Fast Music Retrieval using Spectrum and Power Information

Tomoya Narita, Masahide Sugiyama

Graduate School of Computer Science and Engineering
University of Aizu, Japan
m5041122@u-aizu.ac.jp

Abstract

This paper proposes and evaluates algorithms for fast music retrieval. The target of this paper is to retrieve music segments (query in retrieval) from music database. The algorithms retrieve music segments from music database by distance of spectrum and difference of power. For reduction of calculation, the pruning method is proposed. The experiment is retrieving ten seconds segment from 100 music database. The experiment results shows a detection rate is 94.30% and retrieval processing time is 14.48 seconds at SNR$= -6.39$dB.

1. Introduction

Multimedia database management and retrieval are in high demand. If music retrieval is possible and fast, a music retrieval engine on the internet can be constructed and a database can be managed easily. Minami et al. showed indexing of video data using music and voice [1].

Several algorithms for fast music retrieval were proposed and evaluated[2, 3]. The algorithms retrieve music segments from music database by distance of spectrum and difference of power.

Kashino et al. presented a Audio Active Search using a Histogram of feature vectors [4]. The Audio Active Search can retrieve and detect a fifteen seconds segment (CM) from about six hour database (TV programs) in one second. The search does not hold time axis information because of Histogram, however the retrieval proposed in this paper holds. The database for experiment in this paper is more noisy than theirs.

Hashiguchi et al. presented a method to retrieve music data by using hamming or singing query [5], however this method is not for fast retrieval, and only small experiments are reported.

This paper shows the algorithms for retrieval of music, new method for reduction of calculation amount, and new experimental results. Section 2, the database for experiment is shown: section 3, the methods for detecting the location of segment and the pruning method for reduction of calculation amount are proposed: Last section explains an experiment of a retrieving segments from music database and discusses its evaluation.

2. Music database

2.1. Feature and database structure

At the music retrieval, the key to the high detection rate is the feature [6] and the key to the retrieval speed is database structure. The following list shows popular features and database structures.

1. Features
   - Music Feature
     (pitch [5] / speed(tempo) / beat [7] / harmony)
   - Audio Feature
     (frequency [1], [2], [4] / signal wave)
   - Meaning Feature (genre (classic/pops/...))
   - Instruments
2. Database Structures
   - Score
   - MIDI
   - extracted feature [1], [2], [4], [5]
   - Music (signal wave data) [5]

In this paper, the feature is frequency in audio feature and the database structure is extracted features for reduction of calculation amount at retrieving.

2.2. Database for experiment

One hundred music database has been constructed. Database has 30 “instrumental” musics, 25 “vocal” musics, and 45 “instrument & vocal” musics for general experiment on various type of music. The maximum length of music is 593 seconds, the minimum length is 52 seconds, and average length is 268 seconds.

3 database sets are made and shown in Table 1. Their recording conditions are different from each other’s. Signal to Noise Ratio (SNR) is used for the measure of noise level.

$$\text{SNR} = \frac{\sum x_l^2}{\sum e_l^2}, \tag{1}$$

$$e_l = x_l - y_l, \tag{2}$$

where, x_l is l-th signal of x, x_l^2 is l-th power of x, y is the signal made by adding noise to x. Generally speaking, for example, if a driver drive a car by 90km/h and record the driver’s voice by a microphone on a dashboard, SNR is -5dB and speech recognition is very difficult at the condition[10]. So the retrieval using Set 2 and Set 3 is very difficult task.

