Music Identification using Chroma Features

Adrian Iftene, Andrei Rusu, Alexandra Leahu

UAIC: Faculty of Computer Science, “Alexandru Ioan Cuza” University,
General Berthelot, 16, 700483, Iasi, Romania
{adiftene, rusu.andrei, alexandra.leahu}@infoiasi.ro

Abstract. In this paper some specific issues related to Music Information Retrieval (MIR) are presented. First part is dedicated to introductive notions from this field in the field, and second part is dedicated to giving details about the system we built for MusiCLEF 2011. An important aspect related to our work is related to searching metrics able to allow us to identify an audio recording in a database with existing songs. Our system uses chroma features associated to a song and apply on it many types of metrics. Some of these metrics wants to find more accurate song of which part of that fragment belong to and other metrics are used to enable us to do this as quickly as possible.

1 Introduction

In the last years, we have seen an explosion in terms of the number of existing audio files on the Internet (music files, video files, audio streaming, etc.). Music Information Retrieval (MIR) is devoted to the study of methodologies for automatic music access. Today, although MIR usually focuses on content-based approaches, music search facilities in commercial systems are still based on simple keyword matching between music tags, with additional exploitation of user profiling and collaborative filtering approaches[1].

As the organizers say, MusicCLEF 2011 aims at promoting the development of new methodologies that allow us direct access to audio files, combined with the use of knowledge from the database associated with audio files. For that, organizers allowed access to information automatically extracted from audio files, and additional they allowed access to contextual information provided by users who have had access to their tags or comments and reviews associated with them.

In edition 2011, the organizers propose two tasks: *Content and Context-based Music Retrieval* and *Music Identification*[2]. For this exercise they offer to participants a test collection of about 10,000 songs stored in MP3 format for both tasks. First task, *Content and Context-based Music Retrieval* is a categorization task and it aims to combine information automatically extracted about the content with information generated by user. The second task, *Music Identification* is closed to what MIR meaning, and it has aim to automatically identify an audio recording and to cluster a group of recordings in the same group.

[1] MusiCLEF 2011 Overview: http://ims.dei.unipd.it/websites/MusiCLEF/index.html

[2] MusiCLEF 2011 Tasks: http://ims.dei.unipd.it/websites/MusiCLEF/tasks.html

In the following chapters we present the basic notions used in this paper and the details about our group approach in an attempt to build a system for the second related to Music Identification task.

2 Basic notions

Sound

The first definition of *sound* from the American Heritage Dictionary of the English language is [1]:

Vibrations transmitted through an elastic solid or a liquid or gas, with frequencies in the approximate range of 20 to 20,000 hertz, capable of being detected by human organs of hearing.

The propagation of sound[3] is a sequence of waves of pressure which propagates through compressible media such as air or water or even solids. During their propagation, waves can be *reflected*, *refracted*, or *attenuated* by the medium. All media have three properties which affect the behavior of sound propagation:

1. A *relationship between density and pressure* determines the speed of sound within the medium.
2. The *motion of the medium itself* (e.g., wind): if the medium is moving, the sound is further transported.
3. The *viscosity of the medium* determines the rate at which sound is attenuated. For many media, such as air or water, attenuation due to viscosity is negligible.

Pitch

Pitch[4] is an auditory perceptual property that allows the ordering of sounds on a frequency-related scale [2]. Pitches are compared as “higher” and “lower” in the sense associated with musical melodies [3], which require “sound whose frequency is clear and stable enough to be heard as not noise” [4]. Pitch is one of the auditory attribute of musical tones, along with duration, loudness, and timbre [5].

Octave

In music, an *octave*[5] is the interval between one musical pitch and another with half or double its frequency. The octave relationship is a natural phenomenon that has been referred to as the “basic miracle of music,” the use of which is “common in most musical systems.” [6]

Pitch class

In music, a *pitch class*[6] is a set of all pitches that are a whole number of octaves apart, e.g., the pitch class C consists of the Cs in all octaves. “The pitch class C stands

[3] The propagation of sound: http://www.jhu.edu/virtlab/ray/acoustic.htm

[4] Pitch: http://en.wikipedia.org/wiki/Pitch_%28music%29

[5] Octave: http://en.wikipedia.org/wiki/Octave

[6] Pitch class: http://en.wikipedia.org/wiki/Pitch_class

for all possible Cs, in whatever octave position." [7] Psychologists refer to the quality of a pitch as its "chroma".

