2005－MPS－56（2）
2005／9／21

スペクトル特性に基づいた Query-by-Example による音楽検索モデル

Yi YU，渡辺知恵美，城和貴
奈良女子大学　大学院　人間文化研究科
〒630-8506　奈良市北魚屋西町，
Email: {yuyi, chiemi, joe}@ics.nara-wu.ac.jp

あらまし　近年，内容ベースの音楽検索における研究はますます多くの関心を引き付けている．適切な特徴セットを用いた類似検索アプローチにより，計算時間を減少させ検索速度を向上させることができる．本論文は音響ベースの音楽検索に対し以下の二点で貢献している：1. スペクトル特性を研究し，隣接しているフレームのスペクトル特徴が非常に関連していることを示している； 2. 基本的な音響特性の分析に基づき，スペクトル相関関係に焦点を合わせた高速で効率的な Query-by-Example による音楽検索モデルを提案している．また本提案手法の評価としてシミュレーション結果におけるスペクトル相関関係（SC）閾値，データ格納量，計算時間の分析を行い，単旋律および多声音楽における有効性を確認した．

キーワード　内容ベースの音楽検索，スペクトル相関関係, pre-filtering, 動的計画法.

Query-by-Example Music Retrieval Modeling based on Spectral Correlation

Yi YU, Chiemi WATANABE, Kazuki JOE
Graduate school of Humanity and Science, Nara Women's University,
Kitauoya nishi-machi, Nara 630-8506, Japan, Email: {yuyi, chiemi, joe}@ics.nara-wu.ac.jp

Abstract　Content based music retrieval is attracting more and more research interest. Suitable feature sets and similarity match approaches can help to reduce the tedious computation time and speed up the retrieval. This article mainly contributes in the two-fold to acoustic based music retrieval: 1. we report a study of the music spectral property and show that the spectral features of adjacent frames are highly correlated; 2. on the basic acoustic characteristics analysis we propose a fast and efficient Query-by-Example Music Retrieval modeling focused on spectral correlation. The extensive evaluations confirm the effectiveness of the proposed retrieval model for both monophonic and polyphonic music. The simulation results are analyzed with a theoretical approach that seeks to obtain the mathematical relation for our retrieval system parameters such as Spectral Correlation (SC) threshold, storage, and computation.

Keyword　Content based music retrieval, spectral correlation, pre-filtering, dynamic programming.

1. Introduction

Content-Based Music Retrieval (CBMR) in database systems is gradually becoming a popular topic, where the query music is matched against the reference melodies under a certain criterion of music similarity. CBMR in acoustic form is generally the most natural but difficult due to the high dimensionality of the features, complex computation, and large database size. To reduce the huge computation with almost no efficient indexing algorithms, many researchers have tried two ways: improving the CBMR by cutting down the dimensionality of the features and optimizing the sequence matching algorithms.

The “energy profile” is adopted as the feature in [1], and the spectrum-based minimum-distance is used to improve the accuracy; both of the feature sequences are compared by DP. Yang [2] also adopted the short time spectrum as the feature, except that only the signal of the local maximum is selected to calculate the feature. A variation of DP methods is used in feature comparison and the result is further refined with the linear filtering. The feature size of the spectrum [1,2] is usually very large, which requires much more storage and computation time. More recently Haitsma et al. constructed a cryptography hash function to classify pre-defined fingerprints of acoustic data in a database. A two-stage search algorithm is built on only performing full fingerprint comparisons at candidate positions pre-selected by a sub-fingerprint search. Harb [4] reported a query by example music retrieval system (QEMR) based on the local (1s) and global (10-20s) acoustic similarities. Additionally, this system presents music similarity evaluation at a high-level such as genre classification.

Despite various approaches that have been carried out on CBMR, little work has been done on analyzing music signal property. We also focus on the spectral similarity measurement between the acoustic query and the acoustic database, especially considering polyphonic music melodies. Unlike the existing methods, we emphasize the analysis of the music property and the spectral correlation of the music signal and remove the spectral redundancy by merging the adjacent similar frames. In order to further improve the retrieval speed we use the low order MFCC to pre-filter the reference melodies meanwhile maintaining the high retrieval ratio.

