Summary “A Comparative Study on Content-Based Music Genre Classification”

- Intro : music can be **hierarchically** classified / classification was primarily done by hand / genres evolve with time / similar instruments, rhythmic patterns, pitch distributions → automatic classification seems feasible

- Features extraction must be *comprehensive* (represent the music well), *compact* (small storage), *effective* (not too much computation to do to extract). *Comprehensive* implies that **high-level** features (closer to human sensitivity) must be preserved.

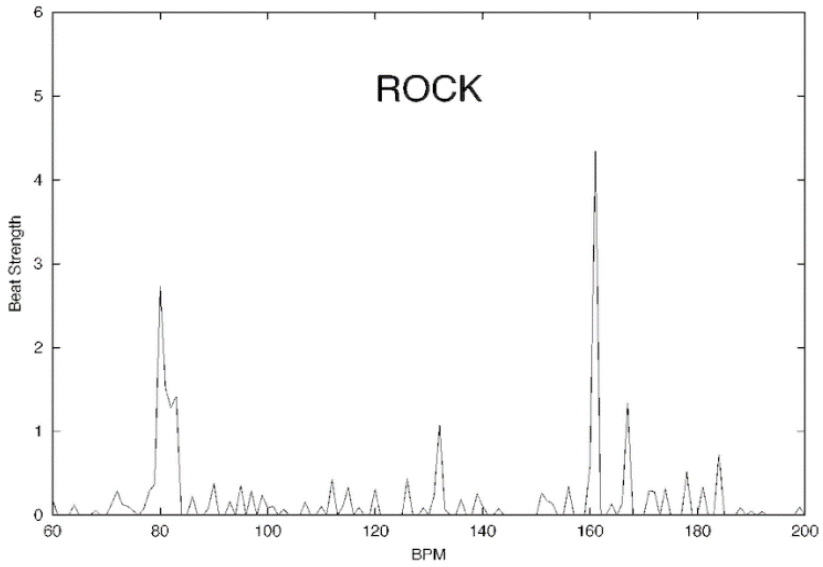
- So far, the main features selected were **rhythmic, pitch-related and timbral** → accuracy was pretty mediocre, can the used features **and** algorithms be improved ?

- Only **content-based** features are used in this study (no information about the author, the singer, the lyrics, the scores… is used)

- Much more work reported on speech recognition than on musical genre classification so far

- **Timbral features :** Short Time Fourier Transform on short overlapping time frames, obtained by *windowing* the signal (usually *Hamming window*, https://fr.wikipedia.org/wiki/Fenêtrage), and then several statistical values are computed : MFCCs (https://en.wikipedia.org/wiki/Mel-frequency_cepstrum), Spectral Centroid, Spectral Roll-off, Spectral Flux, Zero Crossings, Low Energy. Timbral features are computed **locally**, thus not representing very well the song **globally, its texture as a whole**.

- **Rhythmic features :** they are commonly represented with a **beat histogram** (obtained by autocorrelation of the signal after segmentation and subdivision in octave bands : https://fr.wikipedia.org/wiki/Corrélation_croisée), that is computed **globally**, over the whole

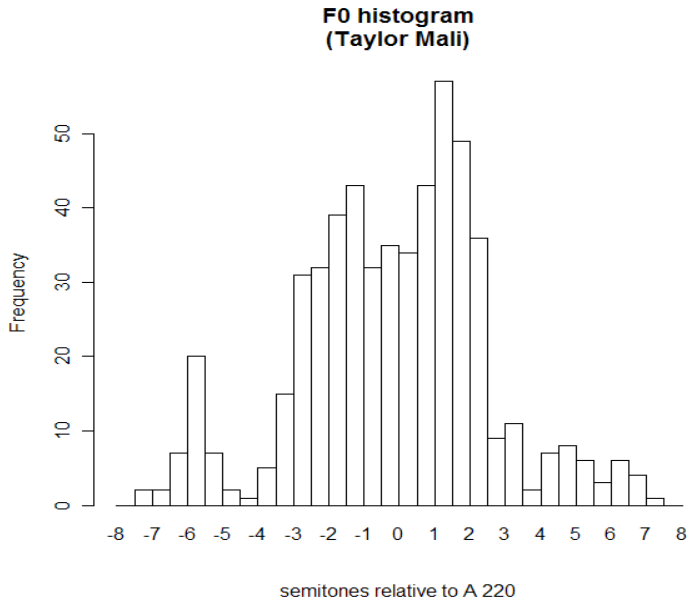


recording, thus losing **local** information. Above is an example of what a beat histogram for a

