REPORT: SEPARATION OF DRUMS FROM MUSIC SIGNALS

Table of Contents

Overview of the problem and solution

The problem to be solved is the separation of drums (percussive components) from music signals. In order to solve this problem, we observe an anisotropy in percussive and harmonic components. The harmonic component usually has a stable pitch and forms parallel ridges with smooth temporal envelopes on the spectrogram, while the energy of a percussive tone is concentrated in a short time frame, which forms a vertical ridge with a wideband spectral envelope [1].

This anisotropy can be explained by the spectrogram below. The harmonic components are the horizontal lines, while the percussive components are described by the vertical lines.

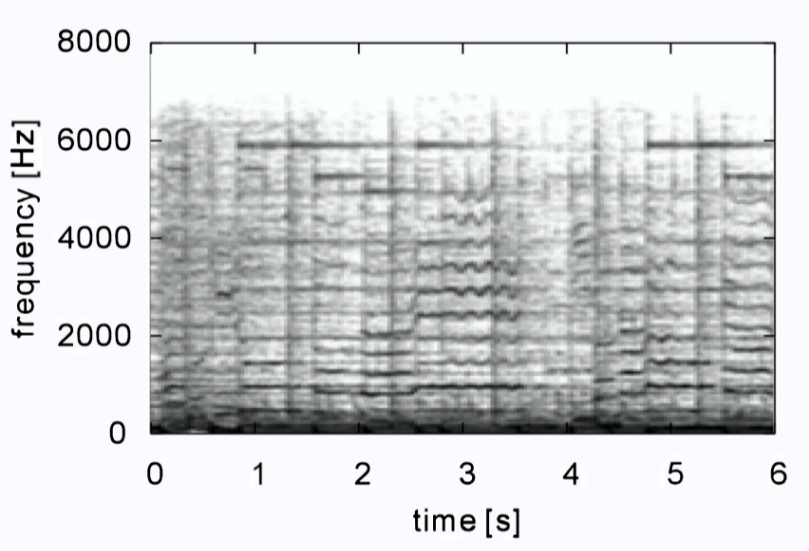

Figure 1. Spectrogram of a popular song [1]

In other words, the harmonic components have a temporal continuity along time (hence horizontal line on the spectrogram), and the percussive components have a temporal continuity along frequency.

In order to utilize this observation, we attempt to find the spectrograms that will maximize the anisotropy. This can be done in an iterative manner to maximize efficiency [1]. The algorithm is derived from the research paper in reference [1].

Algorithm implementation

- Firstly, make sure the audio sample is in .wav format.
- Read the .wav file to a NumPy array & obtain its sampling rate and perform Short Time Fourier Transform on the signal. This is done by using the Librosa library v0.7.1 in Python.
- A range-compressed version of the power spectrogram derived from the sequence obtained in the previous step is then calculated. The range

compression coefficient γ $(0 < \gamma \leq 1)$ is determined in the next section Evaluation.

- The power spectrogram for harmonic (H) and percussive (P) components of the signal are initialized, each equals half of the signal's power spectrogram.
- Some constants are then initialized:
    - k_{max}: number of iterations which the algorithm will search for the optimal separated power spectrograms. The optimal value for this constant is determined in the next section Evaluation.
    - σ_H and σ_P: control the weights of vertical and horizontal smoothness. We set both of them to 1.
- The algorithm then goes into a while loop, repeatedly updating the values of H and P with the changing value of Δ. The Δ value is found iteratively by the auxiliary function *find_delta*. The values of H and P are then changed according to equations (21) and (22) from [1].
- Once the while loop is completed, the outputs in terms of H and P are binarized to be either 0 or the value from the range-compressed version of the power spectrogram calculated earlier. Thus, effectively separating the harmonic and percussive components in the initial signal.
- The Inverse Short Time Fourier Transform is then performed, and the components are converted back into the WAV signals: "*H.wav*" for harmonic-only audio and "*P.wav*" for percussion-only audio.

