A Music Similarity Function Based On Time-Frequency Analysis and Pyramid Kernel

Soe Myat Thu
University of Computer Studies,Yangon
thuthu052228@gmail.com

Abstract

In this paper, content-based music similarity function from acoustic music signal in an efficient way is considered. This function is to determine similarities among songs, particularly, a piece of input music signal compared with storage music song's signal into the database and retrieve a whole music song according to the input query. Representing the music signal having sparse nature is accomplished by Matching Pursuit with time-frequency dictionaries. In order to match a candidate segment with the query segment, the music signal similarity measure is performed by Spatial Pyramid Matching. Evaluation results on music similarity illustrate that our music similarity function is better than such previous approaches.

Keywords: Content-based music similarity, Matching Pursuit, Spatial Pyramid Matching.

1. Introduction

Music Similarity from audio signals is an interesting topic that receives a lot of attention these days. These feature sets are designed to reflect different aspects of music such as timbre, harmony, melody and rhythm. Individual sets of audio content and social context features have been shown to be useful for various MIR tasks such as classification, similarity, recommendation. Among them, similarity is crucial for the effectiveness of searching music information and the music segmentation.

Various representations have been proposed for musical signals features. The time-domain representation (waveform) and frequency-domain (STFT or spectrogram) representation are the very basic and most widely used. Logan and A. Saloman present an audio content-based method of estimating the timbral similarity of two pieces of music that has been successfully applied to playlist generation, artist identification and genre classification of music. The signature is formed by the clustering, with the K-means algorithm, of Mel-frequency Cepstral Coefficients (MFCCs) calculated for short frames of the audio signal [1]. Another content-based method of similarity estimation, also based on the calculation of MFCCs from the audio signal, is presented by Aucouturier and Patchet. A mixture of Gaussian distributions is trained on the MFCC vectors from each song and are compared by sampling the distributions (generating random points according to one distribution and estimating their likelihood based on the other distribution) in order to estimate the timbral similarity of two pieces [2]. B. Logan and A. Salomon's technique also forms a signature for each song based on K-means clustering of spectral features. For each song, this computes a signature based on K-means clustering of frames of MFCCs with audio sampled at 16kHz and divide this signal into frames of 25.6ms overlapped by 10ms [3]. M. Goto describe a method for obtaining a list of chorus (refrain) sections in compact-disc recordings of popular music and also tested on 100 songs of the popular-music database "RWC" Music Database [4]. Music used to show repetition and similarities on different levels, starting from consecutive bars to larger parts like

chorus and verse. Some authors have tried to take this into account and proposed methods operating on several temporal levels. Jehan constructed several hierarchically related similarity matrices [5]. Finally, D.Turnbull, G. L.E. Pampalk and M. Goto develop a set of difference features that indicate when there are changes in perceptual aspects (e.g., timbre, harmony, melody, rhythm) of the music. By combining these features and formulating the problem as a supervised learning problem using a publicly available data set. For all metrics, performance is averaged over the 100 songs in the data set [6].

2. A Content-Based Music Similarity Function

This content-based music system is represented music signal decomposition for music features with a dictionary and pattern similarity discovery with pyramid kernel level. For the challenges of matching a candidate segment with the query segment, the system could significantly improve similarity measure using Spatial Pyramid Matching. And the retrieval time could considerably improve using Matching Pursuit (MP) Method. Our particular approach to choose music song also makes it possible automatic retrieval using matching pursuit feature sets, for example for using the browsing system rapidly through a list of possible song of interest returned by a search engine. By guiding us to the most significant parts of a music song, it also allows the development of fast and efficient method for searching very large collections based purely on the audio content of the song, sidestepping the computational complexity of existing content-based search methods.

