# A STATISTICAL APPROACH TO MUSICAL GENRE CLASSIFICATION USING NON-NEGATIVE MATRIX FACTORIZATION

*André Holzapfel and Yannis Stylianou*

University of Crete,
Computer Science Department, Media Informatics Lab
{hannover, yannis}@csd.uoc.gr

## ABSTRACT

This paper introduces a new feature set based on a Non-negative Matrix Factorization approach for the classification of musical signals into genres, only using synchronous organization of music events (vertical dimension of music). This feature set generates a vector space to describe the spectrogram representation of a music signal. The space is modeled statistically by a mixture of Gaussians (GMM). A new signal is classified by considering the likelihoods over all the estimated feature vectors given these statistical models, without constructing a model for the signal itself. Cross-validation tests on two commonly utilized datasets for this task show the superiority of the proposed features compared to the widely used MFCC type of representation based on classification accuracies (over 9% of improvement), as well as on a stability measure introduced in this paper for GMM.

***Index Terms***— Music Genre Classification, Non-negative Matrix Factorization, Gaussian Mixture Model, MFCC

## 1. INTRODUCTION

When investigating the structure of music, it is considered to have a vertical and a horizontal dimension [1]. These dimensions can be recognized when reading the score of the piece. Vertical dimension involves harmonic relations of synchronous sounds and the instrumental timbre, while horizontal dimension involves rhythm and melodic elements. From a signal processing point of view, both dimensions are depicted in a spectrogram-type representation of a piece. In this case, however, timbres have been additively mixed and they are not vertically sorted like in a score. When a piece of music is perceived to be similar to another piece of music, it is expected that these pieces share some of the elements in both directions. It would be interesting to be able to automatically detect these common elements when only a music signal is presented.

Measuring the similarity of pieces of music is an important and challenging task. As the size of digital collections of musical data is growing bigger there is a need for automatically organizing these data. But as music similarity is highly subjective and correctness of measures are difficult to judge it is practical to restrict the task to a more feasible problem. This is the assignment of musical pieces to a set of classes. These classes are usually referred to as musical genres. Some databases have been published in which the musical content can be distributed for non-commercial purposes and a genre classification already exists [2]. These databases may be used as benchmarks for measuring music similarity.

There have been several tries to measure the similarity of music by incorporating features for the vertical description of sounds. These approaches usually rely on Mel-Frequency Cepstrum Coefficients (MFCC) [3]. It has been shown that modeling the vertical structure using MFCC, an upper performance bound may be reached [4]. This raises the task for searching for a new feature set by exploiting the vertical structure of music more efficiently.

As mentioned above the musical instruments are additively mixed in a spectrogram. Techniques to find these components from a mixture include *Independent Subspace Analysis* (ISA) and *Non-negative Matrix Factorization* (NMF). NMF was successfully applied to the decomposition of sound mixtures in [5]. Recently NMF was used for the classification of musical instruments in [6]. These approaches follow a deterministic path for classification; first, a fixed set of spectral bases is defined and then, classification is performed by projecting the input signals into the space generated by these fixed spectral bases.

In this paper, we present a feature set that captures the vertical dimension of music by computing an NMF on spectrograms of music signals. The factorization step provides base vectors of the spectral space where the signal is supposed to lie within. The number of base vectors is determined based on a Singular Value Decomposition of the spectrogram before factorization. For a given musical genre, a Gaussian Mixture Model (GMM) is built on all the base vectors computed from the training data. For classification we do not model the songs statistically; classification decision is based on the likelihoods of the song feature vectors given the statistical models. The ability of such a feature set to extract significant characteristics from the music to be classified is the central item of this paper. Our approach does not include descriptors for the horizontal structure of music, such as melody and rhythm. We rather aim to evaluate a new set of features for the vertical dimension.

Section 2 will give a detailed overview of the feature calculation frontend and Section 3 will describe the classification approach. An objective way for measuring the performance of the proposed system is presented in Section 4. For this purpose, a general approach to measure sensitivity in the classification task using GMM is presented along with experiments using the proposed system. Details about our experiments, a description of the databases we have used, and the obtained results are presented in Section 5. Conclusions drawn from our experiments are provided in Section 6.

