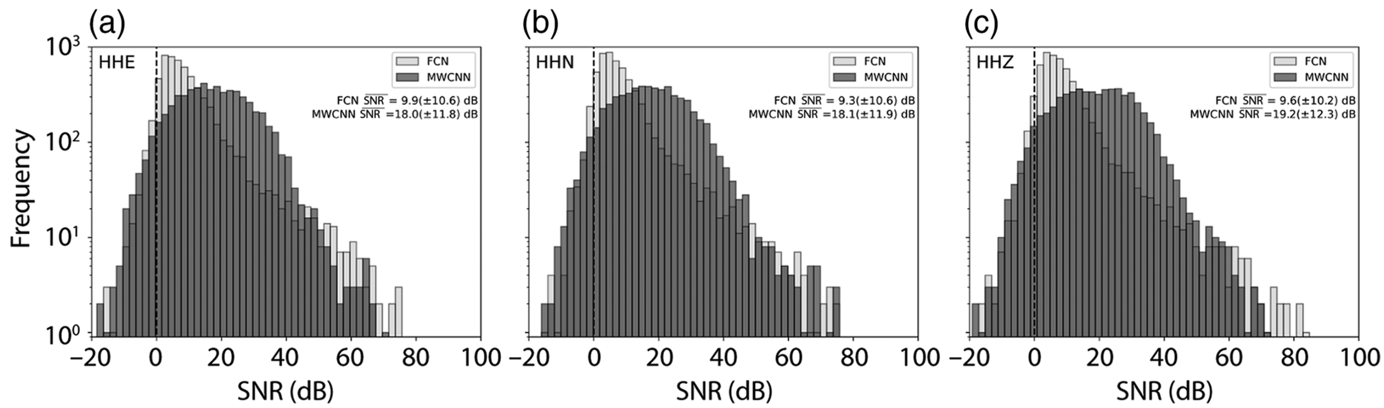

Figure 7. Histogram plots comparing the distributions in calculated SNR values generated from MWCNN and FCNN (see legend) denoised waveforms for a set of (a) 5952 detections on the HHE channels, (b) 5895 detections on the HHN channels, and (c) 6080 detections on the HHZ channels of selected UUSS stations, respectively. The average SNR values from each set of denoised waveform data are also recorded.

MWCNN denoising model should be able to better improve signal detection capabilities on a station when compared with standard band-pass filtering techniques.

In addition to comparing the performance of the MWCNN denoising model to band-pass filtering techniques on real-world data sets, we also compare the MWCNN model performance to that of FCNN denoising models. In the Tibi *et al.* (2021) study, the FCNN model was trained solely on the vertical component of the singular BRPU station signal data. However, to improve the transportability and general denoising ability of the models, we have expanded the training data set to incorporate signal waveform data from all three components of a set of 11 UUSS network stations (Fig. 1). To accurately compare the denoising performance of both models, we also retrained the FCNN denoising model on this expanded training data set. Therefore, to properly examine the performance of both deep-learning denoising models on real-world data sets, we also needed to expand the evaluation data set. This new three-component evaluation data set consists of signal detections from the period 2018–2020 and contains a total of 17,927 waveforms, with 5952 from the HHE, 5895 from the HHN, and 6080 from the HHZ components, respectively. Denoising results show that the MWCNN model was able to produce higher average SNR values (18.0 dB) on the evaluation data set when compared with the FCNN model (9.9 dB) across the three components (Fig. 7). Both models were able to achieve a similar degree of denoising performance across the three components, although the MWCNN model performed better on the HHZ component (19.2 dB) when compared with the horizontal components (18.0 dB). The distributions of the denoising results again suggest that the MWCNN model is more prone to underestimating the signal energy for those extremely signal-dominated waveforms when compared with the FCNN model. The trade-off is that the MWCNN model creates more moderately to highly signal-dominated waveforms ($12.0 < \mathrm{SNR} < 50.0$) when compared with the FCNN model across all three components. Overall, the MWCNN denoising model outperforms the FCNN denoising model when examining UUSS signal detection data across all three components in the ability to increase the SNR value and thus lead to improved signal detection.

