Multimedia Systems

Assignment A2. Low-Level features and timbre characterization

FEUP 2019/2020 – António Sá Pinto 27/04/2020

1. Goal

The goal of this assignment is to understand, implement and evaluate a simple set of low-level audio descriptors and analyse their distribution over a collection of sounds, which are samples of isolated notes from musical instruments. It is divided in 2 weeks.

2. Resources

Available implementations:

- MIR.EDU Vamp Plugins for feature extraction (https://github.com/justinsalamon/miredu)
- (Matlab/Octave) Additional code for low-level feature extraction. MPEG-7 MATLAB (http://mpeg7.doc.gold.ac.uk/mirror/v1/Matlab-XM/index.html). You can download this code from Moodle.
- (Python) Essentia (Bogdanov et al., 2013)
- (Matlab) MIR Toolbox (Lartillot & Toiviainen, 2007)
- (Matlab) TimbreToolbox (Peeters et al., 2011)

Sound material:

- Samples (isolated notes) from different instruments. ("InstrumentalSounds.zip" - Download from Moodle)
- Sounds from www.freesound.org. You may use this as a complement if you feel the need for more specific sounds (Note: Download only isolated notes)

3. Tasks

Task 1 (week1)

Please review the paper by Peeters (Peeters, 2004)*"A large set of audio features for sound description (similarity and classification) in the cuidado project"*, to make sure that you understand the following descriptors:

Time-domain:
Instantaneous
1. RMS/Energy; 2. Zero Crossing Rate
Global
3. Log-attack time; 4. Temporal centroid; 5. Effective duration

Frequency-domain:
Instantaneous
6. Spectral centroid; 7. Spectral spread; 8. Spectral variation / spectral flux; 9. Spectral flatness

Please pick 2 descriptors by group (one from time-domain and another from frequency-domain), depart from the formula and explain the expected values for a sinusoid and white noise. Calculate these values and comment on your results. If they're not implemented in your software library, please find that implementation in another library (e.g. MPEG7 Matlab) and use it.

Task 2 (week1)

Implement a function to obtain, for a given audio file, the mentioned set of **instantaneous descriptors** (1,2,6-9).
To start, use similar analysis parameters: windowsize = 60 ms, hopsize=10 ms, no zero padding.
Create plots to visualize the extracted instantaneous low-level descriptors and study their evolution for a small set of instrument samples (e.g. percussive, string, wind instrument). Play around with the STFT analysis parameters (windowsize, hopsize, etc.), and try to obtain the best compromise.

Task 3 (week1)

Implement a function to obtain, for a given audio file, the mentioned set of **global descriptors** (3,4,5), as well as statistics of the previous **instantaneous** descriptors (mean, standard deviation, min, max).
Study the values of these descriptors for the previous instrumental samples and analyse how they represent the following aspects: percussive/non-percussive sounds, sustained/non sustained, low-pitch/high pitch, and instrument.
In order to do that, you can build 2-D plots visualizing the values of 2 descriptors for the different samples, e.g.:

- Spectral Flux mean vs Spectral Spread mean
- Spectral Flux mean vs Spectral Flatness
- Spectral Centroid mean vs Zero Crossing Rate mean
- Temporal Centroid vs Log Attack Time (you would need to normalize temporal centroid by the duration of each sound).

Task 4 (week1)

Imagine and describe in a short paragraph (max 5/6 lines) a sound-based multimedia application for which you could use your previous work in this assignment.

Week2:

- A further task to announce
- Write a report with the above work.

Task 5 (week2)

Classification
Label each audio samples in terms of:
- Percussive / non percussive
- Sustained / non sustained
- Instrument (first letters of the title)

Choose a set of rules to automatically identify, by means of the descriptors that you have extracted, if an analysed sound is:
- Percussive vs non-percussive sounds
- Sustained vs non-sustained sounds
- Instrument

Evaluate the accuracy of your system using accuracy (% of correctly identified or classified samples). For a more comprehensive evaluation, you can consider precision & recall measures (http://en.wikipedia.org/wiki/Precision_and_recall).