Evaluation

Three different measurements of evaluation were used: signal-to-noise ratio, spectrogram and auditory evaluation on two audio samples provided by the course: *police03short.wav* and *project_test1.wav*.

Signal-to-noise ratio

The signal-to-noise ratio is defined as

$$SNR = 10 \log_{10}\left(\frac{\sum_t s(t)^2}{\sum_t e(t)^2}\right) \quad (1)$$

where s(t) is the original signal and e(t) is the original signal minus the separated signal. A ratio higher than 1 (greater than 0 dB) indicates more signal than noise. [2]

Signal-to-noise ratios were calculated to observe the effects of γ (range compression coefficient) and the number of search iterations on the quality. The observations on *police03short.wav* are illustrated in the following 2 plots.

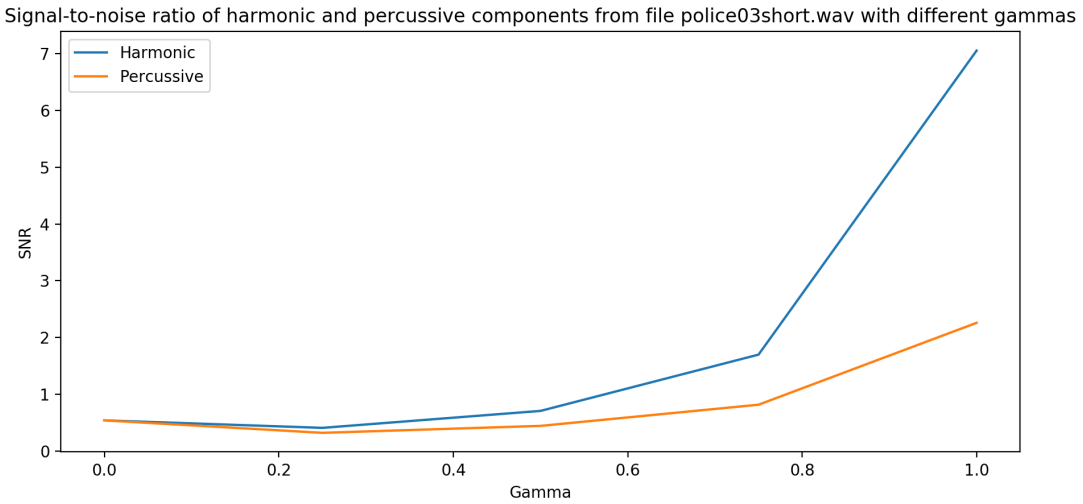

Figure 2. Different γ values' effects on the separation signal-to-noise

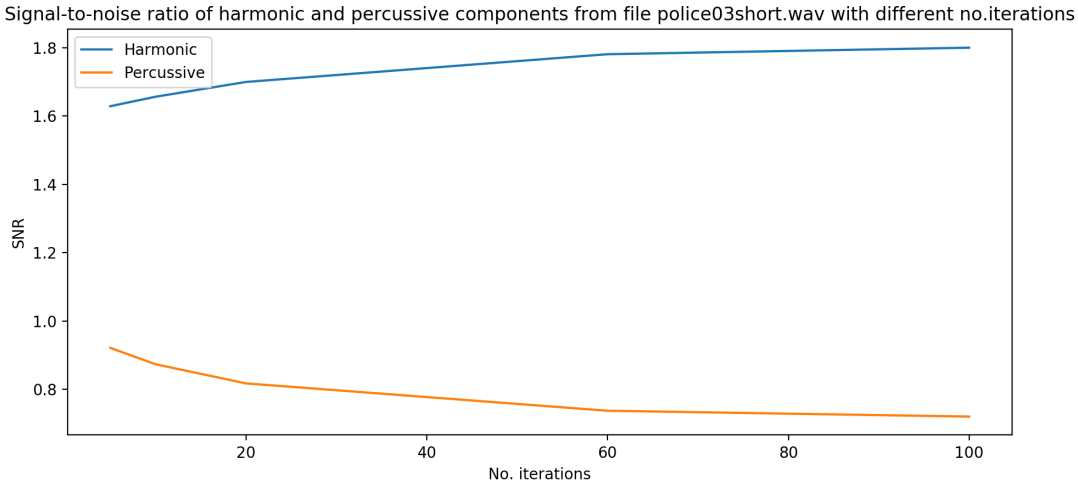

Figure 3. Different numbers of iterations' effects on the separation signal-to-noise