Evaluation of continuous data

The goal of using the MWCNN denoising model is to improve upon standard waveform preprocessing techniques, such as frequency filtering, to yield better signal detection capabilities. To evaluate the ability of the MWCNN denoising model to improve upon this metric, we examine a 24 hr segment of data in both its band-pass filtered and denoised forms using the same signal detection function. Our goal is to determine whether more signals could be detected on the denoised version of the 24 hr segment and to what degree the SNR values of detected signals are improved. Here, we use a basic short-time average over long-time average (STA/LTA) detector configured to function on data recorded on 19 September 2019 on the HHZ channel of station BRPU. This day was selected due to there being 12 events reported in the Utah Event Bulletin as occurring that day, meaning the region was seismically active that day and that there could potentially be missed signal detections. After performing a limited search for the best-performing detection parameters on both the raw and denoised data sets, the STA/LTA parameters were set to 5 s for the short-time window and 60 s for the long time window, with "on" and off-detection threshold values of 9.5 and 1.0, respectively. These same parameters were used for both the band-pass filtered and denoised versions of the data to determine the difference in signal detection capabilities.

The 1–15 Hz band-pass filtered and denoised versions of the 19 September 2019 waveform segment are shown in Figure 8a, along with markers denoting at what times a signal detection was made, colored by whether a signal was detected on both versions of the waveform data or solely on the denoised data. On the band-pass filtered waveform data, we were able to detect 21 signals; however, when we ran the STA/LTA detector on the denoised waveform data, we were able to detect a further 10

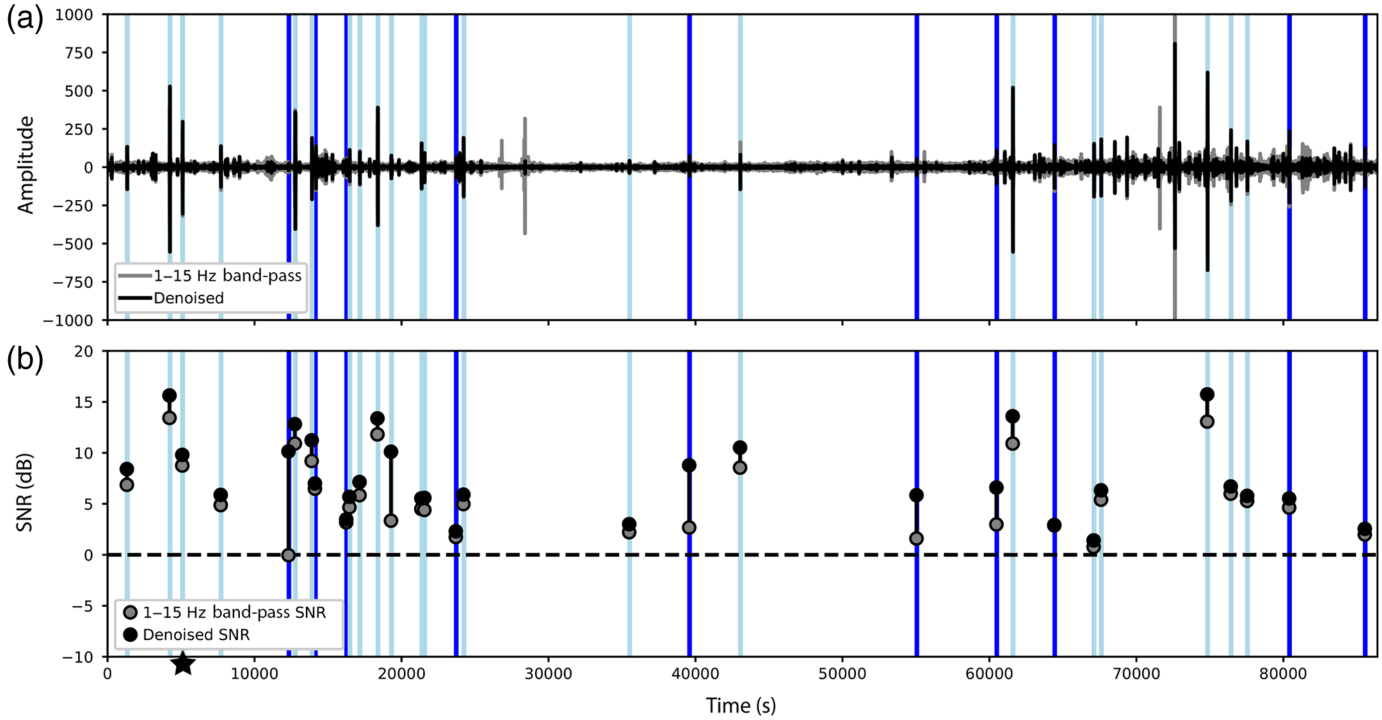

Figure 8. (a) Plot of 24-hr-long 1–15 Hz band-pass filtered and denoised waveform data (see legend) from channel HHZ of station BRPU from 19 September 2019. Signal detections are indicated with vertical lines colored by whether the events were detected only on the denoised data (blue) or on both the denoised and band-pass filtered waveform data (light blue). (b) Comparison plot showing the difference in SNR values calculated using the denoised and band-pass filtered waveform data (see legend) for all signals detected in the 24 hr period. The star on the time axis at about 5000 s indicates the timing of the detection associated with the event discussed in Figure 9. The color version of this figure is available only in the electronic edition.