Task 6

Write the Report. Maximum 4 pages, with references.

4. Evaluation Criteria

- 2 points: Task1
- 4 points: Task2
- 4 points: Task3
- 1 points: Task4
- 3 points: Task5
- 4 points: Task6 (quality of written report)
- 2 points: Best classification results or quality of work among groups.

5. References

Bogdanov, D., Wack, N., Emilia, G., Gulati, S., Herrera, P., Mayor, O., Roma, G., & Salamon, J. (2013). Essentia: An Audio Analysis Library for Music Information Retrieval. *ISMIR 2013*, 2–7.

Lartillot, O., & Toiviainen, P. (2007). A Matlab Toolbox for Musical Feature Extraction from Audio. *Proc of the 10th International Conference on Digital Audio Effects DAFx07*, 1–8. http://dafx.labri.fr/main/papers/p237.pdf

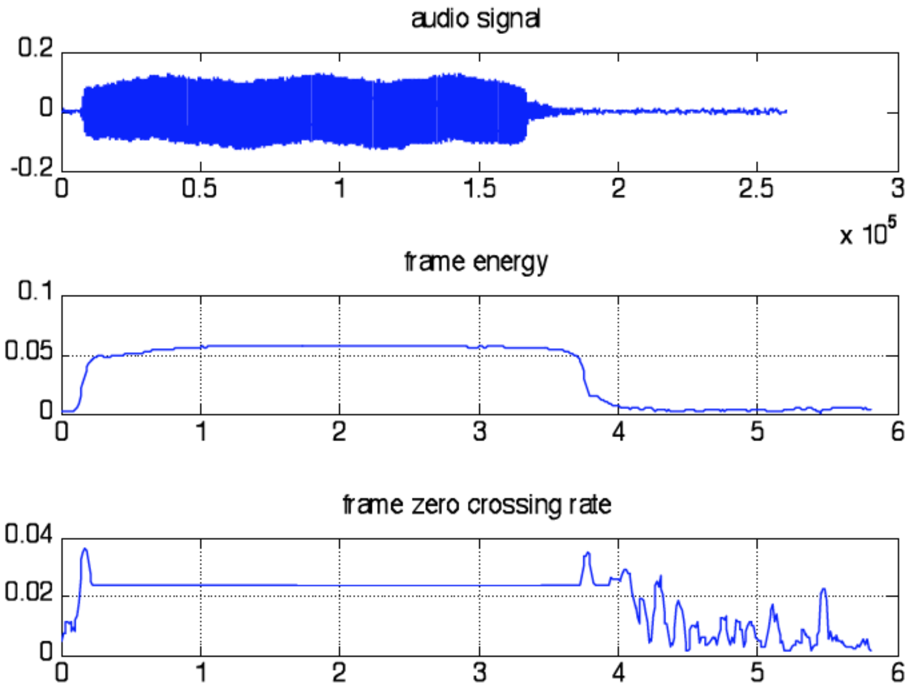
Peeters, G. (2004). *A large set of Audio features for sound description (similarity and classification) in the CUIDADO project*.