Table 1: 3 database sets

Set	Recording Condition	SNR (vs Set 1)
1	Digital transferred from CDs	-
2	Analog transferred from CDs	-6.39 dB
3	Analog transferred from Audio Tapes	-3.85 dB

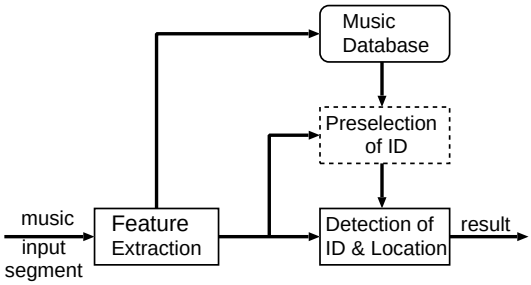

Figure 1: Procedure of Retrieving

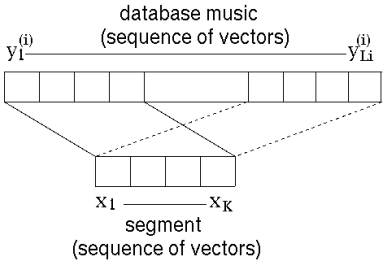

Figure 2: Vector Segment Matching

3. Retrieval algorithms

There are 2 requests in music retrieval.

1. Detection of music ID
2. Detection of location

In this paper, both requests are achieved and evaluated. Figure 1 shows the procedure of retrieving. At first, the algorithm preselects music IDs; after that, detects the music ID and its location. The algorithm for detecting music IDs and locations are only proposed in this paper.

3.1. Matching methods using spectrum

For detecting the music ID and location, vector segment matching (VSM) was proposed. Figure 2 shows the detail of VSM.

Here, $\boldsymbol{y}_l^{(i)}$ is l-th vector of i-th music, $\boldsymbol{x}_k$ is k-th vectors of input segment, $L_i(\approx 40000)$ is the number of frames in music, K is number of frame in input segment, I is number of music, and $d(\boldsymbol{y}_l^{(i)}, \boldsymbol{x}_k)$ is distance between $\boldsymbol{y}_l^{(i)}$ and $\boldsymbol{x}_k$. If the distance is close to zero, two segments are similar.

$$l, i = \arg \min_{\substack{1 \le i \le I \\ 0 \le l \le L_i}} \sum_{k=1}^{K} d^2(\boldsymbol{y}_{l+k}^{(i)}, \boldsymbol{x}_k) \tag{3}$$

In order to reduce calculation amount, following method, VQ-VSM, was proposed. Figure 3 shows the detail of VQ-VSM.

At first, this method extracts the feature of music data and makes VQ codebooks for each music; next, sequences of vectors of music on database are quantized by their own VQ codebooks so sequences of vectors is translated to sequences of representatives; after that, distance tables are generated. Distance

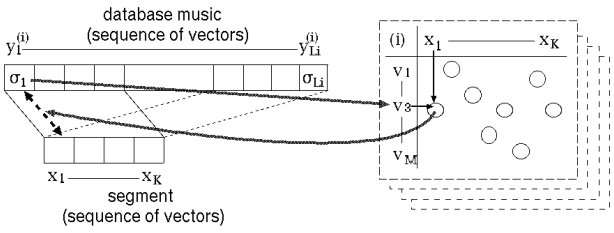

Figure 3: VQ-VSM

tables have the elements that is distance between representatives and vectors of input segment; finally, at calculating the distortion by VSM, uses the distances in distance tables.

Music database need not to have sequences of vectors because music is represented by sequences of indices of representatives.

Here, $\sigma_l^{(i)}$ is l-th pointer of representatives of i-th music, $\boldsymbol{V}^{(i)} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_M\}$ is VQ codebook of i-th music, and $d(m, k)$ is distance between representative $\boldsymbol{v}_m$ and $\boldsymbol{x}_k$.

$$l, i = \arg \min_{\substack{1 \le i \le I \\ 0 \le l \le L_i}} \sum_{k=1}^{K} d^2(\sigma_{l+k}^{(i)}, k) \tag{4}$$

For Normalization of acoustic distortion, Cepstrum Mean Subtraction[8] is used in Eq. (4).

Eq. (5) reduce the calculation amount in Eq. (4).

$$l, i = \arg \min_{\substack{1 \le i \le I \\ 0 \le l \le L_i}} \sum_{k=1}^{K/s} d^2(\sigma_{l+ks}^{(i)}, ks) \tag{5}$$

Here, s is skip value.

