Assignment 2

Benedith Mulongo, Cathrine Bergh

April 2019

1 Pitch and Intensity Profiles

1.1 Pitch

The following figures are showing the pitch and intensity profiles which were extracted from the raw sound signals. For the pitch profiles, the y-axis shows frequency in log scale (Hz) and the x-axis shows the time given in seconds. The correct time was calculated by multiplying the size of the data with half the window size (since there is a 50% overlap of the windows), as shown in Listing 1 for Melody 1. The length of the recordings were 7.49 seconds for Melody 1, 5.81 seconds for Melody 2 and 5.52 seconds for Melody 3, which agrees with the time the intensity goes to zero in Figures 10, 11 and 12.

Listing 1: Example code for plotting pitch profiles

```
% Load melody
[melody1,fz_1] = audioread('melody_1.wav');

% Extract music features
features1 = GetMusicFeatures(melody1,fz_1,window_size);

% Plot pitch profile
set(gca,'YScale','log')
X = 0.5*window_size*[1:length(features1(1,:))];
plot(X,features1(1,:))
title('Pitch Profile, Melody 1','FontSize',16)
ylabel('Frequency [Hz]')
xlabel('Time [s]')
```

The pitch profiles shown in Figures 4, 6 and 8 show some short, high-frequency pitches when a new note is being hit followed by the expected frequency of the hummed notes heard in the recordings. In the end of the profiles there is some noise when the recording is stopped. Figures 5, 7 and 9 show the pitch profiles zoomed in on the part giving the frequency of the hummed notes. By looking closer at the profile for Melody 1 in Figure 5, we can validate that this is the case. By listening to the recording it can be concluded that the following sequence of notes are hummed:

Perceived Sequence of Notes										
F_3	F_3	$C_3^{\#}$	$C_3^{\#}$	$C_3^{\#}$	$D_3^{\#}$	F_3	F_3	$D_3^{\#}$	$C_3^{\#}$	C_3
Corresponding Frequencies										
171.4	171.4	136.1	136.1	136.1	152.7	171.4	171.4	152.7	136.1	128.4

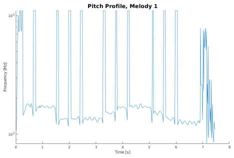

Figure 1: Pitch profile of Melody 1

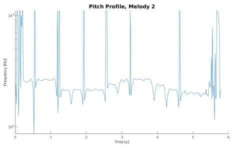

Figure 2: Pitch profile of Melody 2

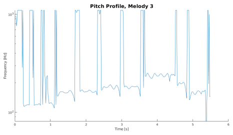

Figure 3: Pitch profile of Melody 3

1.1.1 Comparison with the zoomed pitch

By comparing the frequencies from the perceived notes with the pitch profile in Figure 5 it is clear that the lower stable frequencies correspond to those that make out the melody in the recording. In Figure 7 the same pattern in tone duration and relative frequencies can be observed, but the absolute frequencies are higher, which means the melody has been transposed approximately four notes higher.

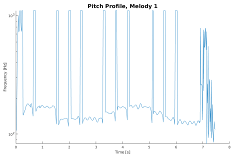

Figure 4: Pitch profile of Melody 1: ”Du Gamla, Du Fria”

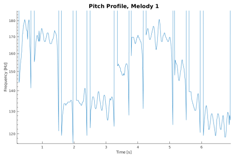

Figure 5: Pitch profile of Melody 1, zoomed

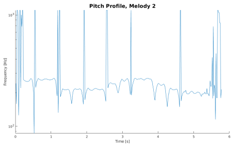

Figure 6: Pitch profile of Melody 2: ”Du Gamla, Du Fria”

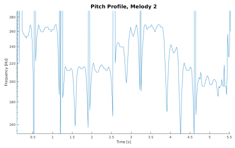

Figure 7: Pitch profile of Melody 2, zoomed

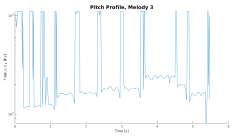

Figure 8: Pitch profile of Melody 3: ”La Marseillaise”

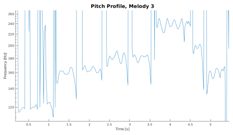

Figure 9: Pitch profile of Melody 3, zoomed

1.1.2 Intensity

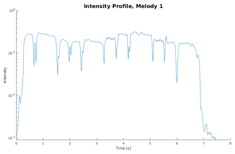

Figure 10: Intensity profile of Melody 1

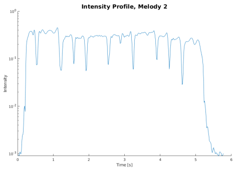

Figure 11: Intensity profile of Melody 2

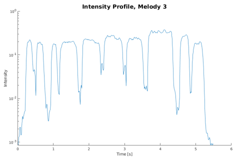

Figure 12: Intensity profile of Melody 3

2 Features extractor description

2.1 Time domain features

2.1.1 Energy

The feature used is the power of the signal defined by :

$$E_i = \frac{1}{W_L} \sum_{n=1}^{W_L} |x_i(n)|^2$$

This features exhibit the concentration of the signal for each frames. The energy varies between high energy states and low energy states.