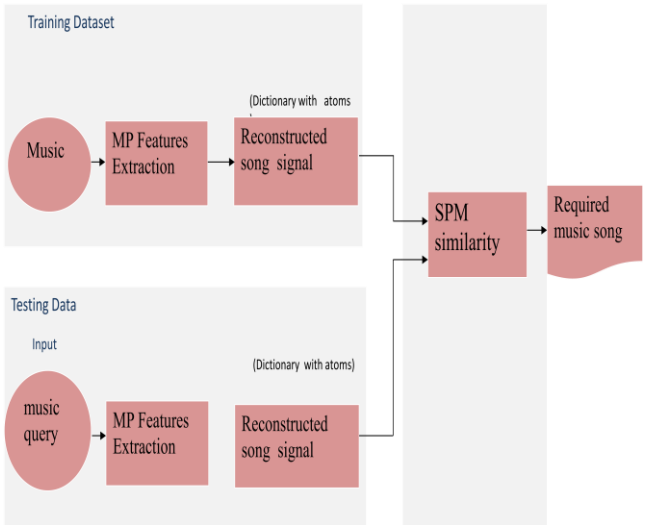


Figure 1. Overview of our content-based music similarity function

A block diagram of the system can be seen in Figure 1. This new system creates matching pursuit features from a query input music signal (10 second) using matching pursuit method. Music signal structural features of input music query are represented as a dictionary with most prominent atoms that match their time-frequency signatures. Metadata database contained music signal structural features sets of very large collections music. Pyramid match kernel measures similarity achieved from a partial matching between two feature sets. A whole music is achieved to determine similarity between input music query features sets and all music features sets contained in the metadata database. The optimal description of the music song from the meta database is found in respect to more similar function defined in Spatial Pyramid Matching method.

2.1 Matching Pursuit Features Extraction using Time-frequency Analysis

Time-Frequency representations are used to analyze or characterize signals whose energy distribution varies in time and frequency. Time-Frequency analysis studies a two dimensional signal function whose domain is the two

dimensional real plane, obtained from the signal via a time-frequency transform. A time-frequency representation describes the variation of spectral energy over time, much as a musical score describes the variation of musical pitch over time.

Matching Pursuit is part of a class of time-frequency signal analysis algorithms known as Atomic Decompositions. These algorithms consider a signal as a linear combination of known elementary pieces of signal, called atoms, chosen within a dictionary.

Matching Pursuit algorithm aims at finding sparse decompositions of signals over redundant bases of elementary waveforms [7].

2.1.1. Time-Frequency Dictionary Approximation

Matching pursuit decomposes music signal into a linear expansion of waveforms that are selected from a redundant dictionary of functions. Wavelet transforms should be designed as follow: **Dictionary:** A dictionary contains a collection of blocks plus the signal on which they operate. It can search across all the blocks (i.e., all the scales and all the bases) for the atom which brings the most energy to the analyzed signal. **Book:** A book is a collection of atoms. Summing all the atoms in a book gives a signal. **Atoms:** An elementary piece of signal. An atom is organized by its Gabor atoms.

Music signal decomposes into the blocks, and several blocks corresponding to various scales or various transforms can be concurrently applied to the same signal, thus providing multi-scales or multi-basis analysis. Then, this updates the correlation by applying the relevant correlation computation algorithm to the analyzed signal, and search the maximum correlation in the same loop. The atoms are created by corresponding to the maximum correlation with the signal and store this atom in the book. The created atom is substracted from the analyzed signal thus obtaining a residual signal, and re-iterate the analysis on this residual [8].

Using Matching Pursuit method is price of efficiency and convergence. Time compression is quite excellent by extracting prominent atoms (features). In order to achieve the required information in our system, the algorithm search the most strongly correlated with the original signal x for each iteration m. It has the maximum inner product $\widehat{w}_m$ with the signal. This MP algorithm uses the following steps:

1. initialization:
$$m = 0, x_{搜} = x_0 = x; \quad (1)$$

2. computation of the correlations between the signal and every atom in dictionary D, using inner products :
$$\forall_w \epsilon\, D: \quad Corr(x_m, w) = |< x_m, w >| \quad (2)$$

3. search of the most correlated atom, by searching for the maximum inner product:
$$\widehat{w}_m = \underset{w \in D}{argmax} Corr(x_m, w) \quad (3)$$

4. subtraction of the corresponding weighted atom $\infty_m \hat{w}_m$ from the signal :
$$x_{m+1} = x_m - \infty_m \widehat{w}_m \quad (4)$$
$where \ \propto_m =< x_m, \widehat{w}_m >;$

5. If the desired level of accuracy is reached, in terms of the number of extracted atoms or in terms of the energy ratio between the original signal and the current residual x_{m+1}, stop; otherwise, re-iterate the pursuit over the residual: $m \leftarrow m+1$ and go to step 2.

Music song signal analysis of our system is desirable to obtain sparse representations that are able to reflect the signal structures. The functions used for MP in our algorithm are Gabor function, i.e. Gaussian-windowed sinusoids. The Gabor function is evaluated at a range of frequencies covering the available spectrum, scaled in length (trading time resolution for frequency resolution), and translated in time. Each of the resulting functions is called an atom, and the set of atoms is a

dictionary which covers a range of time-frequency localization properties. The Gabor function in our new search model is defined as

$$g_{s,u,\omega,\vartheta}(t) = K_{s,u,\omega,\vartheta}\left(\frac{t-u}{s}\right)\cos[2\pi\omega(t-u)\theta] \quad (5)$$

where (s,u,ω,θ) denotes the parameters to the Gabor function, with s, u, ω, θ corresponding to an atom's position in scale, time, frequency and phase, respectively. We focus on a dictionary by choosing 3000 gabor atoms of length 11025 sample points. The advantages of gabor dictionary representation is characterized the signal time and frequency domain by resulting a few reconstruction error (signal to noise ratio is 10.99). The Gabor dictionary was implemented with the parameters of atoms chosen from dyadic sequences of integers [9].