3.2. Matching methods using power information

VSM and VQ-VSM are the retrieval methods using spectrum pattern. On the other side, it is well known that the power information is effective in speech recognition, so the retrieval method using power information was proposed.

Here, p_n is n-th power, $\delta p_n = p_n - p_{n-1}$ is the difference of power p_n, and $d(\delta p, \delta q)$ is distance between δp and δq. Segment is retrieved by detecting l and i satisfied the following equation.

$$l, i = \arg \min_{\substack{1 \le i \le I \\ 1 \le l \le L_i}} \sum_{k=1}^{K/s} d^2(\delta q_{l+ks}^{(i)}, \delta p_{ks}) \tag{6}$$

Here, $q_l^{(i)}$ is l-th power of i-th music, and p_k is k-th power of input segment. This method is referred as "δPSM".

VQ-VSM-δPSM is the method which combined Eq. (6) and Eq. (5).

$$D_l^{(i)} = \sum_{k=1}^{K/s} \Big(d^2(\sigma_{l+ks}^{(i)}, ks) + w \times d^2(\delta q_{l+ks}^{(i)}, \delta p_{ks}) \Big) \tag{8}$$

$$l, i = \arg \min_{\substack{1 \le i \le I \\ 1 \le l \le L_i}} D_l^{(i)} \tag{9}$$

Here, $w(\ge 0)$ is weight. In Eq. (9), $w = 0$ is equal to VQ-VSM.

Table 3: Relationship of Methods and Accumulate Detection Rate (Top 10) (%):Segment = Set 2

Method	VQ-VSM	δPSM	VQ-VSM-δPSM			
Feature	Mel Cepstrum	δc_0	Mel Cepstrum & δc_0 ($w = 120$)			
Skip	$s = 8$	$s = 8$	$s = 8$	$s = 16$	$s = 22$	$s = 16$
Pruning	No use	No use	No use	No use	No use	Use
Inst.	90.66	78.66	93.66	88.66	83.66	86.66
Vocal	85.60	92.80	96.80	94.40	91.60	94.00
Inst. & Vocal	97.77	99.77	99.77	99.55	99.11	99.55
Avg.	**92.60**	91.70	97.20	**95.00**	92.60	**94.30**
Time	$\mathbf{25.17}sec$	14.62sec	44.56sec	$\mathbf{17.85}sec$	7.98sec	$\mathbf{14.48}sec$

Table 4: Relationship of Methods and Accumulate Detection Rate (Top 10) (%):Segment = Set 3

Method	VQ-VSM	δPSM	VQ-VSM-δPSM			
Feature	Mel Cepstrum	δc_0	Mel Cepstrum & δc_0 ($w = 120$)			
Skip	$s = 8$	$s = 8$	$s = 8$	$s = 16$	$s = 22$	$s = 16$
Pruning	No use	No use	No use	No use	No use	Use
Inst.	81.33	66.66	84.66	80.33	73.00	73.00
Vocal	71.20	67.20	82.80	74.44	69.20	73.20
Inst. & Vocal	95.11	93.77	95.55	94.44	94.22	86.44
Avg.	**85.00**	79.00	89.10	**85.20**	81.60	**79.10**
Time	-	-	-	-	-	$\mathbf{12.80}sec$

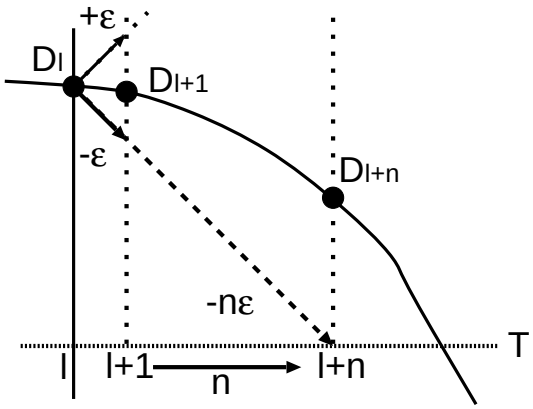

Figure 4: Concept of Pruning Method

3.3. Pruning Methods

For more reduction of calculation amount, pruning method is proposed. Figure 4 shows the concept of pruning method. Here, n is possible skip width by pruning, and T is threshold for detection that detects locations when $D_l^{(i)}$ is less than T.