2. Music Spectral Analysis and Feature Selection

Dissimilar to other kinds of audio, music has strong descriptive composition—score, which implies the music theme and is composed of notes. For the simplicity of description, we define NoteSet as the simultaneously initiated notes, either a single note, or the combination of two or more notes.

Figure 1 is an example dealing with different aspects of music, from score to spectrum. The music piece is taken from Chinese folk, Alamuhan. Here a NoteSet contains one note. The analysis is similar when a NoteSet consists of multiple notes. Fig.1(a) presents the score of Alamuhan and Fig.1(b) is the corresponding energy profile calculated from the original waveform signal. Fig.1(c) shows the frames for feature analysis and extraction. Within a NoteSet the music has a long-term stable spectrum structure during which the spectrum of adjacent frames are highly correlated; then it transits to the next NoteSet and during the transition it experiences an short-term unstable spectral structure. The variable duration of a NoteSet in different performances usually lead to tempo variations problem, and in results requires complex matching techniques.

The procedure from Fig.2(c) to Fig.2(d) is to merge the features by removing the redundant frames according to the SC. In the ideal cases, only one frame of the stable state is necessary to represent the NoteSet; other frames are redundant and can be removed. In this way, only a small percentage of the frames remain; both the storage and computation are reduced. In addition, much of the tempo variation is removed, thus complex DP algorithms are unnecessary, and a simpler algorithm can be used for the feature sequence match.

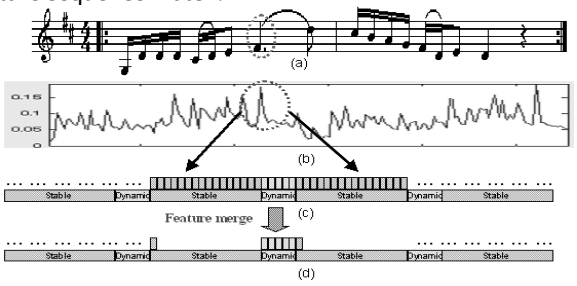

Fig.1 Music signal property.

When spectral similarity is adopted as the criterion for CBMR, the spectral profile is the most important. A popular parametric spectrum is cepstrum and MFCC has special properties suitable for CBMR. MFCC is insusceptible to pitch shift and requires less storage compared with STFT. MFCC not only concisely represents the spectral structure it also has a special property due to its resolution capability. The low order MFCC is the low frequency components of the DCT, and reflects the basic spectral profile, which roughly stands for the music score; the high order MFCC is the high frequency component of the DCT, and reflects more details of the energy distribution in the different Mel bands. In other words, the first few MFCC coefficients gives an outline of spectrum; the rest of high order MFCC coefficients increases the accuracy of the spectrum. Based on this property, the retrieval can be further accelerated with two-step retrieval by a pre-filtering method.

3. SC-based Music Retrieval Model

Given the above discussions, we present a fast and efficient music retrieval model. In our model, we defined the following parameters:

- S_i: STFT of the i[th] frame.
- M_i: MFCC of the i[th] frame.
- $\rho_{i,j}$: correlation between S_i and S_j.
- ρ_{th}: spectral correlation threshold.
- $\delta(\rho_{th})$: the compression ratio of the frames when the SC threshold is set to ρ_{th}.
- L: MFCC order.
- L_1: number of low order MFCC used for pre-filtering in the first step in Fig.2.
- S: survive rate of the retrieval, the ratio between query output and all the references in the database.
- S_1: survive rate of the pre-filtering.
- S_2: survive rate of the second step retrieval.

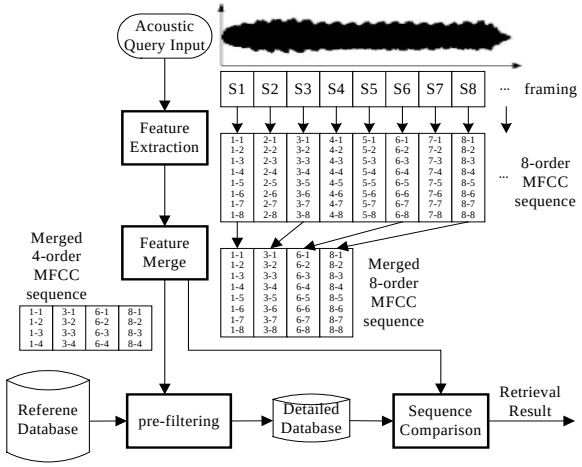

Fig.2 Acoustic music retrieval model.