signals in addition to the 21 previously detected. In Figure 8b, we show a comparison of the SNR values. Across the 24 hr segment, the SNR values for the denoised data are higher than those for the data processed using the 1–15 Hz band-pass. This improvement in SNRs is the driving factor for the increased number of signal detections.

We have observed that the MWCNN denoising model can improve the signal detection capabilities of individual station components through the improvement of the SNR values of detected signals. What we can also observe is that events are more easily detected when we examine the collective denoising results from a group of stations. Figure 9a shows a record section of the HHZ channel recordings from a set of stations located within 200 km of an **M** 2.0 event the origin time of which is recorded as 01:10:17.000 UTC on 19 September 2019 and the epicenter location of which (39.402°N, 110.3°W) is only ~39 km away from station BRPU. This event is marked by the star on the *x* axis in Figure 8b, showing that it was detectable at station BRPU on both the filtered and denoised waveform data. When examining the record section made up of raw waveform data, we can only observe the phase arrival times on the trio of stations closest to the event, with the rest of the local to near-regional distance stations being too noisy to properly pick up on. Band-pass filtering the waveform data (Fig. 9b) leads to improved signal detectability across the recording stations. However, the prephase arrival noise energy observed on farther-distance stations (>100 km) remains high, which could lead to difficulties capturing accurate phase arrival times. Figure 9c, on the other hand, shows the denoised versions of the waveform data from the same set of stations. On this denoised record section, we can not only more clearly observe the phase arrivals on the trio of closest stations, but we can observe the phase arrivals on the more distant stations due to lower prephase noise levels and more than triple the number of *P*-phase detections we use to construct this event. We show that using the MWCNN denoising model, we can both improve the SNR values of previously detectable events, making their phase arrival times clearer, and yielding greater numbers of phase detections on more distant stations to detect and build more events with higher numbers of associated phases.

CONCLUSIONS

In this study, we have presented a CNN model architecture that incorporates wavelet-transform functions to improve seismic waveform denoising performance, described the training process of said model, and evaluated its performance on constructed test and collected real-world data sets from UUSS network stations against other seismic denoising methods. The trained MWCNN denoising model has demonstrated the ability to more accurately separate the designated signal and noise