From these plots we can see that setting the range compression value to 1 (the highest possible value) will yield the highest SNR, minimizing the effect of noise. Changing the number of iterations, however, does not significantly affect the SNR. Similar observations are also observed in *project_test1.wav*. Therefore, we recommend setting γ to 1 and the number of iterations to at least 20 but increasing it won't yield noticeable improvement.

Spectrogram

Similar to SNR, spectrograms of the separated audio samples were drawn to examine the effects of γ and number of iterations in separation quality. Below is the spectrogram of *police03short.wav*

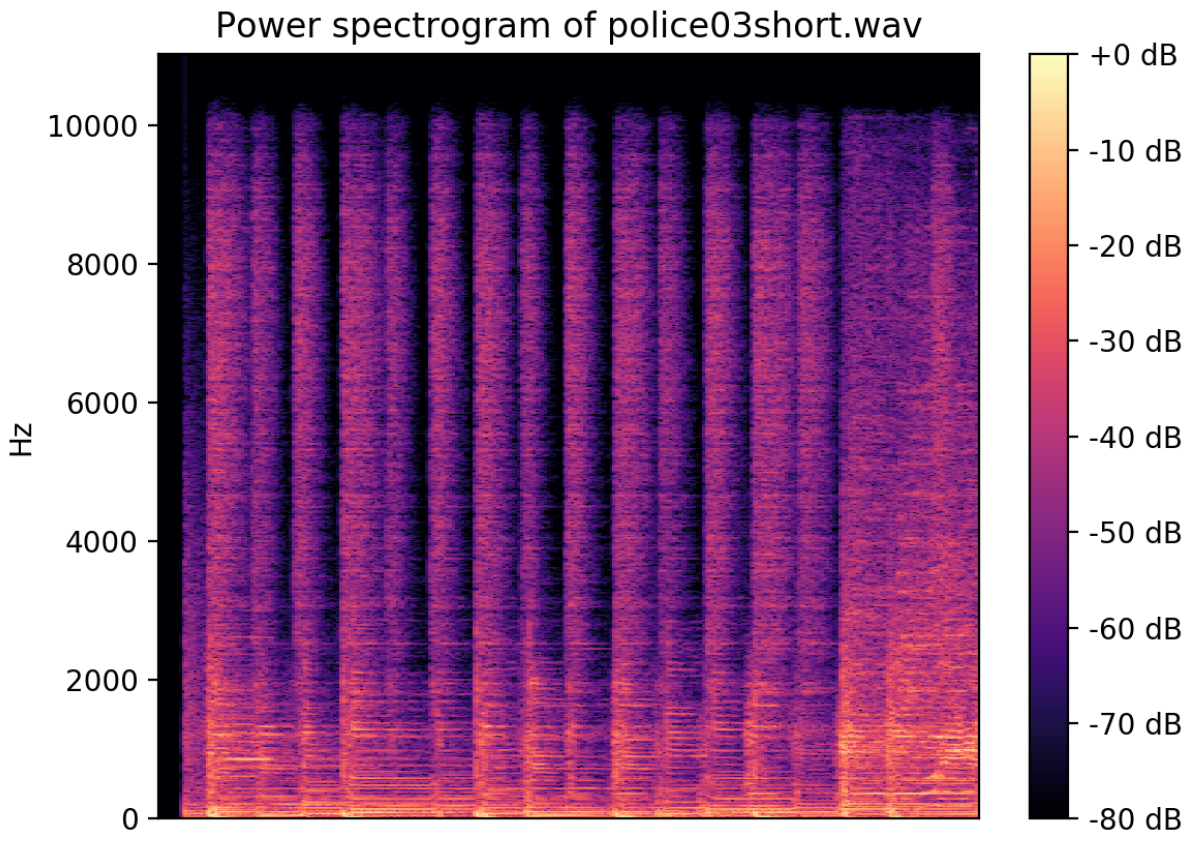

Figure 4. Spectrogram of police03short.wav

The spectrograms below describe the separated components the signal.