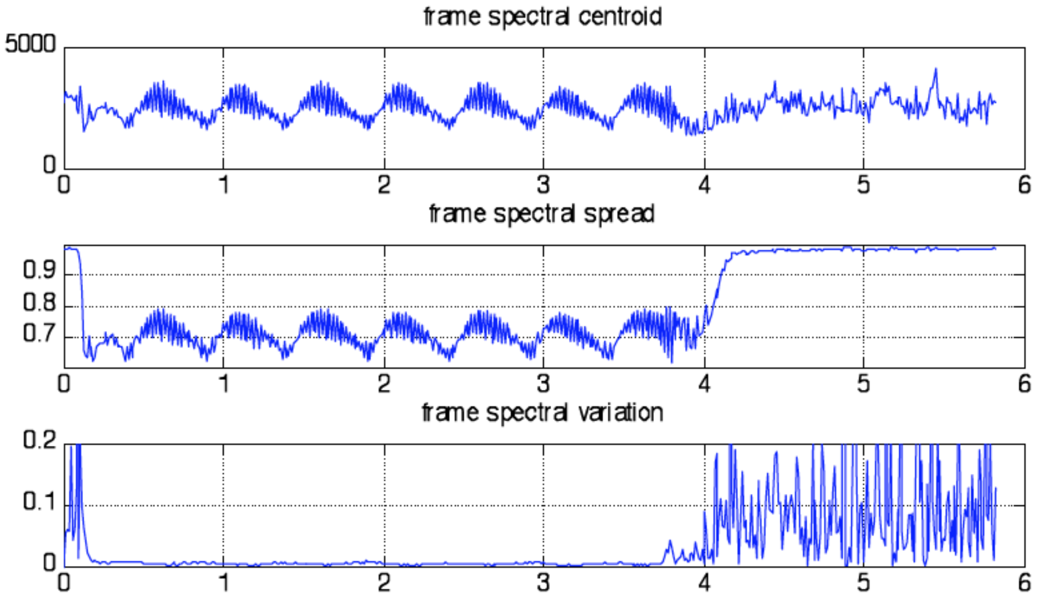
Peeters, G., Giordano, B. L., Susini, P., Misdariis, N., & McAdams, S. (2011). The Timbre Toolbox: Extracting audio descriptors from musical signals. *The Journal of the Acoustical Society of America*, *130*(5), 2902–2916. https://doi.org/10.1121/1.3642604

6. Example

Audio file: acco_co_md_sequ_mf_do4_12.wav

Instantaneous descriptors (square window applied just for testing purposes!!!!):





Global Descriptors:

logAttackTime=-1 (threshold = 20%-80%)
temporalCentroid=2.1131
zcr_mean=0.019278
zcr_std=0.0081513
spec_centroid_mean=2500.4536
spec_centroid_std=496.2628
spec_spread_mean=0.79401
spec_spread_std=0.13328
spec_variation_mean=0.034877
spec_variation_std=0.062566