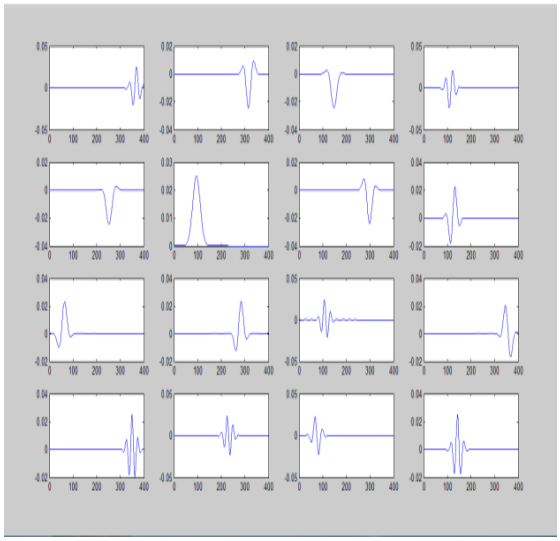
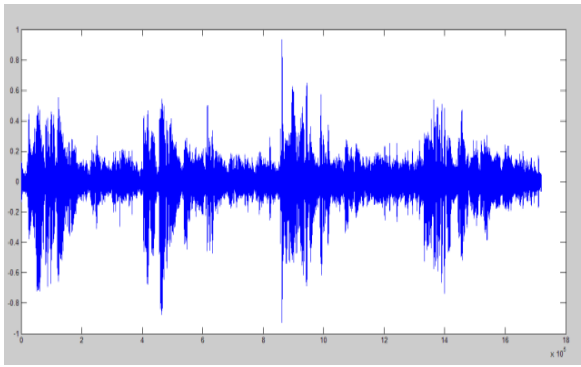
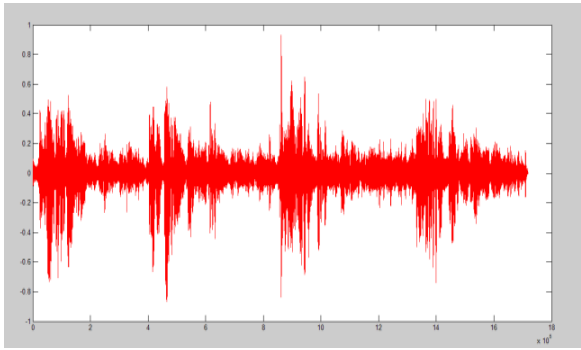


Figure 2. Decomposition of signal using MP with 16 gabor atoms

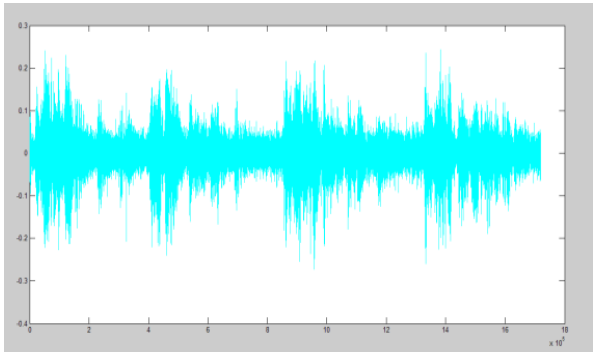
Flexible decompositions play an important role to represent signal components whose localizations in time and frequency vary widely. Signal components must be expanded into waveforms which are called time-frequency atoms. In figure 2, it is the decomposition of signal using Matching Pursuit method. These waveforms are called time-frequency atoms which are an example of sixteen Gabor atoms from a whole song.



(a) original song signal



(b) reconstructed song signal



(c) residual song signal

Figure 3. (a)original signal (b)Reconstruction of a music song using Gabor dictionary with MP features (c)residual song signal

A music song is reconstructed using Gabor atoms as a dictionary in Figure 3. In (a) and (b) original song signal is decomposed into MP features using matching pursuit method and the best atoms are selected to reconstruct a music song without distortion. In (c) the residual song signal after song reconstruction.