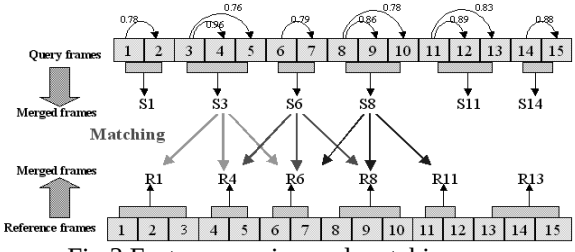

Fig.3 Feature merging and matching.

Our music retrieval model is shown in Fig.2. It contains the following modules:

Feature extraction: The query music is re-sampled and framed. For each frame the short time-spectrum S_i is calculated. Then the SC $\rho_{i,j}$ and MFCC feature M_i are computed.

Frame merge: Out of the continuous frames with

an SC bigger than a certain threshold, for example $\rho_{th} = 0.7$, only the first frame is kept. Fig.3 gives an example of frame merge. For the query music, frame 1 and 2 are spectral similar, so the two frames are merged to one frame s1. By merging the neighboring spectral similar frames, only few of the frames remain, which depends on the SC threshold ρ_{th}. The percentage of the remaining frames is $\delta(\rho_{th})$.

Pre-filtering: in the first step, only the first L_1-order MFCC of the query is used to roughly choose some candidates, removing most of the unlikely references in the music database. Merely a small percentage S_1 of reference music survives as the candidates, and constructs a new database. In the second step, with all the L-order MFCC coefficients, a percentage S_2 of target music is obtained from the small database.

Feature sequence match: the merged query sequence is matched against the merge one of the reference melody at different time shifts. At a specific time shift, each frame in the query is compared with several neighboring frames in the reference music so as to consider the remaining time variation effect.

4. Retrieval Model Analysis

By merging the adjacent spectral similar frames, the storage is reduced to the percentage of $\delta(\rho_{th})$. The other advantage of frame merge is that most of the tempo variation is mitigated. In Fig.3 the query and reference music have different timing where the same notes (for example, 3rd, 4th and 5th frames in the query and 4th and 5th frames in the reference) have different length. However, by merging frames, the redundant information is removed. If the two sequences have the same score, it is reasonable that the merged sequences almost have the same timing, that is, the tempo feature of the merged query is nearly the same as that in the merged reference music.

Merging the similar frames decreases the computation requirement. Assume that the average number of frames of all the references is R, and that of query is Q. The average computation for DP, C_{DP}, is given in Eq.1. By merging frames, the average number of remaining frames for the query and references is $\delta(\rho_{th})\cdot R$ and $\delta(\rho_{th})\cdot Q$ respectively. The corresponding computation, C_{FM}, is given in Eq.2. With the two-step retrieval, in the pre-filtering stage, L_1 MFCC coefficients are used, and S_1 percentage of references survive; in the second stage, the surviving references are searched with L MFCC coefficients to get the best targets. The total computation of the two stages, C_{PF}, is given in Eq.3

$$C_{DP} = R \cdot Q \cdot L \quad (1)$$

$$C_{FM} = [\delta(\rho_{th})\cdot R]\cdot[\delta(\rho_{th})\cdot Q]\cdot L \quad (2)$$

$$\begin{aligned} C_{PF} &= [\delta(\rho_{th})\cdot R]\cdot[\delta(\rho_{th})\cdot Q]\cdot L_1 \\ &+ [\delta(\rho_{th})\cdot R\cdot S_1]\cdot[\delta(\rho_{th})\cdot Q]\cdot L \\ &= [\delta(\rho_{th})\cdot R]\cdot[\delta(\rho_{th})\cdot Q]\cdot[L_1/L + S_1] \end{aligned} \quad (3)$$

Due to pre-filtering, the total computation in Eq.3 is decreased by a factor $F_{PF} = L_1/L + S_1$ compared with Eq.2. Reducing either L_1 or S_1 can decease the computation.