Example of Task 2

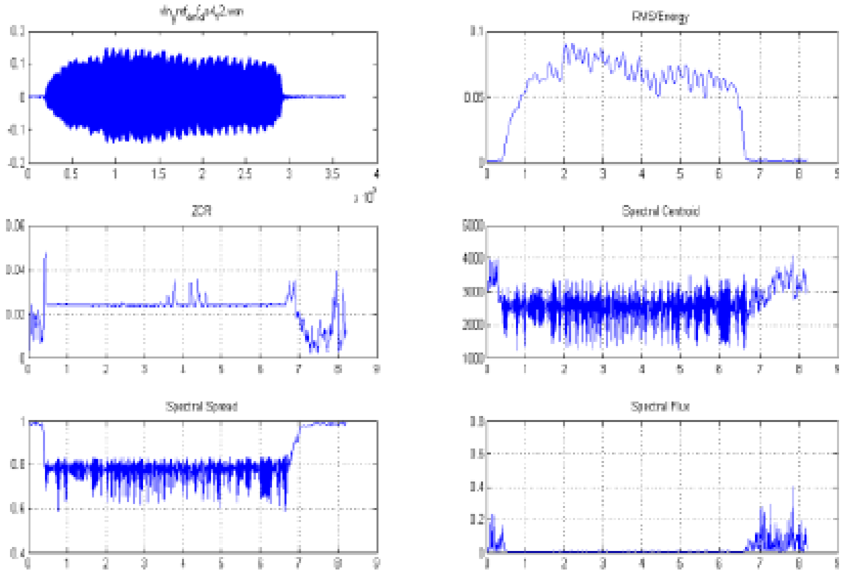


Figure 1. Low Level Features for Violin

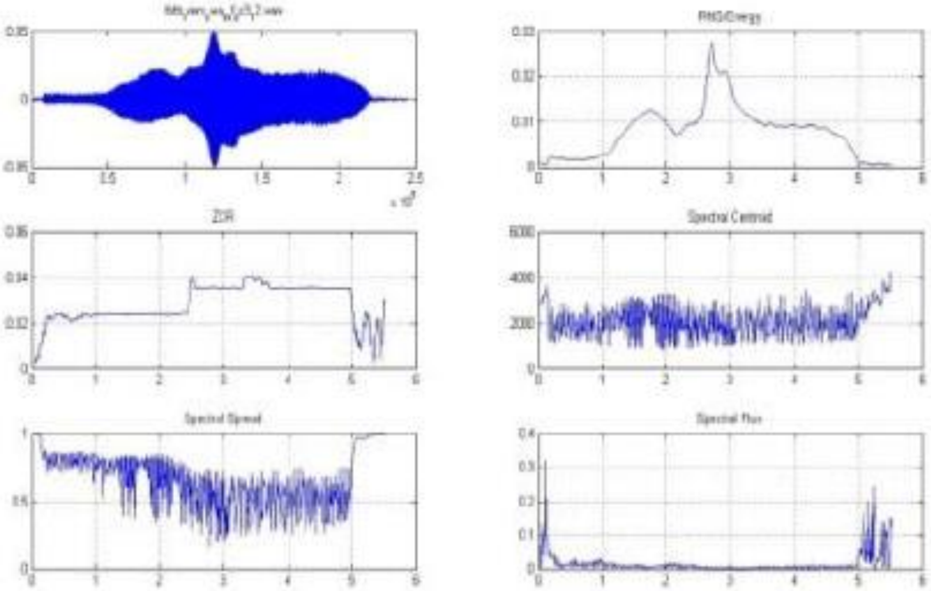


Figure 2. Low Level Features for Tuba

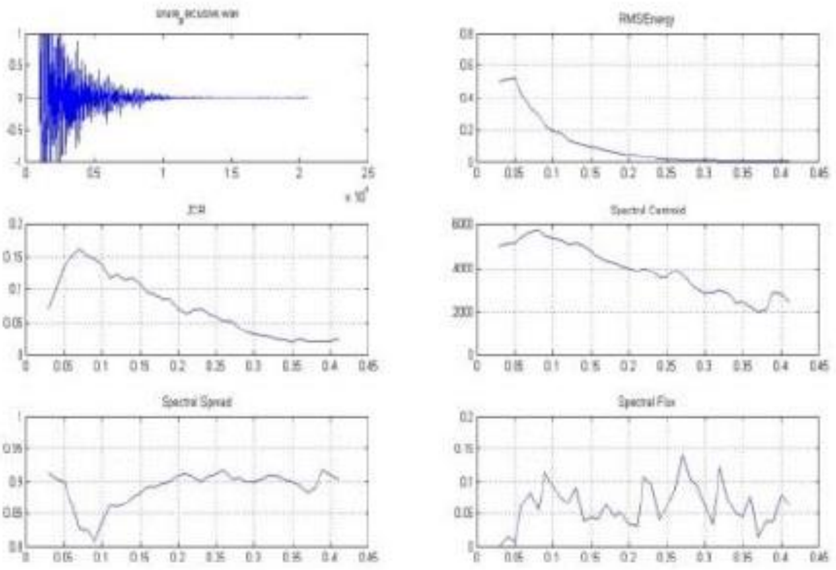


Figure 3. Low Level Features for Snare

Example of Task 3 (Bad example, as it’s not easy to discriminate between classes)