2.2 SPM based Similarity using Matching Pursuit Features

Spatial Pyramid Matching is to find an approximate correspondence features between two features sets step by step level. At each level of resolution, SPM works by placing a sequence of increasingly coarser grids over the features.

A pyramid matching pattern kernel allows for multi-resolution matching of two collections of features in a high-dimensional appearance space, however it discards all spatial information. Another problem is the quality of the approximation to the optimal partial match provided by the pyramid kernel degrades linearly with the dimension of the feature space. In our system, the approximate matching pattern discovery (SPM) is constructed the pyramid level and then the number of matches at each level is given by histogram intersection function. In determining SPM, SPM is used step by step level to improve matching musical data space and taking a weighted sum of the number of matches. At any fixed resolution, two feature points are said to match if they fall into the same cell of the grid. For matching pattern discovery, our system used histogram intersection function. Let X and Y are two features sets and construct a sequence of grids at each resolution. Each level has 2^l cells along each dimension(d), for a total of $D=2^{dl}$ cells. The histogram intersection function is as follows:

$$\mathcal{I}(H_X^l, H_Y^l) = \sum_{i=1}^{D} \min(H_X^l(i), H_Y^l(i)) \qquad (6)$$

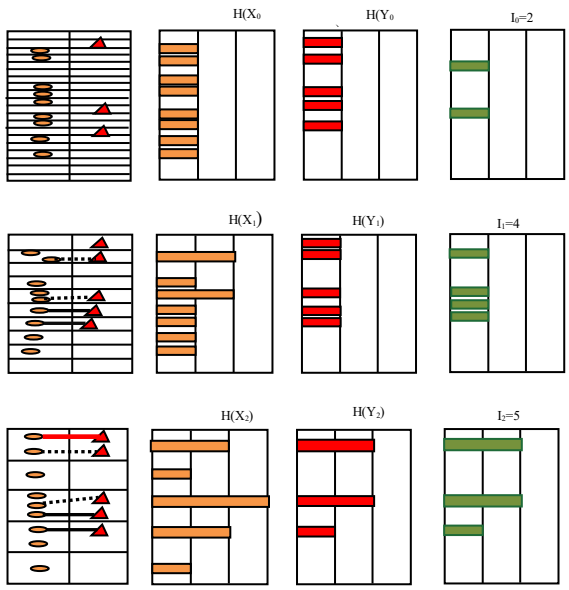
In the following, it will be abbreviated

$$\mathcal{I}(H_X^l, H_Y^l) \; to \; \mathcal{I}^l.$$

To achieve more definitely pattern matching, our system modified step by step level pyramid kernel function. The number of matches found at level 'l' also includes all the matches found at the finer level l+1. Therefore, the number of new matches found at level l is given $\mathcal{I}^l - \mathcal{I}^{l+1}$ for l=0,…….L-1. The weight associated with level l is set to $\frac{1}{2^{L-l}}$, which is inversely proportional to cell width at that level. The definition of a matching pyramid kernel is:

$$K^L(X,Y) = \mathcal{I}^L + \sum_{l=0}^{L-1} \frac{l}{2^{L-l}} (\mathcal{I}^l - \mathcal{I}^{l+1}) \qquad (7)$$

[10].



(a) Features sets (b) Histogram pyramids (c) Intersection points

Figure 4. (a)Features sets, (b)Histogram pyramids and (c)Intersection points.

A pyramid matching determines a partial correspondence by matching feature points once they fall into the same histogram bin as shown in Figure 4. In this example, two 1-D feature sets are used to form one histogram pyramid. Each row corresponds to a pyramid level. In (a), the set X is on the left side, and the set Y is on the right. (Features Points are distributed along the vertical axis, and these same points are repeated

at each level.) Bold dashed lines indicate a pair matched at this level, and bold black lines indicate a match already formed at a former resolution. In (b) multi-features histograms are shown, with bin counts along the horizontal axis. In (c) the intersection pyramid between the histograms in (b) are represented.

3. Evaluation Results

The performance of the new similarity system evaluates in simulations browsing the similar structure of a set popular music pieces. We performed to evaluate our system against based on Hierarchical Dirichlet Process (HDP). The HDP is a nonparametric Bayesian model and compute timbral similarity between recorded songs. Like the Gaussian Mixture Model (GMM), it represents each song as a mixture of some number of multivariate Gaussian distributions. This dataset consists of 121 songs from South by Southwest (SXSW) Dataset and 50 data pieces for testing. The Average Precision (AP) of HDP based similarity algorithm on large dataset is 40%. [11].