## 2. FEATURE SET

### 2.1. NMF

A spectrogram, $\mathbf{X} \in \mathbb{R}^{N_c \times k}$, contains in its columns $k$ vectors of $N_c$ coefficients computed using the magnitude of the *Short Time Fourier*

This work has been funded by the German Academic Exchange Service (DAAD)

*Transform* (STFT) on signal $\mathbf{x}$. Usually, to decompose this representation of a mixture into its elementary components it is assumed that the number of observation vectors $k$ is higher than the number of the elementary components. Assuming that the observations are the output of an additive mixture, the observation matrix $\mathbf{X}$ can be approximated by

$$\mathbf{X} \approx \mathbf{W}\mathbf{H} = \sum_{i=1}^{d} \mathbf{w}_i \mathbf{h}_i \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{N_c \times d}$ is the mixing matrix, $\mathbf{H} \in \mathbb{R}^{d \times k}$ is the resulting component matrix, $\mathbf{w}_i$ is the $i$-th column of $\mathbf{W}$, and $\mathbf{h}_i$ is the $i$-th row of $\mathbf{H}$. Parameter $d$, with $d < k$, denotes the number of components contained in the mixture. In this paper, we consider the columns of $\mathbf{W}$ to represent a *possible* spectral base of the signal. Rows of $\mathbf{H}$ contain the temporal weights throughout the mixture.
NMF performs the factorization of matrices shown in (1) by minimizing the error function:

$$D(\mathbf{X}||\mathbf{W}\mathbf{H}) = \sum_{i,j} \left( \mathbf{X}_{i,j} \log \frac{\mathbf{X}_{i,j}}{[\mathbf{W}\mathbf{H}]_{i,j}} - \mathbf{X}_{i,j} + [\mathbf{W}\mathbf{H}]_{i,j} \right) \tag{2}$$

under the constraint that $\mathbf{W}, \mathbf{H}$ and $\mathbf{X}$ are non-negative. This problem is guaranteed to converge to a local minimum using efficient gradient decent algorithms with multiplicative updates as shown in [7].

### 2.2. Feature Calculation

The features describing the spectral space are calculated as shown in figure 1. The preprocessing step avoids the influence of recording conditions which are not considered as significant for classification. It includes removal of mean values and normalization to an average sound pressure level of $L = 96dB$ by applying

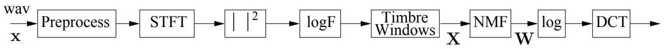

**Fig. 1**. Calculation of the features used for the statistical model of musical genres

$$\mathbf{x}_{norm} = \left( \frac{\mathbf{x}}{\sqrt{\frac{\sum_{i=1}^{N_s} |x_i|^2}{N_s}}} \right) 10^{96/20} \tag{3}$$

to the zero mean signal vector $\mathbf{x}$ of length $N_s$ (in samples). Then, the power spectrum from the STFT of the signal is computed on a 40ms Hamming window with $50\%$ of overlap. The next step is a conversion from the linear frequency abscissa to a logarithmic axis. We use eight bands per octave ranging from 65.5 Hz to 8 kHz. This conversion is following the *AudioSpectrumEnvelopeType* descriptor of the MPEG-7 standard. It enables a more compact description of the signal, reducing from $N_c$ to $N_{Bands} = 56$ (8 bands per octave $\cdot$ 7 octaves). The choice of eight bands per octave has been motivated by the tempered musical system of western music in which tonal scales contain seven steps from the fundamental tone until its octave. Having computed these vectors for a whole song, a spectrogram representation is then obtained. This is segmented into smaller sub-spectrograms that represent non-overlapping windows of $t_{Block}$ seconds length in the time signal. Each sub-spectrogram is then factorized using NMF, providing a spectral base in the columns of matrix $\mathbf{W}$ (see (1)). The final step of the feature calculation is a *Discrete Cosine Transform* (DCT) on the logarithm of the spectral base vectors. This helps to reduce the dimensionality of the space. Note that the whole process is similar to the calculation of MFCC. Because of this, any performance improvement can be then attributed to the factorization step. For a given spectrogram the determination of the optimum values for the temporal length, $t_{Block}$, of the timbre window and the number, $d$, of spectral base vectors to compute, should be defined. We have tested values for $t_{Block}$ from $0.25$ seconds to $3$ seconds. In order to get a value for $d$, we have varied the values of ratio:

$$\phi = \frac{\sum_{j=1}^{d} \sigma_j}{\sum_{i=1}^{N_{Bands}} \sigma_i} \tag{4}$$

from $0.95$ to $0.6$, where $\sigma_i$ is the $i$-th singular value of a *Singular Value Decomposition* (SVD) of the spectrogram to be factorized. For evaluation, a subset of four classes (classical, disco, metal, rock) from the first database has been used, while a mixture of Gaussians with five components with full covariances has been built for each genre (see Section 3 for details). The best classification was achieved with $t_{Block} = 0.5s$ and $\phi = 0.6$, resulting in $d = 3$ for these data. Note that these values are then fixed for all the following experiments.

## 3. STATISTICAL MODEL AND CLASSIFICATION

In order to construct the models for the music genres, we calculate the features for all samples of the database and store the features for each class separately. Then, a *Gaussian Mixture Model* (GMM), $\theta_i$, for each genre is built (i.e., with $i = 1...C$, where $C$ denotes the number of genres), using a standard *Expectation Maximization* (EM) algorithm [8]. EM algorithm is initialized by a deterministic procedure based on the Gaussian means algorithm presented in [9]. A new song is classified into a genre by computing the likelihood of its features given the genre models, $\theta_i$, with $i = 1...C$. Summing up these likelihood values, the song is assigned to the genre that has the maximum summation value. The principle of the model training and classification is depicted in Figure 2.

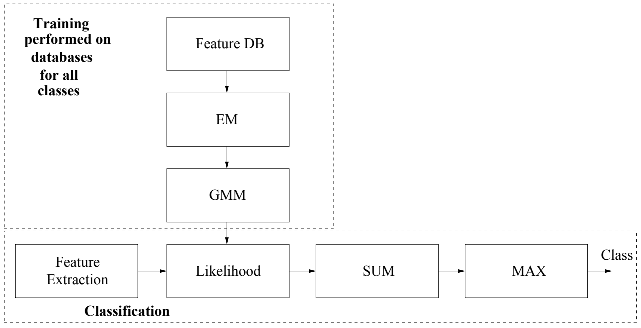

**Fig. 2**. Model estimation and classification of data

## 4. JUDGMENT OF PERFORMANCE

### 4.1. Baseline System

In order to evaluate the performance of our classification approach it is necessary to compare with some kind of standard procedure used in many recent publications. For this, we implemented a *baseline* system that is as close as possible to the proposed classification system except for the feature calculation approach. The form of the baseline system was motivated by [4] which is a state of the art system for capturing the vertical structure of music. The model estimation and classification procedures follow exactly the pattern described by Figure 2. As proposed in [4], MFCCs have been used as features for the baseline system.

### 4.2. A Measure of Sensitivity

In addition to comparing the performance of the proposed classification system with that of the baseline system, we also judge the quality of the classifiers based on a measure that estimates their sensitivity (or stability).

In order to judge the stability of a trained GMM, a method based on Kullback Leibler divergence (KLD) was implemented. The KLD between two distributions $f$ and $g$, is given by

$$KL(f||g) = \int f(x) \log \frac{f(x)}{g(x)} dx \tag{5}$$

Since there is no closed form expression for KLD in a GMM context, a possible way to get a distance measure in this case is by generating $M$ samples from $f$ and then approximate KLD, by:

$$KL(f||g) \approx \frac{1}{M} \sum_{t=1}^{M} \log \frac{f(x_t)}{g(x_t)} \tag{6}$$

Based on (6) a symmetric distance measure may be then defined as:

$$D_{KL}(f,g) = KL(f||g) + KL(g||f) \tag{7}$$

Our problem consists of the classification into one of $C$ classes. Performing an $n$-fold cross-validation we will get a set of $n \times C$ GMMs. We can now determine the distances between the GMMs of different classes using (7) for each of the $n$ cross-validation runs separately. The minimum of these values throughout the cross-validation runs gives us the least distance, $D_{inter}$, between two different classes. Then we calculate the distances within the classes throughout the different cross-validation runs. The biggest value along all classes, $D_{intra}$, gives us a measure of how much the model differs throughout the cross-validation due to diversity of the data set. We can now define a condition measure for a specific feature set, computed by:

$$Cond = \frac{D_{inter}}{D_{intra}} \tag{8}$$

Obviously values for $Cond$ smaller than 1 for a specific feature set imply that a classification with this feature set might be unreliable. This is because there is a high variability between models built from a different set of data for a specific genre, while at the same time there is a relatively small distance between the models for different genres.