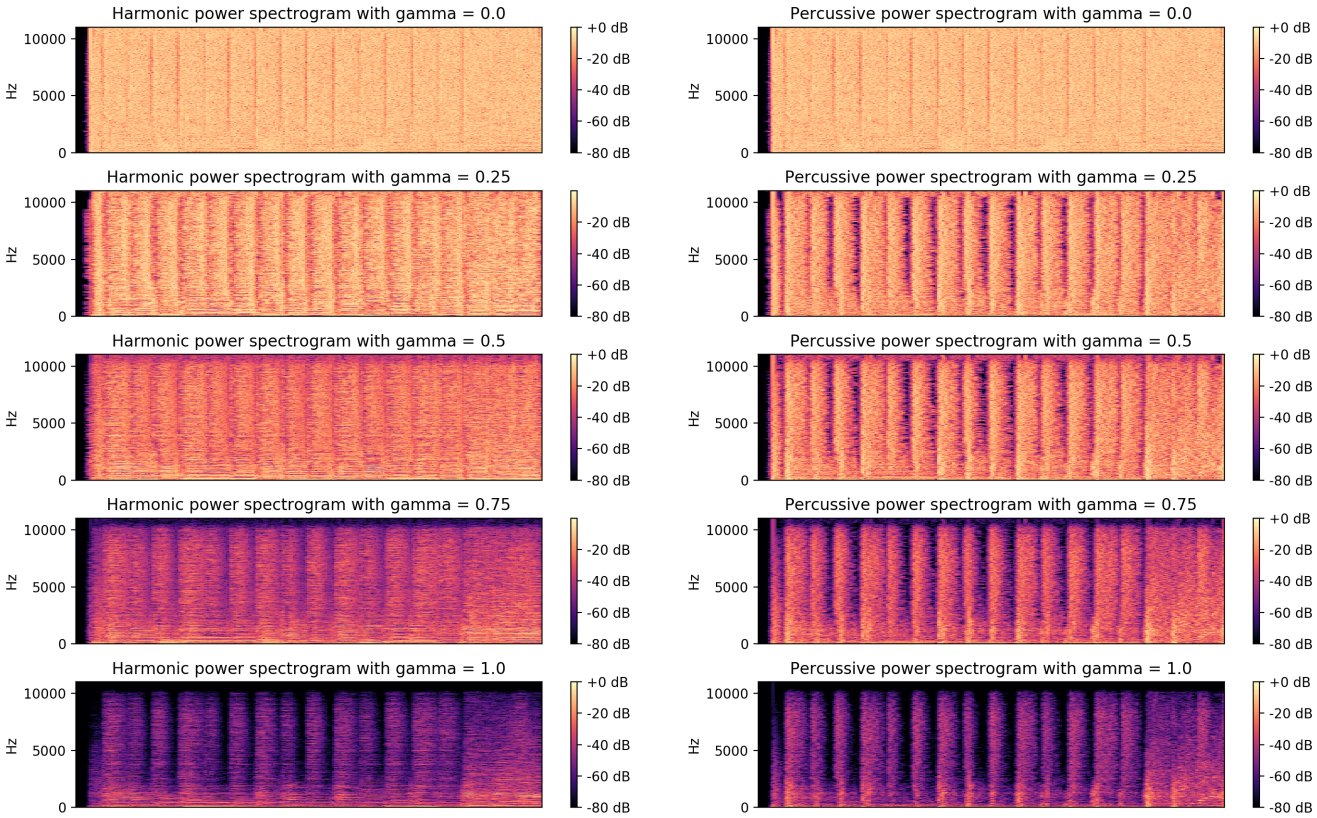

Figure 5. Spectrograms of police03short.wav describing the effect of different γ values