2.1.2 Loudness

Loudness tries to capture the subjective perception of sound pressure. It can be calculated by the means of Steven’s power law :

$$loudness_i = (\sum_{k=0}^{N-1} |x_i(k)|^2)^{0.67}$$

It measures how loud a specific sound behaves in terms of its amplitude.

2.2 Frequency domain features

The spectral centroid and Spectral spread is a measure of spectral position and shape.

2.2.1 Spectral centroid

The spectral centroid is a measure of the location of the spectrum's center of mass

$$C_i = \frac{\sum_{k=1}^{Wf_l} kX_i(k)}{\sum_{k=1}^{Wf_l} X_i(k)}$$

2.2.2 Spectral spread

Spectral spread computes the measure of the bandwidth, the spread of the signal in the frequency domain:

$$S_i = \sqrt{\frac{\sum_{k=1}^{Wf_l}(k - C_i)X_i(k)}{\sum_{k=1}^{Wf_l} X_i(k)}}$$

2.2.3 Pitch transpose

In order to recognize two melodies performed in different keys as the same melody, the feature extractor was designed to transpose the melodies to a common reference note before making any advanced transformations. Code describing the function that handle these transposed melodies is shown in Listing 2. The idea is very simple. First, the average of the frequencies is computed over all frames, then these averages are aligned to some predetermined reference frequency - in this case 130.84 (C_3), but any frequency could be chosen. This way, sequences with the same melodic pattern get the same averages (or very similar, depending on the amount of noise) and each tone aligns to a similar pitch. A limitation with this approach is that noise will affect the averages, which especially makes it difficult to properly align very short melodies. On the other hand, when melodies become longer the approach is expected to work better and reduce the effect of noise. In addition, this approach can handle the case when pitch jumps more than an octave as the average of all notes is considered. More naive methods considering each octave separately could give strange results in this case.

Listing 2: Function transposing frequencies

```
function feats_out = transpose(feats_in)
    % Compute average frequency
    avg_freq = mean(feats_in(1,:));
    % Align average to C3 (or choose any other tone)
    shift = 130.83/avg_freq;
    feats_in(1,:) = shift*feats_in(1,:);
    feats_out = feats_in;
end
```

2.3 Integration with output distributions class

We have three different output distributions :

- Discrete distribution in class **DiscreteD**
- Gaussian distribution in class **GaussD**
- Mixtures of Gaussian distribution in class **GaussMixD**

We will use the Gaussian distribution, because almost all the features extracted from the audio signal are real values and have not been processed by quantization, therefore the Gaussian distribution is here a sensible choose.

There is not clear idea how to connect the output of the features extraction and the output distributions class, however if we consider the vectors of each feature are Gaussian distributed then we can create a Gaussian Distribution of each them and connect them to each states.

That is what is done. Each features represent a Gaussian distribution and the number of states equal the number of features, they are connected in such way to build a hidden Markov model from which we can sample.

The next chapter shows how the features behave over time, we can clear distinguish similar sound from dissimilar ones.

