

# User's Guide to Deep Learning Denoiser

---

Copyright [2022] National Technology & Engineering Solutions of Sandia, LLC (NTESS). Under the terms of Contract DE-NA0003525 with NTESS, there is a non-exclusive license for use of this work by or on behalf of the U.S. Government. Export of this data may require a license from the United States Government.

---

**SAND2022-14205 O**

For questions or comments, please contact Rigo Tibi ([rtibi@sandia.gov](mailto:rtibi@sandia.gov))

## Prerequisite:

To run the CNN denoiser software, the following packages are required:

- Python Version 3.7.3 or newer
- TensorFlow Version 2.5.0 or newer
- Keras Version 2.5.0 or newer

## Components:

The CNN denoiser software consists of the following components:

- (a) CNN model ***cnn\_denoiser\_3comp.h5***
- (b) Python script ***ConvNet\_Input\_Creator\_Norm.py***
- (c) Python script ***recover.py***
- (d) Python script ***wfmplot.py***

- The model ***cnn\_denoiser\_3comp.h5***, as the extension indicates, is in Hierarchical Data Format.
- The script ***ConvNet\_Input\_Creator\_Norm.py*** is used to generate the input data for the denoiser. Inputs to the script are two NumPy files: one containing raw data for 60-second long waveforms sampled at 100 Hz, and the other containing the metadata corresponding to each of the waveforms.

Each row in the file containing the raw waveform data consists of the waveform ID in the first column, followed by the point-by-point amplitude values for that waveform. Each row in the metadata file consists of the waveform ID in the first column, followed by station code, channel, sample rate in Hz, phase (*P*, *Pn*, *Lg*, etc.), start time in epoch, arrival time in epoch, number of points. The start and arrival times are used to select the correct windows for the SNR estimation.

Outputs of the script are two separate NumPy files, one containing the normalized real parts (***stft\_raw\_data\_real\_norm.npy***), and the other the normalized imaginary parts (***stft\_raw\_data\_img\_norm.npy***) of the Short-Time Fourier Transforms (STFTs) for all the waveforms contained in the input file.

- The script ***recover.py*** uses the STFTs files generated as described above to separate the signal (denoised waveform) from the noise waveform for each raw seismogram in the dataset. Inputs to the script are the file containing the raw data, as described above and the two files containing the transforms (***stft\_raw\_data\_real\_norm.npy*** and ***stft\_raw\_data\_img\_norm.npy***). Outputs are NumPy files containing the denoised waveforms and the noise waveforms, respectively. Each row in each output file consists of the waveform ID in the first column, followed by the point-by-point amplitude values.
- The script ***wfmpplot.py*** generates a figure showing both the raw seismogram and the corresponding denoised waveform for a specified waveform ID. The generated figure is saved in the subfolder **plots** that is automatically created by the script if it does not exist.

### **Notes:**

The CNN model was built using signal data recorded by the IMS station MKAR. For that reason, the model might not perform well for other stations.

### **References:**

- 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.
- Koch, C., R. Tibi, and C. Young (2022). The 3C Convolutional Neural Network Denoising Model for Station MKAR: Training and Evaluation, *Technical Report*, SAND2022-