Our system has a ability to test on a collection of tracks including English (125 tracks) and Myanmar (75 tracks) music genres in the database. Our dataset contain 200 songs from the popular music songs such as jazz(14), metal(15), country(30), rock, punk, hip-hop songs, etc. These songs were varying lengths (3-5 minutes) and 44100 Hz sampling rate, 16 bits per sample with mono channel. All songs were trained on the different sets of features vectors for each song consisted of MP features. 3000 feature vectors were extracted from one second long. Features were calculated from a rectangular window length 16384 sample points with 50% overlap. Input music pieces (10 seconds) segment and a whole music song into the database makes up for training and testing. There are 200 music songs for training dataset and 200 data pieces are for testing dataset. For the retrieving required song, similarity matrix is calculated one second by one second divided by pyramid levels (three different levels).

For the performance evaluation, our system tested with Precision, Recall and F-measure which are standard metrics of music similarity quality. Evaluation results for our MP-based similarity function are given as Table 1 and Table 2.

Our MP-based Similarity System			
Number of songs	**Correct songs**	**Precision**	**Average-Precision**
5	**5**	**100%**	**100%**
50	**47**	**94%**	**96%**
100	**89**	**89%**	**93%**
150	**133**	**88%**	**91%**

Table 1. Performance measures of retrieval quality for our similarity system on small music dataset.

Our MP based Similarity system					
Number of songs	**Correct songs**	**Precision**	**Recall**	**F-measure**	**Average Precision**
200	**179**	**89%**	**90%**	**89%**	**90%**

Table 2. Performance measures of retrieval quality for our similarity system on large music dataset.

In designing the searching speed to be speed in our function, the proposed system improved by applying the threshold for the large dataset. The system calculates the number of matching features between the input query length and the candidate songs in the database. For sequential searching, musical-similarity matches in useful-size musical databases are to be unacceptably slow because of large storage data points. To address this problem, the system decides the required threshold value point. The best threshold we selected in our search system is 1.85. If the sum of matches in musical pieces is exceeded the threshold value, the search system

decides that is can find this query from the database without searching the other music in the database and returns the whole song. If the sum of the matches is below the threshold value, the system decides to find the whole database and then returns the required song from the database. The threshold value is defined from the calculation of statistical analysis.

4. Conclusion

In content-based music information system, music similarity for searching and browsing particular music songs in an efficient manner is still demanding. We demonstrate new similarity function for assessing the similarity songs from the large data collection. The system uses matching pursuit and spatial pyramid matching for determining significant features of music pieces and retrieving music songs in efficient way. Retrieving similar music pieces from a database is completed by matching the MP features space step by step level using spatial pyramid matching. Better accuracy on large collection of musical songs are achieved upon the whole architecture of the system.

References

[1] M. Levy, M. Sandler and M. Casey, "Extraction of high-Level Musical Structure from Audio Data its Application to Thumbnail Generation", *EPSRC grant GR/S84750/01.*

[2] J-J. Aucouturier and F. Pachet. Music similarity measures: What's the use? In Proceedings of ISMIR 2002 Third International Conference on Music Information Retrieval, September 2002.

[3] B. Logan and A. Salomon," A Content-Based Music Similarity Function", *Cambridge, Massachusetts 02142 USA.*

[4] *M. Goto, "*A Chorus-Section Detecting Method for Musical Audio Signals", *ICASSP Proceedings, pp.V-437-440, April, 2003.*

[5] T. Jehan." Hierarchical multi-class self similarities." *In Proc. of 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pages 311–314, New Platz, New York, USA, Oct. 2005.*

[6] D.Turnbull, G.L.E.Pampalk and M.Goto "A Supervised Approch For Detection Boundaries in Music using Difference Features and Boosting", *Austrian Computer Society (OCG),2007.*

[7] S.G. Mallat and Z.Zhang, "Matching Pursuits With Time-Frequency Dictionaries", *IEEE Transactions on Signal Processing, VOL.41,NO.12,December 1993.*

[8] S.Krstulovie and R. Gribonval, "The Matching Pursuit Tool Kit".

[9] S.Chu,S.Narayanan and C-C.Jay Kuo,"Environmental Sound Recognition using MP-Based Features", *University of Southern California,Los Angeles,CA 90089-2564.*

[10] S.Lazebnik and C.Schmid, "Spatial Pyramid Matching".

[11] M. Hoffman, D. Blei and P. Cook, "Content-Based Musical Similarity Computation Using The Hierarchical Dirichlet Process", *ISMIR –session 3a Content-Based Retrieval, Categorization and Similarity 1, 2008.*