3 Features evolution over time

3.1 Signal energy

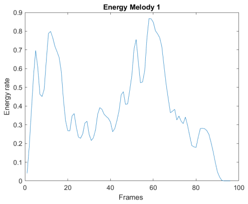

Figure 13: Energy of Melody 1

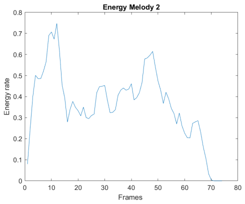

Figure 14: Energy of Melody 2

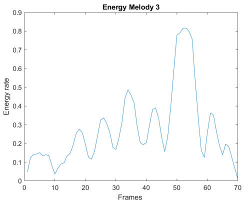

Figure 15: Energy of Melody 3

3.2 Signal loudness

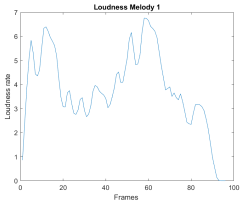

Figure 16: Loudness of Melody 1

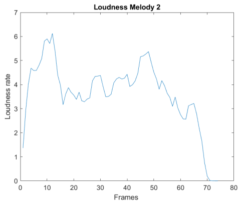

Figure 17: Loudness of Melody 2

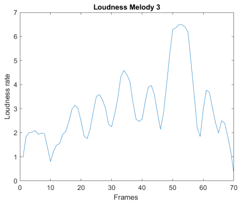

Figure 18: Loudness of Melody 3

3.3 Spectrum centroid

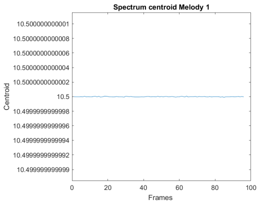

Figure 19: Centroid of Melody 1

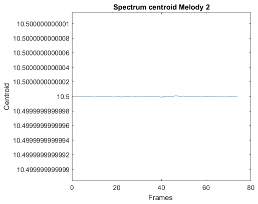

Figure 20: Centroid of Melody 2

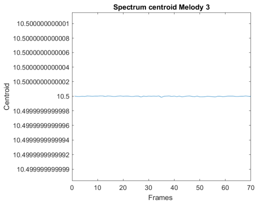

Figure 21: Centroid of Melody 3

3.4 Spectrum spread

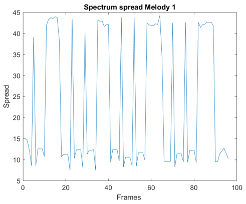

Figure 22: Spread of Melody 1

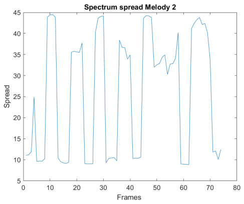

Figure 23: Spread of Melody 2

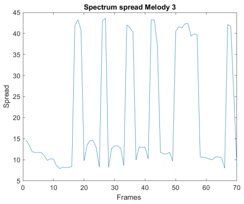

Figure 24: Spread of Melody 3

3.5 FFT magnitude

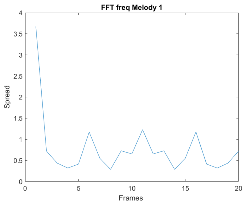

Figure 25: FFT of Melody 1

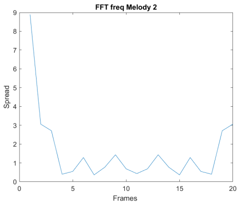

Figure 26: FFT of Melody 2

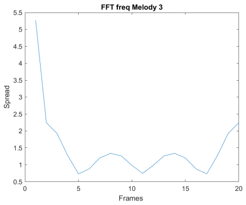

Figure 27: FFT of Melody 3

3.6 FFT angle

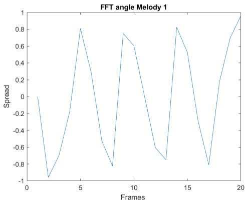

Figure 28: FFT angle of Melody 1

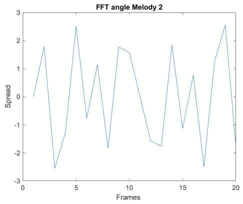

Figure 29: FFT angle of Melody 2

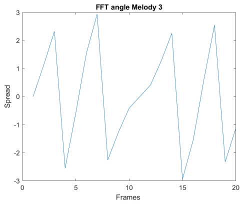

Figure 30: FFT angle of Melody 3

4 Comparison between transposed pitch and original signal

HERE MY THOUGHTS

Transposed signals are shown in Figure **??**, where the average of the signals are at frequency 130 Hz.

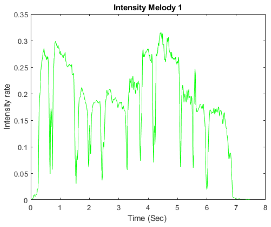

Figure 31: Intensity feature

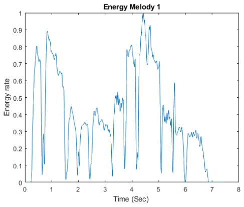

Figure 32: Energy feature for default melody 1

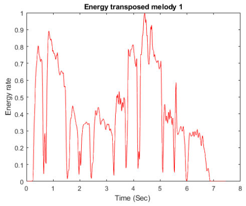

Figure 33: Energy feature for transposed melody 1

The following figures above show how the energy features behave for the transposed intensity melody and the default intensity melody 1. The result shows that the energy feature is invariant and insensible to the transposition with was required. The same is true for the loudness features and the rest as described in the attached Matlab code.

5 Discussion

There is of course some weakness with the set of features implemented and the procedures used in order to compute them. Taking as example the energy feature which gives the level of variability of the sound for each frames, knowing the number of frames and their length then it possible to create two dissimilar sounds where their variability in time is not similar but varies similarly in average in frames, that can influence the ability of the classifier to distinguish them. All the features exhibits this drawbacks, however we have features that rely both on the intensity and the pitch, which hopefully compensate their respective weakness.