5. Experiments and Results

The experiments are conducted on the acoustic database with both monophonic and polyphonic melodies. Our music database consists of 166 melody pieces and is generated from Chinese folks (44 pieces from 12-Chinese-girl band) and western instruments sound (122 pieces performed by different instruments). Each piece is segmented into 60-second-long melodic slip. Query melodic samples are segmented into 6-8 seconds long. However, the query samples are the different versions of the reference database, that is, query music and reference melodies may have different tempo and pitch shift. In the simulation, generally $L = 8$, $L_1 = 4$, $S_1 = 0.2$, and $\rho_{th} = 0.7$ except especially pointed out.

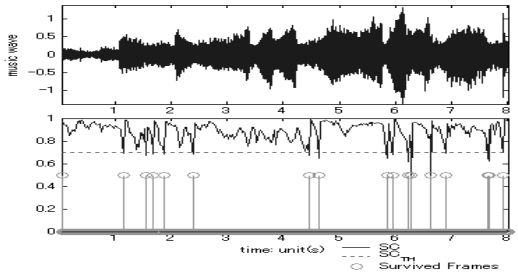

Fig.4 Spectral correlation and frame merge, $\rho_{th} = 0.7$ (No Word melody from 12-girl band).

Figure 4 shows the SC (ρ) of the frames for a relatively simple melody, where most of the adjacent frames have a bigger SC than the SC threshold. Out of the total 344 frames, only 4.9%, 17 frames are kept. When music score becomes more complex and the spectral features changes frequently, more frames are kept. However, still most of frames are merged.

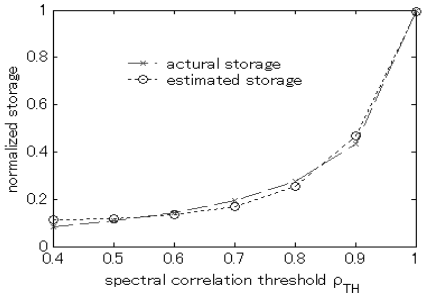

Fig.5 Normalized feature storage.

Figure 5 shows the normalized feature storage $\delta(\rho_{th})$ with respect to SC threshold ρ_{th}. When $\rho_{th} = 0.4$, $\delta(\rho_{th})$ is less than 10%. $\delta(\rho_{th})$ mono-

tonically increases as ρ_{th} does. When ρ_{th} reaches 1.0, no frames are merged and $\delta(\rho_{th})$ equals 1. From the experiment, we approximate the relation between the storage and the SC threshold ρ_{th} as below:

$$\delta(\rho_{th}) = 0.0001e^{9\rho_{th}} + 0.1085 \quad (0 < \rho_{th} < 1) \qquad (4)$$

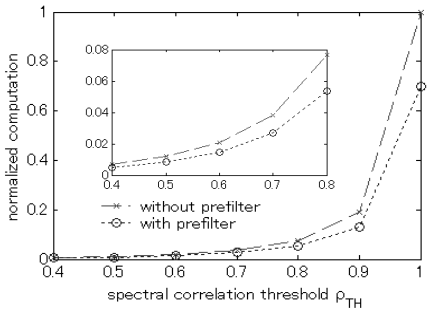

Fig.6 Normalized average computation.

Figure 6 shows the computation with respect to ρ_{th}. The upper curve stands for the computation with only frame merge, stated in Eq.2; and the bottom curve is the computation with both frame merge and pre-filtering, stated in Eq.3. The computations are normalized by the one given in Eq.1. When calculating Eq.2 and Eq.3, $\delta(\rho_{th})$ is the experiment result described in Fig.6. The computation in both Eq.2 and Eq.3 is proportional to $[\delta(\rho_{th})]^2$, so the computation reduction is very efficient. At $\rho_{th} = 0.7$, $\delta(\rho_{th}) = 0.195$, and the computation is reduced to 0.038, nearly 1/25 of the original computation.