Chroma

A *chroma*[7] is an attribute of pitches. Similar, a "pitch class" is a set of all pitches sharing the same chroma. The concept behind chroma is that octaves play a basic role in music perception and composition [8]. Chroma features have been already used in music retrieval applications [9]. Thus, the application of chroma features to identification task has been already been proposed for classical [10] and for pop [11] music.

3 Metrics

For the MusiCLEF 2011, the organizers provide a set of songs which are described by Chroma vectors. In order to extract chroma from mp3 songs, they used the MIRToolbox[8]. Chroma features are considered as pointers to the recordings they belong to, playing the same role of words in textual documents. The information on the time position of chroma features is used to directly access to relevant audio excerpts.

Chroma features consist of a twelve-element vector with each dimension representing the intensity associated with a particular semitone, regardless of octave. The vector has one single integer value and it is obtained through a hashing function. For each song we have a file that contains many chroma vectors that describes the respective mp3 song. We also have a set of fragments of songs that are also described by these vectors.

An example of this type of vector is:

```
  0.98892,   0.95418,   0.89027,   0.93907,   0.75066,   0.77229,
0.71884, 0.72182, 0.89031, 0.9806, 0.85977, 1
```

We can see this vector as a point in a *n*-dimensional space (with *n* equal to 12). From this point of view we decide to use *n*-dimensional metrics in order to calculate distance between chroma associated to songs from our database (Alicante fonoteca in our case).

3.1 Euclidian Metrics

First used metric was **Euclidean**[9] **distance**. In a *n*-dimensional space if we have two points $p = (p_1, p_2, \dots, p_n)$ and $q = (q_1, q_2, \dots, q_n)$ then we can define the *Euclidean distance* between p and q the following value:

[7] Chroma: http://en.wikipedia.org/wiki/Pitch_class

[8] MIRToolbox: https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox

[9] Euclidean distance: http://en.wikipedia.org/wiki/Euclidean_distance

$$d(p,q) = d(q,p) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2}$$

In this case *p* represents a chroma vector from a song from Alicante fonoteca database (more precisely *p* represent a line from a file associated to a song), and *q* represents a chroma vector for a fragment that need to be identified.

Based on this Euclidian distance we define a *global Euclidean distance*, like a sum of Euclidean distances corresponding to all consecutive lines from file associated to the song that must be identified and consecutive lines from a song from our database. The sequence of lines, from our song, from database, which has the global Euclidian distance with minimum value, given to us the *minimum global Euclidean distance*.

In order to classify a song, we need to calculate all possible minimum global Euclidean distances between it and songs from our database and in the end to select the song with minimum global Euclidian distance. In the end we decide that this song is similar with the song from database with shortest minimum global Euclidean distance.

3.2 Relative Euclidian Metrics

This metric is based on Euclidian metrics, but it aims to eliminate similar differences which exist on different octaves in two chromas. Thus, in a *n*-dimensional space if we have two points $p = (p_1, p_2, \dots, p_n)$ and $q = (q_1, q_2, \dots, q_n)$ then we define the *Relative Euclidean distance* between *p* and *q* the following value:

$$d_r(p,q) = d_r(q,p) = \sqrt{(p_1 - q_1)^2 + \cdots + (p_n - q_n)^2 - (n-1) \times \min_{\mathrm{i}} \{(p_i - q_i)^2\}}$$

In comparison with above metric, this metric comes to cover cases when a song is written in other gamma. Obviously it is possible that this metric does not always work well, but we want that after tests on training data, to find a way to automatically identify cases where it would be preferable to use.

3.3 Scalability Metrics

Because it is possible to have very many songs in our database is possible like above metrics not be effective, in terms of scalability. From this reason we try to find a way to reduce the search space, making an obvious reduction in quality of final solutions. For this case instead to consider all lines from a file with chromas associated to a song, we consider the following values for all octaves from these files: *minimum*, *average* and *maximum* value. In this way instead to consider around 30,000 lines for a song from our database and instead to consider around 5,000 lines for a song that must be identified, we consider only three lines of values for both types of files. Thus for two songs s_1 and s_2 there are two metrics available based on Euclidian metrics from above:

$$d_s(s_1, s_2) = d(s_{1min}, s_{2min}) + d(s_{1avg}, s_{2avg}) + d(s_{1max}, s_{2max})$$

and

$$d_{sr}(s_1, s_2) = d_r(s_{1min}, s_{2min}) + d_r(s_{1avg}, s_{2avg}) + d_r(s_{1max}, s_{2max})$$

After we calculate all values for scalability metrics between a song that must be identified and all songs from our database, we decide to classify this song using the minimum value d_s or d_{sr}.

