Signal Denoising Methods

Adrita Khan

October 9, 2024

Abstract

This document presents various algorithms and methods to test for signal denoising, ranging from classical techniques like Fourier Transform to more modern approaches such as Deep Learning-based Autoencoders.

Contents

1 Introduction 3

2 Wavelet Transform Denoising 3

3 Kalman Filtering 3

4 Wiener Filter 3

5 Empirical Mode Decomposition (EMD) 3

6 Savitzky-Golay Filter 4

7 Non-Local Means Denoising 4

8 Principal Component Analysis (PCA) 4

9 Total Variation Denoising (TVD) 4

10 Non-Stationary Noise Filtering (NSNF) 4

11 Short-Time Fourier Transform (STFT) 5

12 Blind Source Separation (BSS) using ICA 5

13 Spectral Subtraction **5**

14 Autoencoders (Deep Learning) **5**

15 Further Considerations **5**

1 Introduction

Here are several algorithms and methods one can test for signal denoising in addition to Classical Fourier Transform (CFT), FFT, and Deep Learning:

2 Wavelet Transform Denoising

Method: The wavelet transform decomposes a signal into different scales (frequency bands) and enables the isolation of noise from the original signal.
Approach: By thresholding the wavelet coefficients (e.g., soft or hard thresholding), noise can be suppressed while preserving signal characteristics.
Advantage: Better at preserving non-stationary signals, and more effective for signals with transient features than Fourier-based methods.
Tools: PyWavelets, MATLAB's wavelet toolbox.

3 Kalman Filtering

Method: Kalman filters are recursive estimators that predict the current state of the signal based on a dynamic model.
Approach: Given a model of the signal and noise, the Kalman filter continuously updates its estimate as new data becomes available, filtering out the noise.
Advantage: Effective for real-time, linear, and time-varying systems.
Tools: filterpy (Python), MATLAB.

4 Wiener Filter

Method: The Wiener filter minimizes the mean square error between the estimated signal and the true signal by applying an optimal linear filter.
Approach: Requires knowledge of the power spectral density of both the signal and noise.
Advantage: Very effective when the noise characteristics are known and stationary.
Tools: scipy.signal.wiener.

5 Empirical Mode Decomposition (EMD)

Method: EMD decomposes the signal into a finite number of intrinsic mode functions (IMFs) that represent oscillatory modes in the signal.
Approach: Noise can be removed by selectively reconstructing the signal using relevant

IMFs.
Advantage: Useful for non-stationary and non-linear signals.
Tools: PyEMD.

6 Savitzky-Golay Filter

Method: A digital filter that fits successive polynomial segments to the signal and smooths it while preserving features like peaks.
Advantage: Ideal for preserving signal trends, especially in signals with high-frequency noise.
Tools: scipy.signal.savgol_filter.

7 Non-Local Means Denoising

Method: A patch-based denoising technique that removes noise by averaging similar patches in a signal, leveraging redundancy.
Advantage: Particularly useful for signals with repetitive structures.
Tools: scikit-image (Python), MATLAB.

8 Principal Component Analysis (PCA)

Method: PCA projects the data onto a lower-dimensional space by finding the principal components.
Advantage: Noise can be removed by reconstructing the signal from only the principal components that capture the most variance.
Tools: scikit-learn, numpy.

9 Total Variation Denoising (TVD)

Method: A regularization technique that reduces noise while preserving edges and sharp features in the signal by minimizing the total variation.
Tools: scikit-image.

10 Non-Stationary Noise Filtering (NSNF)

Method: Filters noise in non-stationary signals using adaptive filters or advanced statistical models that update with time.

Tools: filterpy, custom algorithms.

11 Short-Time Fourier Transform (STFT)

Method: STFT applies the Fourier Transform over short, overlapping windows of the signal.
Advantage: Suitable for signals where the frequency content changes over time.
Tools: scipy.signal.stft.

12 Blind Source Separation (BSS) using ICA

Method: ICA decomposes the observed signal into statistically independent components, separating noise from the signal if they are independent.
Tools: scikit-learn, MNE-Python.

13 Spectral Subtraction

Method: Estimates noise from silent segments of the signal and subtracts it from the noisy signal in the spectral domain.
Tools: Custom numpy implementations.

14 Autoencoders (Deep Learning)

Method: Autoencoders are unsupervised neural networks that learn to map noisy inputs to a clean reconstruction of the signal.
Tools: TensorFlow, PyTorch, Keras.

15 Further Considerations

Hybrid Approaches: Combining different methods (e.g., FFT + Wavelet Transform, or FFT preprocessing + Deep Learning) can often improve results by leveraging the strengths of multiple techniques.
Regularization Methods: One can also experiment with regularization techniques in neural networks or statistical filters to prevent overfitting when denoising.