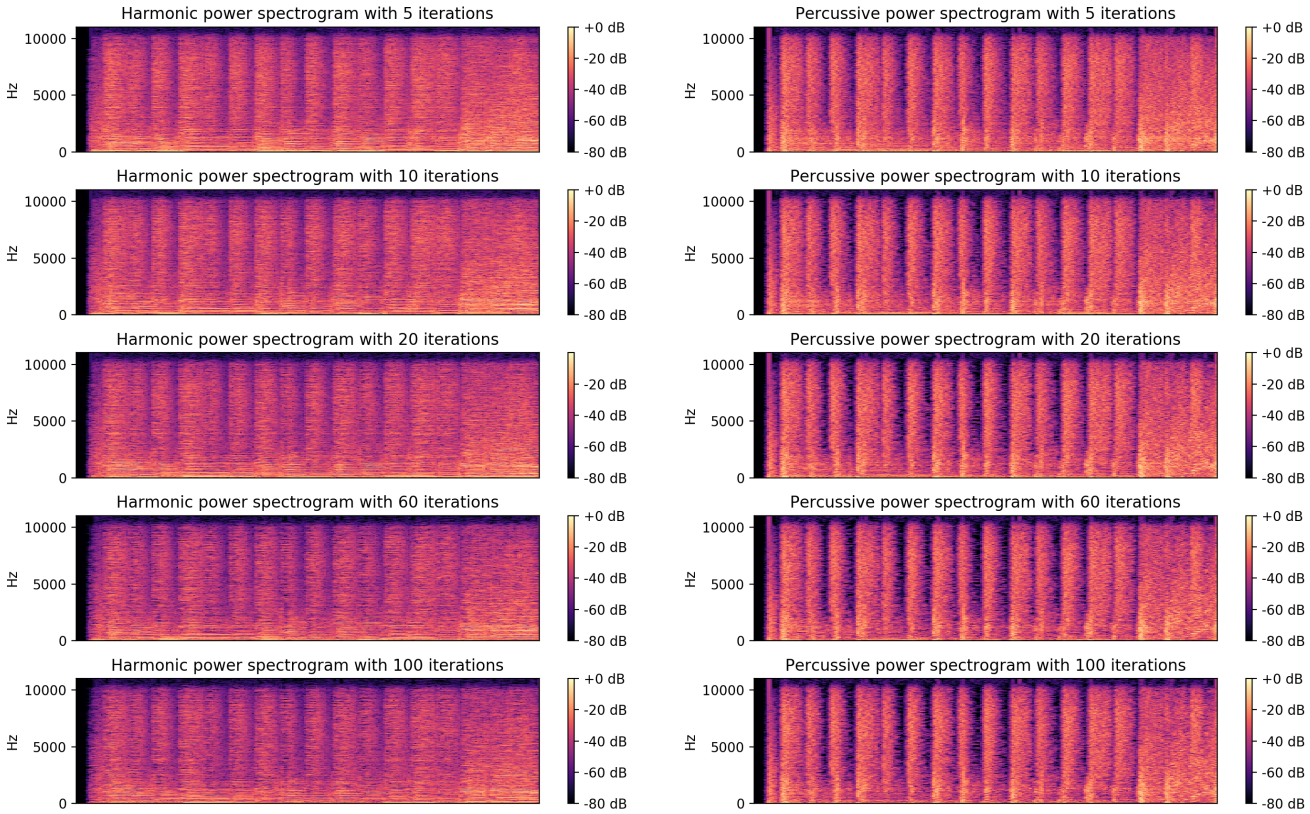

Figure 6. Spectrograms of police03short.wav describing the effect of different no. iterations

The first spectrogram has confirmed the effect of range compression coefficient γ: the lower it is, the more noisy the results are. Unsurprisingly, little difference is shown with a variety of numbers of iterations. To better examine the separation quality, the separated spectrograms of *police03short.wav* from *H.wav* and *P.wav* are shown below, with recommended values (γ = 1, number of iterations = 20).