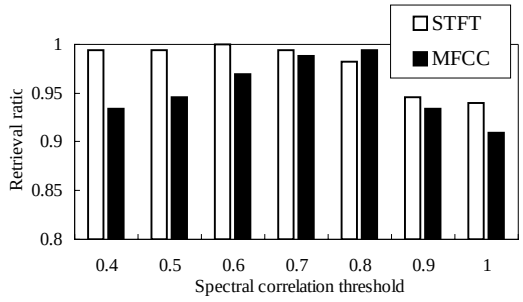

Fig.7 Top-4 retrieval ratio of STFT and MFCC with respect to ρ_{th} (without pre-filtering).

Though the pre-filtering method is only applicable to MFCC, the frame merge technique is suitable for both STFT and MFCC. Figure 7 verifies the top-4 retrieval ratio of STFT and MFCC with respect to different ρ_{th}. The pre-filtering method is not used. It is obvious that both STFT and MFCC work fairly well when ρ_{th} is in [0.6, 0.8].

Figure 8 gives the retrieval ratio for both top-4 retrieval and top-1 retrieval under different ρ_{th}. For most of the cases, the retrieval ratio with pre-filtering is almost the same as that without pre-filtering for both top-4 and top-1 retrieval. When ρ_{th} is smaller than 0.7, the retrieval ratio increases as ρ_{th} does. The retrieval ratio is almost unchanged when ρ_{th} is within [0.7,0.8]. As ρ_{th} increases further, the retrieval ratio for both top-1 and top-4 retrieval decreases. This is due to the fact that as ρ_{th} gets very big, most of the feature frames are kept, and the simple feature match method adopted in our proposal can not deal well with the remaining time variation. Under all cases, top-1 retrieval result is worse than top-4 result, however, it still gives good result when ρ_{th} is within [0.7, 0.8].

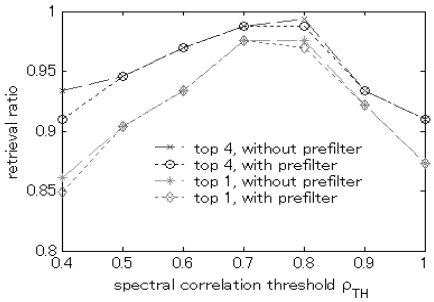

Fig.8 Top-4 and Top-1 retrieval ratio of MFCC with respect to ρ_{th} ($S_1 = 0.2$, $L_1 = 4$).

From Fig.5-8, both the storage and the computation increase as ρ_{th} does. The retrieval ratio reaches its maximum when ρ_{th} lies within [0.7, 0.8]. When the SC threshold ρ_{th} is set to 0.7, the pre-filtering MFCC order L_1 is set to 4, and the surviving ratio is 0.2, the system can achieve almost the highest retrieval ratio with little storage and computation.

6. Conclusion

In the article we have analyzed music signal property, explained the reason of tempo variation, and argued that the adjacent frames tend to have strong spectral correlations. On these basis, we have proposed a Query-by-Example music retrieval model, which has shown the following merits: (1) it removes the spectral redundancy and in turn reduces the feature storage of the reference music in the database; (2) both the query and the reference melodies have a short feature sequence, which improves the retrieval speed; (3) most of the tempo variation is removed, thus a simple feature sequence match method can be used; (4) relying on the characteristic of MFCC, two-step retrieval further speeds up the whole retrieval.

References

[1] J.Foote, "ARTHUR: Retrieving Orchestral Music by Long-Term Structure", ISMIR, 2000.

[2] C.Yang, "Music Database Retrieval Based on Spectral Similarity", ISMIR, 2001

[3] J.Haitsma and T.Kalker, "A Highly Robost Audio Fingerprinting System", Proc. of the Third International Conference on Music Information Retrieval, pp.107-115, 2002.

[4] H.Harb and L.Chen, "A Query by example music retrieval algorithm", proceedigs of the 4th European Worshop on Image Analysis for Multimedia interactive Services, pp.122-128, 2003