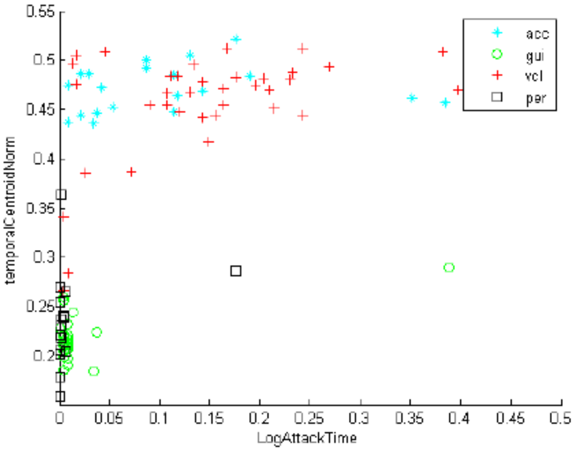


Figure 4. *Log-Attack Time* (mean) vs *Temporal Centroid* (normalized)

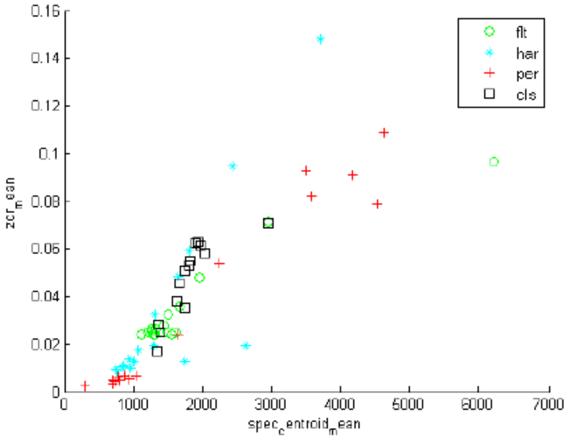
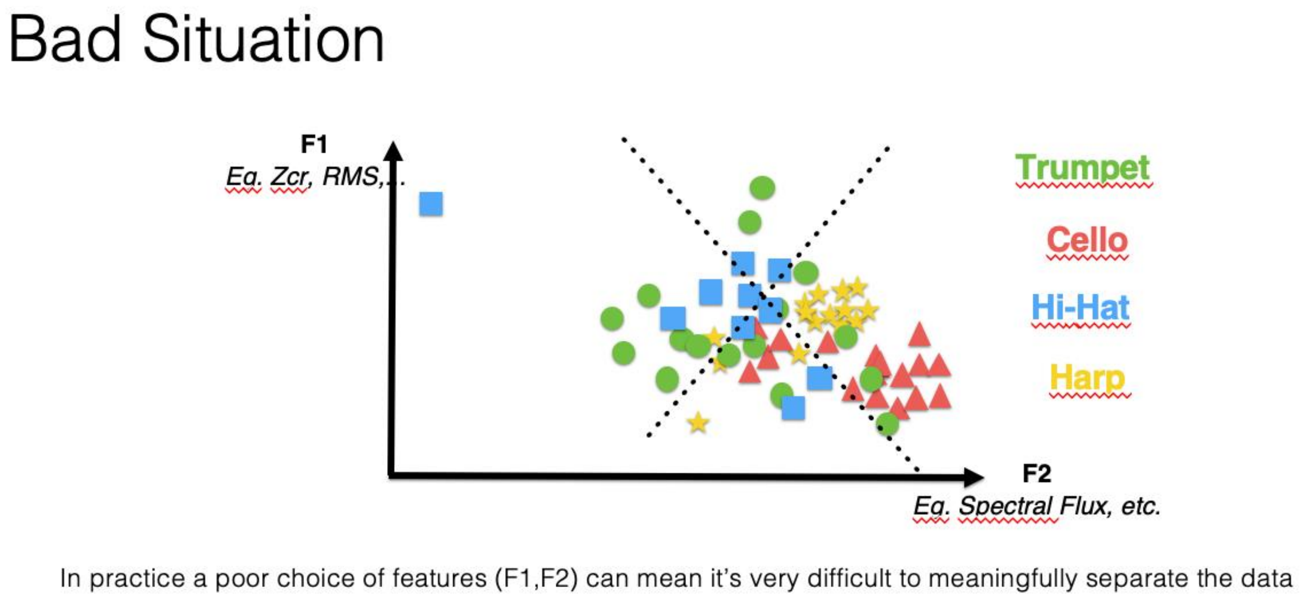