Suppose $D_l^{(i)} > T$. If ϵ satisfies for all arbitrary l and i

$$\mid D_l^{(i)} - D_{l+1}^{(i)} \mid < \epsilon, \tag{10}$$

$D_{l+n}^{(i)}$ satisfies the following iunequality.

$$D_l^{(i)} - n\epsilon < D_{l+n}^{(i)} \tag{11}$$

Table 2: Analysis Conditions

Sampling Rate	16,000Hz
Bit	16 bit
Window Length	32ms (512 point)
Frame Shift	8ms (128 point)
cepstrum analysis order	16th
window function	Hamming window
Pre emphasis	$(1 - 0.97z^{-1})$
frequency warping parameter(mel)	$\alpha = 0.41$ [9]
codebook size	256 [3]

When the following inequality holds,

$$T \leq D_l^{(i)} - n\epsilon < D_{l+n}^{(i)} \tag{12}$$

$D_{l+n}^{(i)}$ is automatically greater than T because of the inequality Eq. (12). Therefore, skip width n is derived as follows:

$$n \leq \frac{D_l^{(i)} - T}{\epsilon} \tag{13}$$

4. Experiment

For experiment, Set 1 is used for database, and Set 2 and Set 3 is used for segments. Length of segments is 10 seconds, and 10 segments are made from each music so total number of segments is 1000. Analysis conditions are shown in Table 2.

Mel (FFT) Cepstrum is chosen for the feature in frequency features[3]. Mel cepstrum $\boldsymbol{c}$ is the feature that adapt to human hearing characteristics by expanding and contracting frequency

Table 5: Number of use of pruning method and shift frame (at Table 3, use pruning)

	N_1/L	N_2/L	Theoretical Time
No Pruning	0%	0%	$(17.85sec)$
Max(n)	16.49%	83.07%	$3.31sec$
Min(n)	12.73%	14.54%	$15.25sec$
Avg	20.28%	24.63%	$13.45sec$

* The time of No Pruning is the actual time at experiment.

axis, so 0-th of mel cepstrum is emphasized the power of low frequency.

The result of experiment is shown in Table 3 and 4. Time is retrieval time that includes sorting routine for accumulated detection. Table 3 shows the result of experiment using segments made from Set 2, and Table 4 shows the result of experiment using segments made from Set 3. Retrieval time of experiment of Table 4 using the methods without pruning seems to be same as Table 3 because the calculation amount is same so shows only with pruning.

Two values ϵ and T of the pruning methods is set by the preliminary experiment using same segments of the experiment. ϵ are set for each musics by the following equation.

$$\epsilon^{(i)} = \arg \max_{\substack{1 \le m \le 10 \\ 0 \le l \le L_i}} \mid D_l^{(i,m)} - D_{l+1}^{(i,m)} \mid, \qquad (14)$$

where, $D_l^{(i,m)}$ is $D_l^{(i)}$ using m-th segment in ten segments for i-th music. However, $\epsilon^{(i)}$ was too large set by Eq. (14) in preliminary experiments, so set by the following Eq. (15).

$$\epsilon^{(i)} = \arg \max_{1 \le m \le 10} \mid D_{l'}^{(i,m)} - D_{l'+1}^{(i,m)} \mid, \qquad (15)$$

here, l' is the correct position.

Initial T is 5.085 at Table 3 and 8.634 at Table 4. The detection rate is over 95% if VQ-VSM-δPSM without pruning is applied with the threshold T set to above value. After that, T is the lowest value in accumulate order, therefore T is changing dynamically.