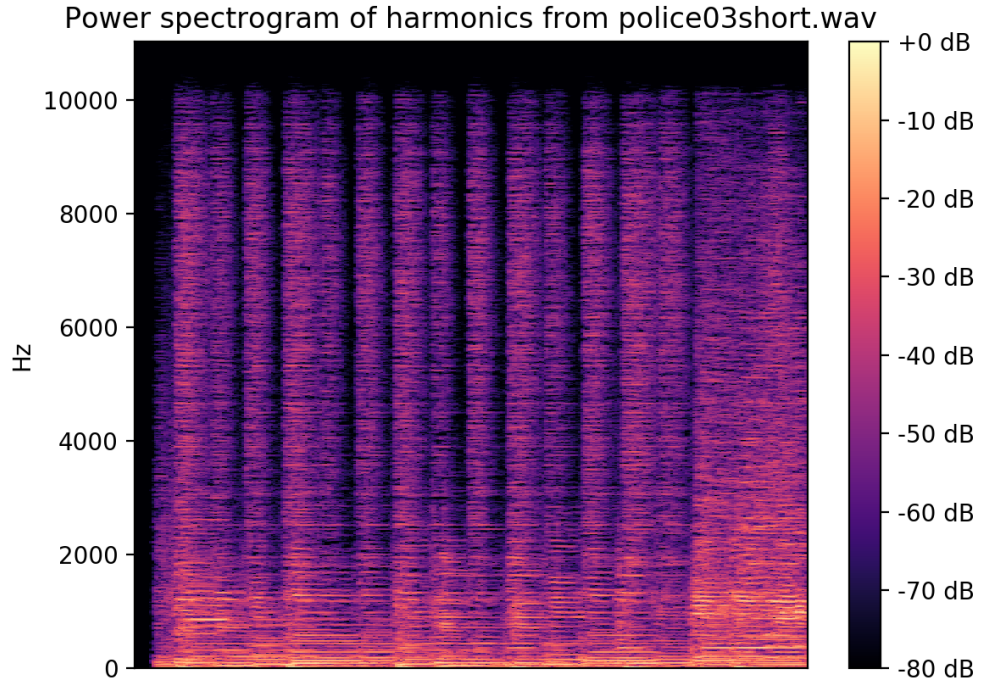

Figure 7. Power spectrogram of harmonics in police03short.wav

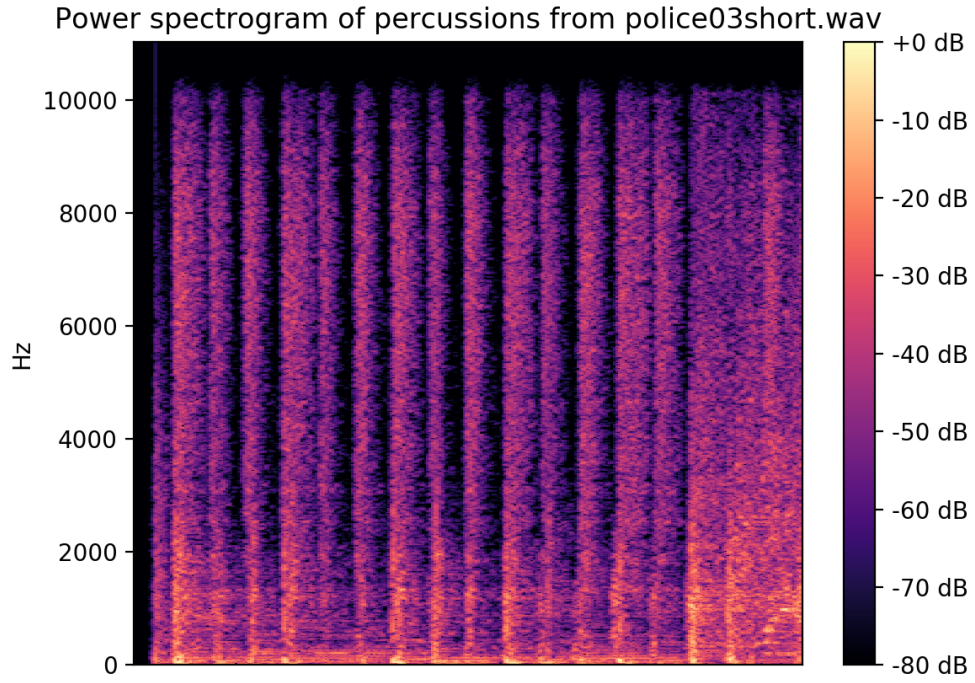

Figure 8. Power spectrogram of percussions in police03short.wav

Comparing these to the original power spectrum, and to each other, we can clearly see the harmonics (horizontal lines) in the upper spectrogram as well as the percussions (straight, vertical lines) in the lower one. This indicates a very positive performance of the algorithm on this audio sample.

However, the same positive result cannot be seen from *project_test1.wav*. While we can see the differences between the two separated power spectrograms, appearances of horizontal and vertical lines are presented in both graphs, indicating that the algorithm does not clearly separate the music signal.