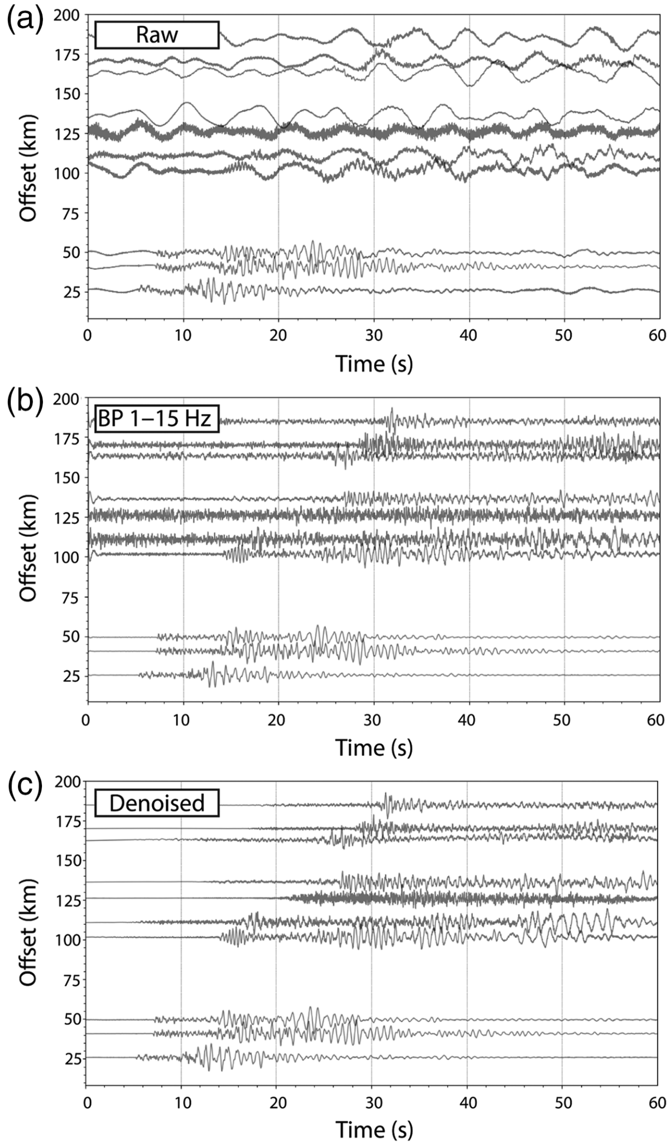

Figure 9. (a) Record section for an **M** 2.0 event with an origin time of 19 September 2019 01:10:17.000 UTC showing raw waveform data from UUSS stations within 200 km of the event location. (b) Record section plot like in panel (a) after filtering the waveforms using a 1–15 Hz band-pass filter. (c) Record section plot like in panel (a) after denoising the waveforms. Observe the difference in pre-*P* noise level between the filtered seismograms and the denoised waveforms.

components of waveforms recorded by the UUSS network stations.

Overall, we found that the MWCNN denoising model was able to outperform the previously developed FCNN denoising model when trained and evaluated on the same sets of UUSS data. This suggests that the information retention improvements related to the incorporation of wavelet-transform functions into the model architecture can lead to improved learning ability in CNN architectures. Evaluation of test data sets showed that the MWCNN model was able to accurately recover the waveform shape (CC) and amplitudes (SDR) of the original signal waveforms. Evaluation of the MWCNN model on real-world event detections showed that the denoising model was able to improve SNR values by an average of 18.4 dB when compared with the raw waveform data across all three station components. In addition, we found that the MWCNN denoising model did not significantly alter the detection times for those previously detected events. Using the MWCNN denoising model, we were able to increase the number of events detected during a seismically active 24 hr period while also increasing the number of phases associated with previously detected events.

The model was trained entirely on data collected from the Utah region, and so the question of model transportability to other regions remains. It should be expected that the amount of noise sources, along with their intensity and characteristics, should vary from region to region and from monitoring network to monitoring network. Thus, it is vital to allow the denoising model to train on the types of noise that one should expect to record to achieve optimal denoising results for a specific region of interest. In addition, although we made attempts to broaden the range of SNR values in the training data set, other seismic regions may have completely different ranges of magnitudes for their signals of interest that fall outside the scope of the data set used to train this version of the MWCNN denoising model. Finally, we specifically limited the scope of training and evaluation data sets to the local to near-regional distance scales; thus, the question of the MWCNN model's performance on global waveform data sets remains as well. Answering these questions about the transportability of the MWCNN denoising model remains a key future research goal. In addition, we will continue to explore the applicability of the MWCNN denoising model to other areas of research, such as noise correlation studies, and its potential incorporation into larger seismic processing workflows.

DATA AND RESOURCES

Waveform data were obtained from the Incorporated Research Institutions for Seismology database available at http://www.iris.edu using the network code UU. A deep-learning model architecture was constructed using the Keras python library available at https://github.com/fchollet/keras. All websites were last accessed in August 2023.

DECLARATION OF CONFLICT OF INTERESTS

The authors acknowledge that there are no conflicts of interest recorded.

ACKNOWLEDGMENTS

This Low Yield Nuclear Monitoring (LYNM) research was funded by the National Nuclear Security Administration, Defense Nuclear Nonproliferation Research and Development (NNSA DNN R&D). The authors acknowledge important interdisciplinary collaboration with scientists and engineers from Los Alamos National Laboratories

(LANL), Lawrence Livermore National Laboratories (LLNL), Nevada National Security Sites (NNSS), Pacific Northwest National Laboratories (PNNL), and Sandia National Laboratories (SNL). Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, Limited Liability Company (LLC), a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under Contract Number DE-NA-0003525. The views expressed in the article do not necessarily represent the views of the U.S. Department of Energy or the United States Government. The authors thank Clinton Koch, Christopher Young, and two anonymous reviewers for their comments and suggestions that helped improve this article.

REFERENCES

Abadi, M., A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.* (2016). TensorFlow: Large-scale machine learning on heterogeneous systems, doi: 10.48550/arXiv.1603.04467.

Bharadwaj, P., L. Demanet, and A. Fournier (2017). Deblending random seismic sources via independent component analysis, *SEG Technical Program Expanded Abstracts*, Society of Exploration Geophysicists, 4898–4902.