rock song looks like (we understand that the BPM is 80, represented by the peak at 80bpm, probably the kick drum, and the peak at 160 is certainly hi-hats).

Features usually extracted are : relative amplitude of the first and the second histogram peak, ration of the amplitude of the second peak divided by the amplitude of the first peak, periods of the first and second peak, overall sum of the histogram…

- **Pitch features :** they are commonly represented with a **pitch histogram**, that is obtained in a very similar manner as the beat histogram, and that typically looks like this :

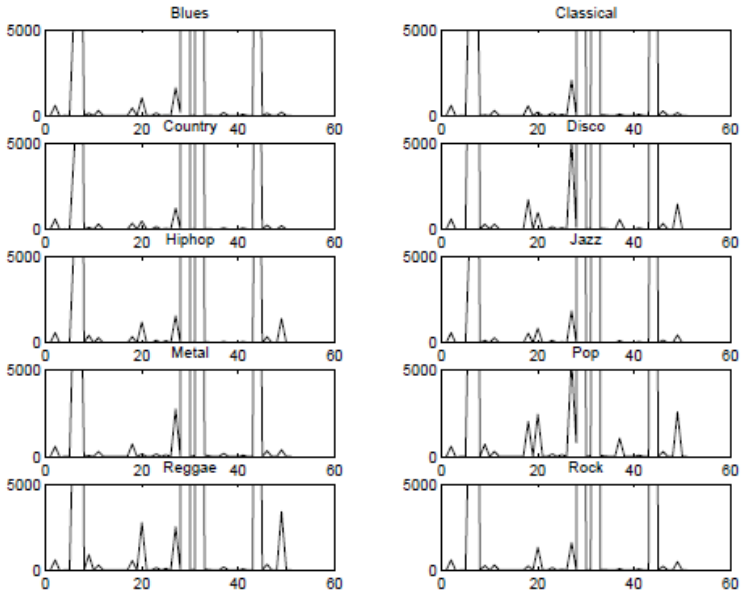


Features usually extracted are : the amplitudes and periods of maximum peaks in the histogram, pitch intervals between the two most prominent peaks, the overall sums of the histograms… It is also **global**, thus losing **local** information.

- New idea : using **wavelet transform** on the signal (https://fr.wikipedia.org/wiki/Ondelette), which is similar to Fourier transform, but not only retains **frequency** information, but also **time-domain** information. It also helps reduce noise, simplifies the signal in the wavelet domain…

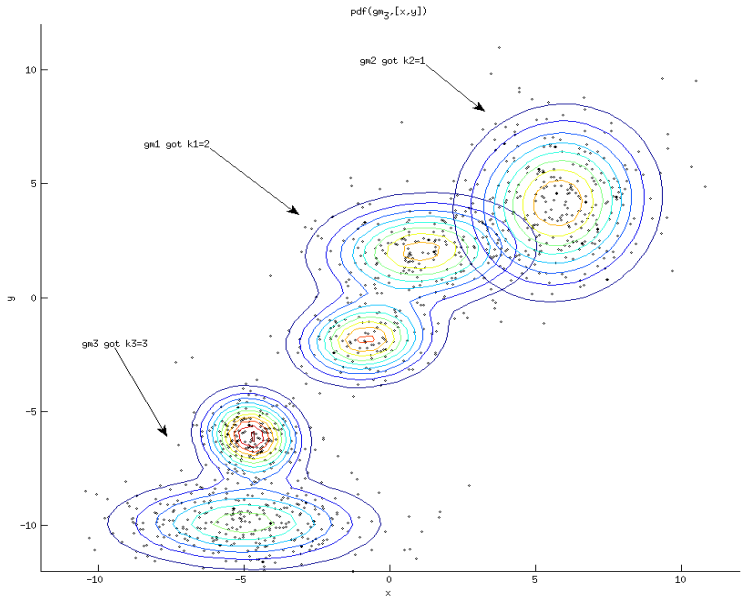
- Wavelet filter used : Daubechies Db8 (8 *vanishing moments*, allow for 7 levels of decomposition) wavelets. The first 3 moments (average, variance, skewness) (*espérance*, *variance, asymétrie*) are used as well as the *subband energy* (mean of the absolute values of the wavelet coefficients for each subband).

