A10-2: Sound and music description, revisited

Audio Signal Processing for Music Applications

Introduction

(This assignment needs a working installation of Essentia library and requires good amount of independent programming.)

In this assignment, you will extend the sound and music description task you did in A9 to a larger set of instruments and explore possible improvements in a task of Instrument identification from single note/stroke sounds. By doing this assignment, you will get hands on experience with Essentia and better insights into complexities arising in a real world Music Information Retrieval problem, with a larger set of descriptors and more instrument classes. You will present the results and findings as a short report.

Gudelines

In A9, you explored the tasks of clustering and classification with sound excerpts of three instruments. Typically, as we add more instruments for clustering, the average performance degrades. In such situations, clustering performance can be improved in several ways, two of which you will explore in this assignment:

- **Improving descriptor selection:** We can use a better set of descriptors, in addition to the ones you used in A9. To improve performance, we also need to typically increase the number of descriptors used for clustering as the number of instrument classes increase.
- **Improving descriptor computation**: In A9, each descriptor you used is a time averaged mean of the feature computed over short frames of the audio file. However, there are segments in an audio file (typically at the beginning and the end) where there is silence or low energy background noise. Such segments should be discarded while computing the global statistics of the descriptors, e.g., low energy regions (silence) have a higher spectral centroid and affect the average spectral centroid adversely if included in computing the average.

You will use Essentia to implement both these improvements. You can use the scripts provided with A9 as a base (get A9 scripts) and build your code using them. You first need to install Essentia to compute some of the descriptors that you will be exploring for the task. You can find the download and install instructions for Essentia here: `http://essentia.upf.edu/`. Essentia has extensive documentation that will be useful in this assignment `http://essentia.upf.edu/documentation/index.html`.

The questions in the assignment have been presented separately for the ease of evaluation. But you will write the answers to all the questions together in a document and upload your report (PDF, 2 pages max., excluding plots/illustrations/parameter listings) in Question 1. You must also upload the code that you write. You will evaluate a minimum of three other peers in this assignment.

Question 1: Downloading sounds

Choose at least 10 different instrumental sounds classes from the following possible classes: violin, guitar, bassoon, trumpet, clarinet, cello, naobo, snare drum, flute, mridangam, daluo, xiaoluo. For each instrument class, use `soundDownload.py` script from A9 to download the audio and

descriptors of 20 examples of representative single notes/strokes of each instrument. Since you will use the sounds also to extract descriptors using Essentia, make sure you download the high quality mp3 and not the low quality mp3 preview. To achieve this in the soundDownload.py script, you can replace `fs.FSRequest.retrieve(sound.previews.preview_lq_mp3, fsClnt, mp3Path)` to `fs.FSRequest.retrieve(sound.previews.preview_hq_mp3, fsClnt, mp3Path)`

In the report, explain your choices, query text and the tags you used. Include the Freesound links to the downloaded sounds.

Question 2: Obtaining a baseline clustering performance

Visualize different pairs of descriptors and choose a subset of the descriptors you downloaded along with the audio (same as A9) for a good separation between classes. Run a k-means clustering task with the 10 instrument dataset using the chosen subset of descriptors. You can use the `soundAnalysis.py` script from A9 for this task. Use the same number of clusters as the number of different instruments.

Report the subset of descriptors used and the clustering accuracy you obtained. Since k-means algorithm is randomly initiated and gives a different result every time it is run, report the average performance over 10 runs of the algorithm. This performance result acts as your baseline, over which you will improve in Question 3.

Obtaining a baseline performance is necessary to suggest and evaluate improvements. For the 10 instrument class problem, the random baseline is 10% (randomly choosing one out of the ten classes). But as you will see, the baseline you obtain will be higher that 10%, but lower than that you obtained for three instruments in A9 (with a careful selection of descriptors).

You will upload a single PDF file containing a report answering all questions of this assignment.

Question 3: Suggest improvements

As you can observe, the clustering performance is poorer with 10 instruments. Using Essentia, you will implement the two different improvements described in the introduction of this assignment:

- **Better and more features**: Shortlist a set of descriptors based on the sound characteristics of the instruments such that they can differentiate between the instruments. The choice of the descriptors computed is up to you. We suggest you compute many different descriptors similar to the ones returned by Freesound API, and additional ones described in the class lectures. The descriptors you used in A9 (but now computed using Essentia) are a good starting point. You can use the Essentia extractors that compute many frame-wise low level descriptors together (`http://essentia.upf.edu/documentation/algorithms\_overview.html#extractors`)You can then use a subset of them for clustering for an improved clustering performance.
- **Computing the descriptors stripping the silences and noise at the beginning/end**: For each sound, compute the energy of each frame of audio. You can then detect the low energy frames (silence) using a threshold on the energy of the frame. Since most of the single notes you will use are well recorded, the energy of silence regions is very low and a single threshold might work well for all the sounds. Plot the frame energy over time for a few sounds to determine a meaningful energy threshold. Subsequently, compute the mean descriptor value discarding these silent frames.

Report the set of descriptors you computed and the performance it achieves, along with a brief explanation of your observations. You can also report the results for several combinations of features and finally report the best performance you achieved. Upload the code for computing the non-silent regions and for computing the descriptors that you used. Apart from the two enhancements suggested above, you are free to try further enhancements that improve clustering performance. In your report, describe these enhancements and the improvement they resulted in.

Please upload now the code. You will be evaluated on the code you upload, and the observations and discussion in your report.