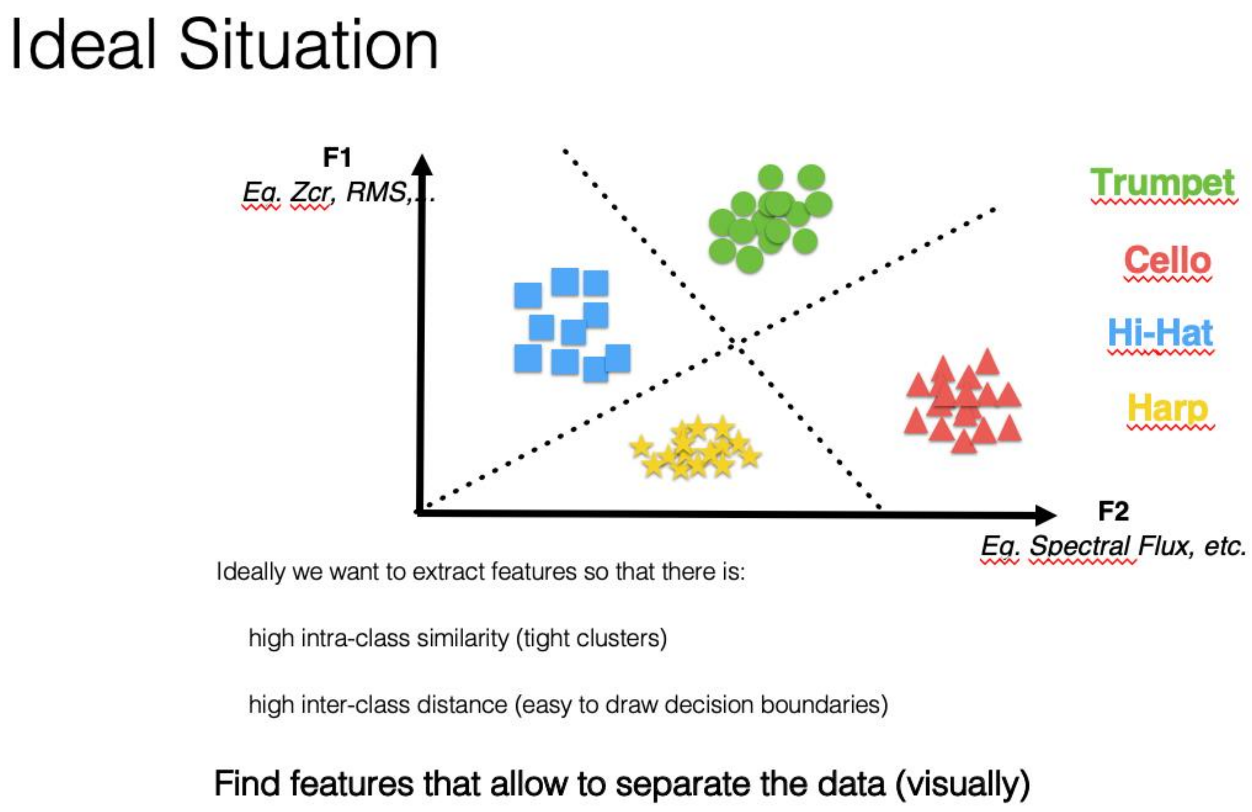
Figure 5. Spectral Centroid (mean) vs ZCR (mean)

Theoretical basis for Task3 and Task5



Bad Situation

In practice a poor choice of features (F1,F2) can mean it's very difficult to meaningfully separate the data



Ideal Situation

Ideally we want to extract features so that there is:

high intra-class similarity (tight clusters)

high inter-class distance (easy to draw decision boundaries)

Find features that allow to separate the data (visually)