Lab 3: Harmonic and tonal analysis

1. Introduction

The goal of this lab is to implement a simple chord and key estimation system based on chroma features.

2. Material

- Software: Matlab/Octave.
- Chroma analysis and synthesis methods:
  - Matlab provided by Dan Ellis at http://labrosa.ee.columbia.edu/matlab/chroma-ansyn/
  - Chroma toolbox by Meinard Müller and Sebastian Ewert http://www.mpi-inf.mpg.de/resources/MIR/chromatoolbox/.
  - MTG approach (HPCP) available as a vamp plugin http://mtg.upf.edu/technologies/hpcp.
- Samples from polyphonic music: MIREX 2005 test set with key annotations (synthesized from MIDI), guitar chords from freesound, any piece from tonal polyphonic music in general.

3. Description

3. A. Chroma feature extraction

Provide a brief description of the procedure for chroma feature extraction proposed by Ellis. Look at the following matlab functions:

```
function Y = chromagram_IF(d,sr,fftlen,nbin,f_ctr,f_sd)
```

```
demo_chroma.m
```

Extract and visualize the instantaneous chroma features (using default parameters) and its average for the whole audio samples, as follows:

```
figure; bar(mean(C'));
aux=(1:12);
set(gca,'xtick',aux);
set(gca,'XTickLabel',{'A';'#';'B';'C';'#';'D';'#';'E';'F';'#';'G';'#';
});
grid
```

Try with at least the following audio samples:

- a monophonic phrase (select one of the samples from Lab2).
- a guitar chord.
- a sample from the MIREX05 key training collection.
- a short audio excerpt that you might consider interesting (e.g. excerpt from a song you like).

You can vary the parameter nbins (default value = 12) to increase the interval resolution and have a better visualization. Comment on the result of the feature extraction process, indicating if the highest bins correspond to the most present pitch classes and if harmonic frequencies are found. Listen and comment on the synthesis results as well.

Provide a brief description of the procedure for HPCP extraction proposed by MTG-UPF, looking at the slides from the Lecture session. Extract chroma features using the vamp plugin through sonic visualizer (or sonic annotator), and compare with previous results.

Provide a brief description of the procedure for chroma feature extraction proposed by Mueller and Ewert, `pitch_to_CRP.m`. Extract chroma features using the chroma toolbox, following the instructions from the web, and compare the average chroma with previous results.

3. B. Chord estimation

Implement a simple chord estimation algorithm by following the next steps:

1. **Compute global or segment chroma vector**. If the considered audio sample corresponds to a single chord, the `chroma` vector is obtained by averaging instantaneous values over the whole audio excerpt (`mean(C')`). If you want to analyze an audio excerpt containing several chords, you should then implement a sliding window method to compute the average of chroma features within 0.5-1 second segments (it might depend on the piece). Window overlap can be set to 25% or 50%, depending on the desired temporal resolution.

2. **Compute correlation with chord profiles**. You should correlate the obtained chroma features with a set of chord profiles. We will only use triad chord profiles:

   Major triad profile: `M = [1 0 0 0 1 0 0 1 0 0 0 0];`
   Minor triad profile: `m = [1 0 0 1 0 0 0 1 0 0 0 0];`

   For each of the 12 possible chord roots, you should ring shift these profiles according to the bin position of the chord root (0 for A, 1 for A#, 2 for B, ...).

```
...
correlation_M = sum( (chroma-mean_chroma) .* (M-mean_M) ) /
(var_chroma*var_M);
correlation_m = sum( (chroma-mean_chroma) .* (m-mean_m) ) /
(var_chroma*var_m);

for i=1:(size-1)
    vector_M = [M(size-(i-1):size) M(1:size-i)];
    vector_m = [m(size-(i-1):size) m(1:size-i)];

    R(1,i+1) = sum( (chroma-mean_chroma) .* (vector_M-mean_M) )
/ (var_chroma*var_M);
    R(2,i+1) = sum( (chroma-mean_chroma) .* (vector_m-mean_m) )
/ (var_chroma*var_m);
end
```

You can also use the `corrcoeff` matlab function. For some selected audio excerpts, visualize the correlation with the 12 major and 12 minor triad profiles.

3. **Select the bin with max correlation** as the estimated chord (e.g. A major).

Evaluate the performance of this simple approach for some isolated chords provided as lab material or downloaded from freesound. Evaluate also the behavior for some chords extracted from polyphonic music. Finally, implement a sliding window method to estimate the chord sequence from a short audio excerpt of your choice containing several chords.

3. B. Key estimation

Use the code previously developed to estimate the key of a given musical piece. You should just replace the triad chord profiles by the tonal profiles proposed by Krumhansl and Kessler's:

Major key profile: `M = [6.35 2.23 3.48 2.33 4.38 4.09 2.52 5.19 2.39 3.66 2.29 2.88];`
Minor key profile: `m = [6.33 2.68 3.52 5.38 2.6 3.53 2.54 4.75 3.98 2.69 3.34 3.17];`

Let's assume a constant key for the analyzed pieces, so that the chroma vector is computed by averaging frame values for the entire piece.

Visualize the correlation values for the 12 major and 12 minor keys for some of the monophonic melodies from Lab2 and the provided MIREX05 examples.

Evaluate the performance of this simple approach for key estimation for some of the examples of the MIREX 2005 test set, and comment on its accuracy. Try also with some other examples from polyphonic pieces you may choose.

Propose ways to improve this simple algorithm. For instance, evaluate with different profiles, e.g. diatonic flat profiles or the profiles proposed in (Serrà et al. 2009).

References

- Müller, M., Ewert, S. Towards timbre-invariant audio features for harmony-based music. IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 3, pp. 649–662, 2010.
- Serrà, J., Gómez, E., Herrera, P., Serra, X. Statistical Analysis of chroma features in Western music predicts human judgements of tonality, Journal of New Music Research, 37(4), pp. 299-309. http://www.tandfonline.com/doi/abs/10.1080/09298210902894085