## 5. EXPERIMENTS

### 5.1. Databases

Two different data sets have been used for the experiments. All the samples are monaural wave files at a sampling frequency of 16000 Hz and quantized with 16 bits. The first database (D1) consists of ten classes[1], each containing 100 subsections of musical pieces of 30 seconds length. The database was collected by Giorgos Tzanetakis [10] and has been used for performance evaluation by other researchers as well [11]. The second database (D2) has been downloaded from the website of the ISMIR contest in 2004[2]. The songs had been selected from the *magnatune* collection. D2 consists of six classes[3] that are not equally distributed as they are in D1. In D2, pieces are full musical pieces and not snapshots as in D1; therefore the lengths of pieces in D2 differ. All the classification accuracies shown in this paper are the means of the obtained accuracies from 5-fold cross-validations on the whole databases.

### 5.2. Classification results

Table 1 shows the classification scores on the two databases. The rows marked with NMF contain results achieved with the system using the NMF-based features while rows marked with MFCC contain results achieved with the baseline system (using MFCC). The values in parentheses denote the number of mixture components in GMM. Full covariance matrices have been used for all experiments. Note, that for D1, we didn't have enough data for the NMF-based features to train the GMM with 30 components. The results show that the proposed system outperforms the baseline system. This is more evident for the second database (D2). Here we were able to increase the number of components further as D2 contains more data. We observed that in most of the cases misclassifications have some musical sense. For example, the genre Rock in D1 was confused most of the time with either Metal or Country. In D2 the Rock/Pop genre was mostly misclassified as Metal/Punk pieces. Genres which are assumed to be very different, like Metal and Classic, were never confused. The worst classification performance for the proposed system was: Rock in D1 (42%, NMF(20)) and World in D2 (37.6 %, NMF(30)). It is worth to note that this behavior in performance is similar for other systems as well [3] [12]. The low performance for these genres may be assigned to their large intra-variance of music style (at least for the analyzed data).

**Table 1**. Classification Accuracies in %

| | Database 1 | Database 2 |
|---|---|---|
| NMF(10) | 69.8 | 70.6 |
| NMF(20) | 72.9 | 70.8 |
| NMF(30) | - | 74.1 |
| MFCC(20) | 71.5 | 64.9 |
| MFCC(30) | 72.0 | 67.8 |

Regarding the time allocated for training, the NMF-based system is very fast compared to the baseline system (with MFCC). For instance, training a 20 component model on D1 took about twenty times longer using the baseline system instead of the NMF-based system. The computation of the features for NMF takes longer than computing MFCC due to the gradient decent algorithm for NMF (about 2.3 times longer). However, regarding the total time for feature calculation and training, NMF-based system is still about 6 times faster than the baseline system.

Even though our system captures only information about the vertical characteristics of music it also performs well in comparison with approaches incorporating more versatile feature sets that include *both* vertical and horizontal directions. On D1, Li and Tzanetakis [10]

[1] Blues, Classical, Country, Disco, Hip Hop, Jazz, Metal, Pop, Reggae, Rock

[2] http://ismir2004.ismir.net

[3] Classical, Electronic, Jazz, Metal/Punk, Rock/Pop, World

reported an accuracy of 71% while recently Bergstra *et al.* [11] reported 83% on the same database. D2 has been used for training in the 2004 ISMIR Audio Description contest. There, the winner reached an accuracy of 78.8% [12] while the second in the rank reached the accuracy of 67.2% [13]. Note, however, that these results have not been obtained from a cross-validation experiment. It is interesting, therefore, to note that for the same database (D2) the NMF-based system in a single validation reached the score of 83.0% (NMF(30)).

