1 Noise Handling and Denoising

This section describes the noise addition and classical signal-processing denoising techniques used in the *Robust Speech Commands* project.

Noise handling is used primarily for:

- evaluating model robustness under additive noise
- studying the impact of classical denoising on downstream keyword spotting

Quantitative results and discussion are provided in the Evaluation Pipeline.

1.1 Noise Addition

Noise is added using samples from the **MUSAN free-sound subset**.

Let:

- $x[n]$ denote a clean speech waveform
- $n[n]$ denote a noise waveform
- $y[n]$ denote the resulting noisy signal

The noisy signal is constructed as:

$$y[n] = x[n] + \alpha\, n[n] \tag{1}$$

1.1.1 SNR-Controlled Scaling

The noise scaling factor α is computed to achieve a desired Signal-to-Noise Ratio (SNR) in decibels:

$$\mathrm{SNR}_{\mathrm{dB}} = 20 \log_{10} \left(\frac{\mathrm{RMS}(x)}{\mathrm{RMS}(\alpha n)} \right) \tag{2}$$

Solving for α yields:

$$\alpha = \frac{\mathrm{RMS}(x)}{\mathrm{RMS}(n) \cdot 10^{\mathrm{SNR}_{\mathrm{dB}}/20}} \tag{3}$$

where the RMS energy is defined as:

$$\mathrm{RMS}(x) = \sqrt{\frac{1}{T} \sum_{n=1}^{T} x[n]^2} \tag{4}$$

This formulation ensures that the added noise achieves the exact target SNR.

1.1.2 Implementation Details

- Noise is added **sample-wise (1-to-1)**: the i-th noise segment is added to the i-th test waveform.
- Noise segments are:
  - trimmed or padded to exactly 1 second
  - shuffled using a fixed random seed
  - selected to match the test set size
- Supported SNR levels are configurable (default: $[0, 5, 10, 20]$ dB).

1.1.3 Silence Handling (RMS Threshold)

If the RMS energy of a clean waveform is below a threshold τ:

$$\text{RMS}(x) < \tau \tag{5}$$

noise is not added and the clean waveform is left unchanged. This prevents amplification of noise in silence segments.

1.1.4 Clipping

After noise addition, the waveform is optionally clipped:

$$y[n] \leftarrow \text{clip}(y[n], -1.0, 1.0) \tag{6}$$

to prevent numerical overflow and invalid audio values.

1.2 Denoising Filters

Two classical denoising techniques are implemented and evaluated. Denoising is applied **only to the noisy test set**, prior to feature extraction.

1.2.1 Wiener Filtering

Goal: Estimate the clean speech signal by minimizing the mean squared error in the frequency domain.

Let:

- $Y(f, t)$ be the STFT of the noisy signal
- $S(f, t)$ be the clean speech STFT
- $N(f, t)$ be the noise STFT

The Wiener estimate is given by:

$$\hat{S}(f,t) = G(f,t)\, Y(f,t) \tag{7}$$

where the Wiener gain is:

$$G(f,t) = \frac{\xi(f,t)}{1+\xi(f,t)} \tag{8}$$

and the **a-posteriori SNR** is:

$$\xi(f,t) = \frac{\max\left(|Y(f,t)|^2 - \Phi_N(f),\, 0\right)}{\Phi_N(f) + \varepsilon} \tag{9}$$

Here, $\Phi_N(f)$ denotes the noise Power Spectral Density (PSD).

1.2.2 Spectral Subtraction

Goal: Subtract an estimate of the noise magnitude spectrum from the noisy spectrum.

Let:

- $|Y(f,t)|$ be the noisy magnitude spectrum
- $|N(f)|$ be the estimated noise magnitude spectrum

The enhanced magnitude spectrum is:

$$|\hat{S}(f,t)| = \max\left(|Y(f,t)| - \alpha|N(f)|,\, \epsilon\right) \tag{10}$$

where α is an oversubtraction factor and ϵ is a spectral floor. The phase is preserved:

$$\hat{S}(f,t) = |\hat{S}(f,t)|e^{j\angle Y(f,t)} \tag{11}$$

The time-domain signal is reconstructed using the inverse STFT.

1.3 Noise PSD Estimation

Accurate noise PSD estimation is critical for both filters. Three estimation strategies are supported.

1.3.1 Known Noise PSD

If a clean noise-only waveform is available:

$$\Phi_N(f) = \frac{1}{T}\sum_t |N(f,t)|^2 \tag{12}$$

This provides the most reliable PSD estimate.

1.3.2 Energy-Based VAD (Blind Estimation)

This method assumes that low-energy frames are noise-dominated.

Frame energy is computed as:

$$E(t) = \frac{1}{F} \sum_{f} |Y(f,t)|^2 \tag{13}$$

Frames below a percentile-based threshold are selected and averaged:

$$\Phi_N(f) = \mathbb{E}_{t \in \mathcal{N}} \left[|Y(f,t)|^2 \right] \tag{14}$$

1.3.3 Minimum Statistics

This approach assumes noise corresponds to low-percentile power values over time:

$$\Phi_N(f) = \text{percentile}_p \left(|Y(f,t)|^2 \right) \tag{15}$$

1.4 Remarks on Performance

Although denoising improves perceptual signal quality, both Wiener filtering and spectral subtraction significantly degrade keyword spotting accuracy across all tested SNR levels.