- **[features used]** All the traditional features are kept (timbre, beat, pitch), and the Daubechies Wavelet Coefficient Histograms (DWCHs) are a new feature. Here is a representation of the DWCHs of 10 songs in 10 different genres.



- Certain binary classification algorithms transition well into multi-class classification (LDA, KNN, regression, decision trees…) while others (ex : SVMs) have to be combined using multiple binary classification steps to obtain a final multi-class result (using one-vs-the-rest, pairwise comparison, ECOC, multi-class objective functions…).

- **[algorithms used]** **SVMs** extended to multi-class using one-vs-the-rest, pairwise comparison and multi-class objective functions / **KNN** / **LDA** / **Gaussian Mixture Models (GMM) :** each class is represented by a linear combination of (multidimensional) Gaussian distributions, obtained through EM (Expectation Maximization algorithm) on the dataset. Then, during the test phase, the test data point is assigned to the class to which it has the highest probability of belonging, according to the GMM. This is what 2D GMM looks like, with 3 classes :



- **[datasets]** The first dataset contains 1000 30-seconds-long songs over ten genres with 100 songs per genre. The ten genres are Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, and Rock. Available for free download here : http://marsyasweb.appspot.com/download/data_sets/
  The second dataset was constructed from the personal collection of one of the authors, with 756 30-seconds-long recordings taken out of 189 albums, with : 109 "Ambient" files, 164 "Classical" files, 136 "Fusion" files, 251 "Jazz" files, and 96 "Rock" files.
  For both datasets, the sound files are converted to 22050Hz, 16-bit, mono audio files.
  Training, testing and accuracy-computing was done using **ten-fold cross validation**.

- **[extraction of data]** use of the MARSYAS software (http://marsyasweb.appspot.com/) to extract the *MFCCs*, Timbral (*FFT*), Rhythmic (*Beat*) and *Pitch* content. DWCHs are also extracted (in Matlab apparently, following the procedure at the end of Section 4.2), but only keeping the 4 most informative subbands.

- **[algs parameters]** **SVM** with one-vs-the-rest, pairwise, and multi-class objective functions, with linear, polynomial, and rbf kernels / **GMM** with 3 Gaussian mixtures / **KNN** with k = 5 / **LDA**

- **[results]** Features' influence on performance : Beat and Pitch features, used individually, perform poorly (between 20% and 30% accuracy, with the "random attribution performance" being 10% on dataset A). FFT and MFCC (both related to timbral features), used individually, reach between 50% and 60%. When combining some of these 4 features together, performance reaches around 70% maximum, with combinations keeping both FFT and MFCC performing significantly better than the others (**only FFT+MFCC** is almost as good as the 4 features combined).

  ➔ **DWCHs alone outperformed all the previous combinations**, reaching more than 75% with certain SVMs (non-trained humans hardly manage to get 70% of good results when trying to classify musical genre).

  When **reducing the number of classes**, performance logically increases (90% accuracy when only genres 1 to 4 are kept).

  Classifiers' influence on performance : **SVMs** with one-vs-the-rest and pairwise comparison outperform the rest by at least 5% each time, except for LDA which manages to achieve decent performance.

  ➔ **The choice of features is more important than the choice of classifier. DWCHs are promising, especially when used with SVMs.**