### 5.3. Stability Measure

Table 2 shows the condition measure for all the systems using (8). From Table 2 it follows that NMF based features provide constantly a higher condition number than MFCC. Moreover, the condition number for NMF based features is always bigger that 1, while for MFCC this number is *always* less than 1. This indicates that for NMF based features the smallest inter class distance is always bigger than the biggest intra class distance; this is not the case for MFCC based models. This provides a further proof of the superiority of the presented feature set compared to MFCC. From Table 2, we also observe that the condition values for the NMF based system decrease as the number of mixture components increases. This is because the intra class distances, $D_{intra}$, are growing faster than the inter class distances, $D_{inter}$, when using more Gaussians. We further observed that this effect diminishes when the number of Gaussians increases. We assume therefore, that further increase will lead to a stable state. Due to the limited size of the databases, this hypothesis cannot, however, be verified. On the other hand this behavior of the NMF based vectors may indicate their ability to amplify *existing* differences between songs classified in the same genre. At the same time, the classification score still remain high compared to the MFCC baseline system (see Table 1) since the smallest inter class distances were observed to be always bigger for NMF based feature than for MFCC.

**Table 2**. Condition Measure

| | Database 1 | Database 2 |
|---|---|---|
| NMF(10) | 1.40 | 1.70 |
| NMF(20) | 1.39 | 1.30 |
| NMF(30) | - | 1.27 |
| MFCC(20) | 0.33 | 0.55 |
| MFCC(30) | 0.44 | 0.64 |

## 6. CONCLUSION

We presented a new feature set based on NMF of the spectrogram of a music signal for the description of the vertical structure of music. We were able to show its superiority to MFCC which is the standard set of features for describing the vertical dimension of sounds. Using the new feature set classification accuracies have been improved by approximately 9%. Moreover, the proposed classification system is more stable than the system with MFCC by over 98% using a stability criterion based on the inter and intra distances of statistical models in a cross validation test. In addition, the new feature set has the advantage of fast training times compared to MFCC. Future work includes the task to extend the new feature set to the horizonal dimension, *i.e.* rhythm and melody. For example, using the rows of $\mathbf{H}$ in (1) may be a starting point for the estimation of beat occurrences.

## 7. REFERENCES

[1] Bob Snyder, *Music and Memory*, Cambridge, MIT Press, 2000.

[2] *http://www.magnatune.com*

[3] Michael I. Mandel and Daniel P.W. Ellis, *Song-level features and support vector machines for music classification*, 6th International ISMIR 2005 Conference, London, UK, 2005.

[4] Francois Pachet and Jean-Julien Aucouturier, *Improving Timbre Similarity: How high is the sky?*, Journal of negative results in speech and audio sciences, vol.1.1, 2004.

[5] Beiming Wang and Mark D. Plumbley, *Musical audio stream separation by non-negative matrix factorization*, DMRN Summer Conference, Glasgow, UK, 2005.

[6] Emmanouil Benetos and Margarita Kotti and Constantine Kotropoulos, *Musical instrument classification using Non-negative Matrix Factorization algorithms and subset feature selection*, ICASSP, Toulouse, 2006.

[7] Daniel D. Lee and H. Sebastian Seung, *Algorithms for non-negative matrix factorization*, Advances in Neural Information Processing Systems, vol.13, 2001, pp. 556-562.

[8] L. Baum and J. Eagon, *An inequality with applications to statistical estimation for probalistic functions of Markov processes and to a model for ecology*, American Mathematical Society Bulletin, 73, 1967, pp. 360-363.

[9] Greg Hamerly and Charles Elkan, *Learning the k in kmeans*, 17th Annual Conference on Neural Information Processing Systems (NIPS), 2003, pp. 281-288.

[10] Tao Li and Georgos Tzanetakis, *Factors in automatic musical genre classification of audio signals*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, NY, 2003.

[11] James Bergstra and Norman Casagrande and Dumitru Erhan and Douglas Eck and Balász Kégl, *Aggregate features and ADABOOST for music classification*, Kluwer Academic Publishers, 2006.

[12] Elias Pampalk, *A matlab toolbox to compute music similarity from audio*, 5th International ISMIR 2004 Conference, Barcelona, Spain, 2004.

[13] West, K. and Cox, S., 2004, *Features and Classifiers for the automatic classification of musical audio signals*, 5th International ISMIR 2004 Conference, Barcelona, Spain, 2004.