INTRODUCTION TO AUDIO PROCESSING PROJECT REPORT: TOPIC 01 - SEPARATION OF A MONAURAL AUDIO SIGNAL INTO HARMONIC/PERCUSSIVE COMPONENTS BY COMPLEMENTARY DIFFUSION ON SPECTROGRAM [1]

Khoa Nguyen - 272580
Hai Luong - 268461
Hung Nguyen - 272585

Abstract— **In this report, we present an implementation and results for COMPLEMENTARY DIFFUSION ON SPECTROGRAM, a simple and fast method to separate a monaural audio signal into harmonic and percussive components, which is much useful for multi-pitch analysis, automatic music transcription, drum detection, modification of music, and so on.**

I. INTRODUCTION

The music signal often consists of two different components: harmonic one and percussive one. On the spectrogram, harmonic signal is represented by the horizontal lines and percussive signal is represented by the vertical lines. The goal of the algorithm is to separate them as well as possible for later process. Note that in this implementation, we only process monaural sound - an audio system in which audio signals are mixed and then routed through a single audio channel.

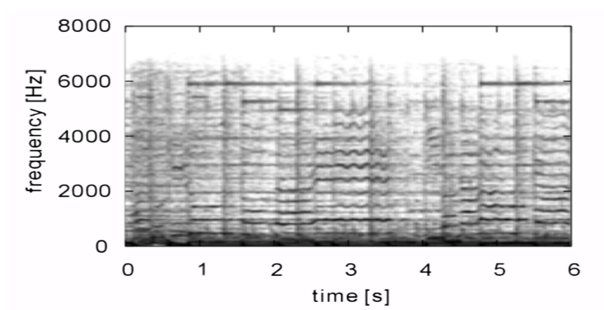

Fig. 1. A spectrogram of a popular music song

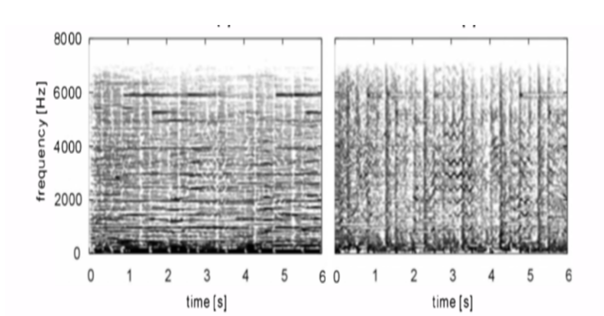

Fig. 2. The spectrograms of the harmonic component (left) and the percussive component (right) of the song after being processed by the algorithm

II. IMPLEMENTATION

The algorithm is implemented in python based on the paper ***Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram*** [1]. The summary of the algorithm without the derivation is shown below in Figure 3.

1. Calculate $F_{h,i}$, the STFT of an input signal $f(t)$.
2. Calculate a range-compressed version of the power spectrogram by
$$W_{h,i} = |F_{h,i}|^{2\gamma} \quad (0 < \gamma \leq 1). \tag{24}$$
3. Set initial values as
$$H^{(0)}_{h,i} = P^{(0)}_{h,i} = \frac{1}{2} W_{h,i}, \tag{25}$$
for all h and i and set $k = 0$.
4. Calculate the update variables $\Delta^{(k)}$ defined as eq. (23).
5. Update $H_{h,i}$ and $P_{h,i}$ as
$$H^{(k+1)}_{h,i} = \min(\max(H^{(k)}_{h,i} + \Delta^{(k)}, 0), W_{h,i}), \tag{26}$$
$$P^{(k+1)}_{h,i} = W_{h,i} - H^{(k+1)}_{w,i}. \tag{27}$$
6. Increment k. If $k < k_{max} - 1$ (k_{max}: the maximum number of iterations), then, go to step 4, else, go to step 7.
7. Binarize the separation result as
$$(H^{(k_{max})}_{h,i}, P^{(k_{max})}_{h,i}) = \begin{cases} (0, W_{h,i}) & (H^{(k_{max}-1)}_{h,i} < P^{(k_{max}-1)}_{h,i}) \\ (W_{h,i}, 0) & (H^{(k_{max}-1)}_{h,i} \geq P^{(k_{max}-1)}_{h,i}) \end{cases} \tag{28}$$
8. Convert $H^{(k_{max})}_{h,i}$ and $P^{(k_{max})}_{h,i}$ into waveforms by
$$h(t) = \text{ISTFT}[(H^{(k_{max})}_{h,i})^{1/2\gamma} e^{j\angle F_{h,i}}], \tag{29}$$
$$p(t) = \text{ISTFT}[(P^{(k_{max})}_{h,i})^{1/2\gamma} e^{j\angle F_{h,i}}], \tag{30}$$
where ISTFT represents the inverse STFT.

Fig. 3. The summarized separation algorithm

The code of the implementation is included in the zip file submitted to the Moodle page of the course.