Table 3 shows that the detection rate using VQ-VSM is 92.60%, δPSM is 91.70%, VQ-VSM-δPSM without pruning is 95.00%, and using VQ-VSM-δPSM with pruning is 94.30%. The retrieval processing time using VQ-VSM-δPSM without pruning is 17.85 seconds and VQ-VSM-δPSM with pruning is 14.48 seconds. Table 5 shows the details of pruning in experiments. Here, L is the sum of frames in all musics, N_1 is the number of times that pruning is carried out (when D is under the threshold T) and N_2 is the sum of skipped frames. 100% means covering all frames. The average of N_2/L is 24.63% and this means the proposed pruning methods skips only 24.63% of all musics. Therefor, the retrieval time without pruning is 17.85 seconds, then derived retrieval time with pruning is 13.45 seconds. However, actual retrieval time is $14.48sec$. The reason is that $\epsilon^{(i)}$ is so various that $\epsilon^{(i)}$ don't match for all segments of i-th music. Table 6 shows the examples of the $\epsilon^{(i)}$.

Table 4 shows that the detection rate using VQ-VSM-δPSM without pruning is 85.20% and the detection accuracy is lower for the different recording condition combination. The detection rate using VQ-VSM-δPSM with pruning is 79.10% and retrieval time is 12.80 seconds.

Table 6: Values of $(D_{l'}^{(i)} - D_{l'+1}^{(i)})$

i	MAX	MIN	AVG
013	16.3890	0.0117	4.1366
046	0.7865	0.2961	0.5556

5. Conclusion

In this paper, the retrieval methods that retrieve segments from music database and pruning methods for fast retrieval have been proposed. Detection Rate is 94.30% in the experiment using database made from Set 3 and Set 2 segments made from Set 2. Pruning method reduce retrieval time by 3.37 seconds.

Future works include retrieval speed, establishment of the way to set ϵ and T, and improvement of the detection rate at using Set 3 segment.

6. References

[1] K. Minami, et al., "Video Handling with Music and Speech Detection," Special Issue on Multimedia and Music in IEEE Multimedia, pp. 17–25, Jul. 1998.

[2] T. Narita, M. Sugiyama, "Music Retrieval using Power Information," Proc. of ASJ, 3-8-4, pp. 145–146, Mar. 2001 (in Jpn).

[3] T. Narita, M. Sugiyama, "Study on Fast Music Retrieval - Evaluation of Normalization of acoustic distortion and Retrieving Efficiency -," Technical Report of ASJ, H2000-100, pp. 7–14, Dec. 2000 (in Jpn).

[4] K. Kashino, G. A. Smith, H. Murase, "Quick Audio Retrieval Based on Histogram Feature Sequences," J. Acoust. Soc. Jpn. (E) 21, 4, pp. 217–219, 2000.

[5] H. Hashiguchi, et al., "Model driven path CDP for music retrieval with hamming query," Technical Report of IEICE, PRMU2000-66, pp. 35–40, Sep. 2000 (in Jpn).

[6] Y. Wang, et al., "Multimedia Content Analysis Using Both Audio and Visual Clues," IEEE Signal Processing Magazine, pp. 12–36, Nov. 2000.

[7] M. Goto, Y. Muraoka, "Parallel Implementation of a Beat Tracking System - Real-time Musical Information Processing on AP1000 -," Transactions of Information Processing Society of Japan, Vol. 37, No. 7, pp. 1460–1468, Jul. 1996 (in Jpn).

[8] Rahim, et al., "Signal Bias Removal for Robust Telephone Based Speech Recognition in Adverse Environments," Proc. of ICASSP94, pp.I-445-I-448, 1994.

[9] S. Imai, Speech Recognition, Kyouritsu Shuppan, 1995 (in Jpn).

[10] K. Shikano, et al., Speech Digital Signal Processing, Shokodo, 1997 (in Jpn).