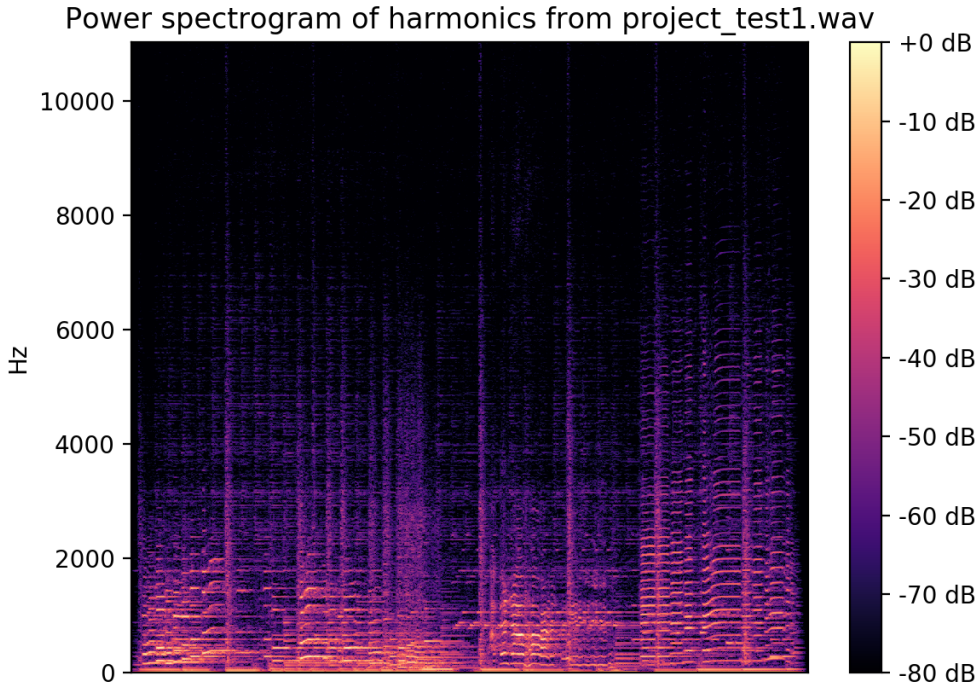

Figure 9. Power spectrogram of harmonics in project_test1.wav

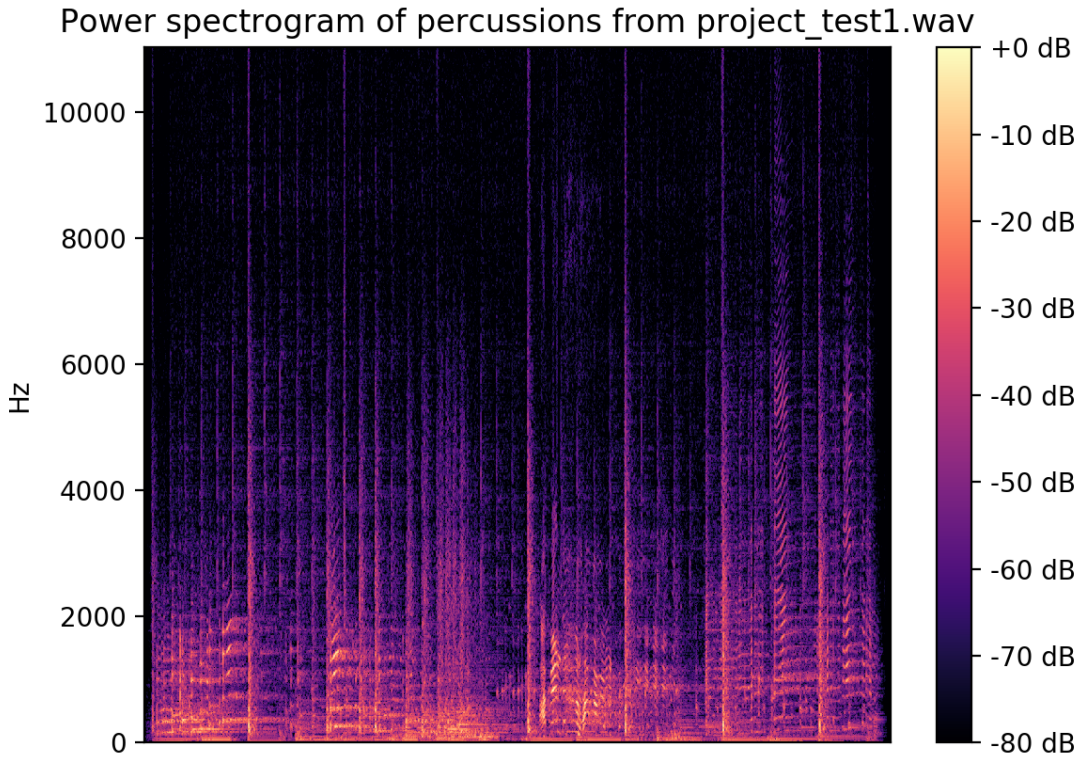

Figure 10. Power spectrogram of percussions in project_test1.wav

Auditory evaluation

Last but not least, we evaluate the separation by listening to the separated audio samples. The evaluation confirms the previous observations:

- Decreasing the range compression coefficient γ increases noise significantly.
- Increasing the number of iterations does not yield observable improvement, even though the number should be set at least at 20 for acceptable quality.
- The separation on *police03short.wav* is much better than *project_test1.wav*, almost no percussion was found in the harmonic-only audio, and very little harmonics can be detected in the percussion-only audio. For *project_test1.wav*, however, while we can clearly hear the separation, some drums can still be heard in the harmonic-only audio, and some singing can be heard in percussion-only audio.

Usage for different types of audio

An explanation for the difference in quality between the two sample test audios is that since *project_test1.wav* is a music sample with singing and certain instruments which have time-varying pitch. This so-called "attack of the pitched tone" has a tendency to belong to percussion-only audio [1], since harmonic-only audio will contain time-continuous frequency components.

On the other hand, *police03short.wav* contains almost only pitched instruments and drums, which will be correctly separated. At the end of *police03short.wav* there

exists some singing, and this singing can be heard from both separated audio samples, therefore further consolidates our explanation.

In conclusion, we believe this algorithm can be used to efficiently separates drum signals from pitched instruments tracks or drum tracks, but for tracks with singing, the results will not be as accurate.

References

[1] Nobutaka Dno, Kenichi Miyamoto, Jonathan Le Roux, Hirokazu Kameoka, and Shigeki Sagayama "Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram". *Proc*. EUSIPCO, Aug 2008

[2] Wikipedia: Signal-to-noise ratio. Available at: https://en.wikipedia.org/wiki/Signal-to-noise_ratio