Büssow, R. (2007). An algorithm for the continuous Morlet wavelet transform, *Mech. Syst. Sig. Process.* **21,** no. 8, 2970–2979.

Chen, Y., Y. Zhou, W. Chen, S. Zu, W. Huang, and D. Zhang (2017). Empirical low-rank approximation for seismic noise attenuation, *IEEE Trans. Geosci. Remote Sens.* **55,** no. 8, 4696–4711.

Chollet, F. (2015). Keras, available at https://github.com/fchollet/keras (last accessed August 2023).

Han, J., and M. van der Baan (2015). Microseismic and seismic denoising via ensemble empirical mode decomposition and adaptive thresholding, *Geophysics* **80,** KS69–KS80.

Hennenfent, G., and F. J. Herrmann (2006). Seismic denoising with nonuniformly sampled curvelets, *Comput. Sci. Eng.* **8,** no. 3, 16–25.

Herrmann, R. B. (1973). Some aspects of band-pass filtering of surface waves, *Bull. Seismol. Soc. Am.* **63,** no. 2, 663–671.

Huang, W., R. Wang, D. Zhang, Y. Zhou, W. Yang, and Y. Chen (2017). Mathematical morphological filtering for linear noise attenuation of seismic data, *Geophysics* **82,** no. 6, V369–V384.

Kingma, D. P., and J. Ba (2014). Adam: A method for stochastic optimization, available at http://arXiv.org/abs/1412.6980 (last accessed August 2023).

Liu, P., H. Zhang, W. Lian, and W. Zuo (2019). Multi-level wavelet convolutional neural networks, *IEEE Access* **7,** 74,973–74,985.

Liu, Y., S. Fomel, and C. Liu (2015). Signal and noise separation in prestack seismic data using velocity-dependent seislet transform, *Geophysics* **80,** no. 6, WD117–WD128.

Mallat, S. G. (1989). A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pattern Anal. Mach. Intell.* **11,** no. 7, 674–693.

Mousavi, S. M., and C. A. Langston (2016). Hybrid seismic denoising using higher-order statistics and improved wavelet block thresholding, *Bull. Seismol. Soc. Am.* **106,** no. 4, 1380–1393.

Mousavi, S. M., and C. A. Langston (2017). Automatic noise-removal/signal-removal based on general cross-validation thresholding in synchrosqueezed domain and its application on earthquake data, *Geophysics* **82,** no. 4, V211–V227.

Nakajima, H., Y. Takahashi, K. Kondo, and Y. Hisaminato (2018). Monaural source enhancement maximizing source-to-distortion ratio via automatic differentiation, available at http://arXiv.org/abs/1806.05791v1 (last accessed August 2023).

Oropeza, V., and M. Sacchi (2011). Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis, *Geophysics* **76,** no. 3, V25–V32.

Ronneberger, O., P. Fischer, and T. Brox (2015). U-Net: Convolutional networks for biomedical image segmentation, *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 234–241.

Ruikar, S. D., and D. D. Doye (2011). Wavelet based image denoising technique, *Int. J. Adv. Comput. Sci. Appl.* **2,** no. 3, 49–53.

The University of Utah (1962). University of Utah regional seismic network, *International Federation of Digital Seismograph Networks, Other/Seismic Network*, doi: 10.7914/SN/UU.

Tian, C., M. Zheng, W. Zuo, B. Zhang, Y. Zhang, and D. Zhang (2023). Multi-stage image denoising with the wavelet transform, *Pattern Recognit.* **134,** 109050.

Tibi, R., P. Hammond, R. Brogan, C. J. Young, and K. Koper (2021). Deep learning denoising applied to regional distance seismic data in Utah, *Bull. Seismol. Soc. Am.* **111,** 775–790, doi: 10.1785/0120200292.

Tibi, R., C. J. Young, and R. W. Porritt (2023). Comparative study of the performance of seismic waveform denoising methods using local and near-regional data, *Bull. Seismol. Soc. Am.* **113,** no. 2, 548–561.

Yuan, S., S. Wang, C. Luo, and T. Wang (2018). Inversion-based 3-D seismic denoising for exploring spatial edges and spatio-temporal signal redundancy, *IEEE Geosci. Remote Sens. Lett.* **15,** no. 11, 1682–1686.

Zhu, W., S. M. Mousavi, and G. C. Beroza (2019). Seismic signal denoising and decomposition using deep neural networks, *IEEE Trans. Geosci. Remote Sens.* **57,** no. 11, 9476–9488, doi: 10.1109/TGRS.2019.2926772.

Manuscript received 6 December 2023
Published online 8 April 2024