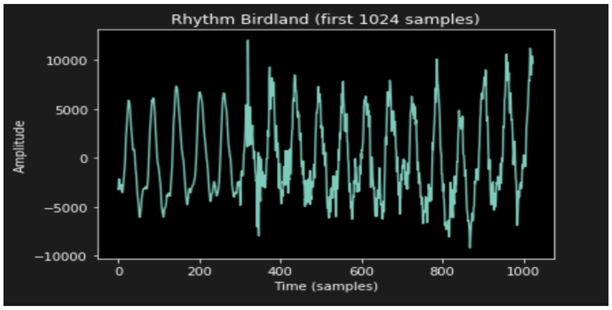

Fig. 4. First 1024 audio samples of rhythm_birdland.wav

III. RESULT

We tried to apply the algorithm on multiple test samples and get good results. For the first audio file *rhythm_birdland.wav*, we get the results (separated Harmonic and Percussive components) with spectrograms plotted in figure 6 and 7.

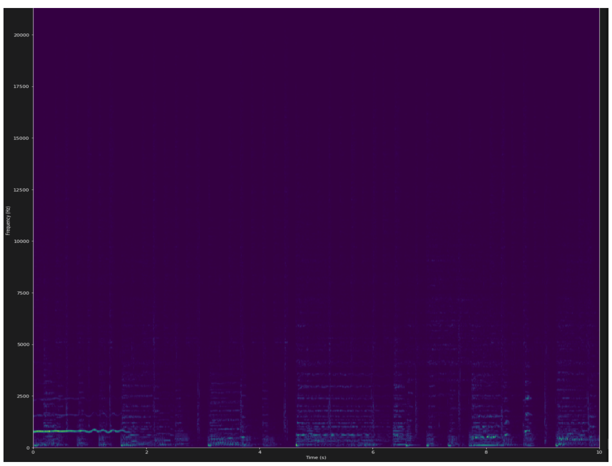

Fig. 5. Spectrogram of rhythm_birdland.wav

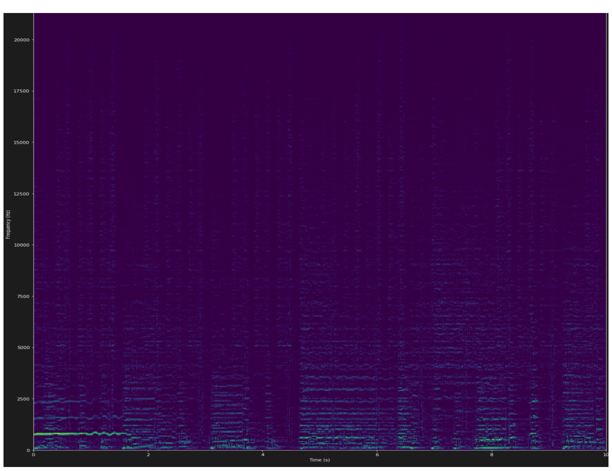

Fig. 6. Spectrogram of Harmonic Component of rhythm_birdland.wav

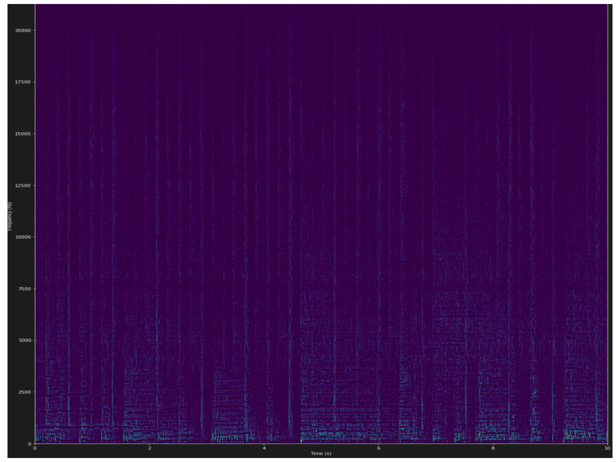

Fig. 7. Spectrogram of Percussive Component of rhythm_birdland.wav

After the algorithm, we check again that if

$$H_{h,i}^{(k_{max})} + P_{h,i}^{(k_{max})} = W_{h,i} \Leftrightarrow \| H_{h,i}^{(k_{max})} + P_{h,i}^{(k_{max})} - W_{h,i} \| = 0 \quad (1)$$

After that, two new audio files will be created in the same folder with the code, one is *Harmonics.wav* and one is *Percussive.wav*. The outputs generated by running the algorithm on different audio inputs with our computers can be accessed as good results on intuition level: the harmonic sound shows the melodies of the input and the percussive sound shows the rhythm of the input.

IV. CONCLUSIONS

Implementing the algorithm gives us a lot of inspiration into the field of Audio Signal Processing, which has big potential for real world applications. Although the implementation of the algorithm is not necessarily difficult, the derivation is very complicated and requires a lot of Mathematical background to understand. Despite of the simplicity of the algorithm, pitched instruments and drums are well separated.

The project is a good learning experience as it gives us a glance into how real world Music Signal are processed.

REFERENCES

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