3.4 Combined Metrics

These metrics start from scalability metrics, but instead to keep only the minimum value, we keep first 10 lowest values. On these 10 lowest values we apply corresponding Euclidian or Relative Euclidian metrics and we identify the most suitable file. In this way we perform a faster identification using the scalability metrics from 3.3, and then using the basic metrics from 3.1 and 3.2 we refine the search and in the end we obtain a better solution.

4 Experiments

Another idea that we had was to use clustering algorithms in order to identify similarities between songs, and similarities between different parts of the same song. Thus we first tried to accomplish that using some Java machine learning libraries that are available (such as Java-ML[10]), we soon observed that much faster results can be computed in MATLAB[11].

So far, we used the k-means clustering method with the Euclidean distance measure (and also implemented some efficient methods to compute the global Euclidean distance between *n* points), and we partitioned each set of chroma vectors into ten clusters. The result is a vector representing each song and containing the cluster indices of each point. Also, in addition to that, we made use of a vector containing the within-cluster sums of point-to-centroid distances. Of course, this raw data would be of no use unless processed in the right way, and since the results described so far were computed for individual songs, is natural that for finding similarities between different songs we have to use this data, but apply meta-clustering algorithms on it. Also, we are currently working on a more precise classifier which will enable us the make even more accurate predictions.

5 Instead of conclusions

Until now, our current work was related to find metrics that allow us to identify a song and to classify it accordingly to chroma features associated to it. The results

[10] Java Machine Learning Library: http://java-ml.sourceforge.net/

[11] MATLAB: http://www.mathworks.com/products/matlab/index.html

obtained until now are promising, but in the next period we must use the training data and to identify thresholds, that will allow us to say that a song is in our database or not.

Another big problem until now was related to scalability of our work, and from this reason we try to find suitable metrics that can be applied in real time. This problem will give us another future direction in our work related to combination of Euclidian metrics with scalability metrics.

Acknowledgements. The research presented in this paper was funded by the Sector Operational Program for Human Resources Development through the project “Development of the innovation capacity and increasing of the research impact through post-doctoral programs” POSDRU/89/1.5/S/49944.

References

1. The American Heritage Dictionary of the English Language, Fourth Edition, Houghton Mifflin Company, 2000, archived from the original on June 25, 2008, retrieved May 20, (2010)
2. Klapuri, A., Davy, M.: Signal processing methods for music transcription. Springer. p. 8. ISBN 9780387306674. (2006)
3. Plack, C.: Pitch: Neural Coding and Perception. Springer. ISBN 0387234721. (2005)
4. Don Michael, R. ed.: The Harvard Dictionary of Music (4 ed.). Harvard University Press. p. 499. ISBN 9780674011632. (2003)
5. Patterson, R., Gaudrain, E., Walters, T. C.: The Perception of Family and Register in Musical Tones. In Mari Riess Jones, Richard R. Fay, and Arthur N. Popper. Music Perception. Springer. pp. 37–38. ISBN 9781441961136. (2010)
6. Cooper, P.: Perspectives in Music Theory: An Historical-Analytical Approach. Pag. 16. ISBN 0-396-06752-2. (1973)
7. Whitall, A.: The Cambridge Introduction to Serialism (New York: Cambridge University Press, 2008): 276. ISBN 978-0-521-68200-8. (2008)
8. Bartschand, M. A., Wakefield, G. H.: Audio Thumbnailing of Popular Music Using Chroma-based Representations. IEEE Transactions on Multimedia. (1996)
9. Miotto, R., Orio, N.: A Music Identification System Based on Chroma Indexing and Statistical Modeling. International Symposium/Conference on Music Information Retrieval. Pp. 301-306. (2008)
10. Müller M., Kurthand, F., Clausen, M.: Audio Matching Via Chroma-based Statistical Features. Proceedings of the International Conference of Music Information Retrieval, London, UK. (2005)
11. Hu, N., Dannenberg, R. B., Tzanetakis, G.: Polyphonic Audio Matching and Alignment for Music Retrieval. Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. (2003)