# An Efficient Tree-based Quantization for Content Based Music Retrieval System

YUK YING CHUNG, *ERIC H.C.CHOI, ZHEN ZHAO,
MOHD AFIZI MOHD SHUKRAN, *DAVID YU SHI, *FANG CHEN
School of Information Technologies, University of Sydney, NSW 2006, AUSTRALIA
* ATP Research Laboratory, National ICT for Australia, NSW 1430, AUSTRALIA

*Abstract:* - In this paper, we have proposed and implemented a new music retrieval system based on content of music wave files. We have investigated different quantization methods by constructing them into the music data histograms as the feature vectors for the music files. There are three important aspects that will affect implementation of the system: audio feature extraction, quantization and distance computation. The proposed new system can allow the users to search the waveform data based on the query music samples. Without converting the audio wave file into a MIDI format, histograms are generated from the query data by vector quantization directly. By using the vector quantization, we can achieve the high accuracy in audio retrieval rate. In the proposed CBMR system, the use of 128 clusters in Kmeans-clustering quantization algorithm can achieve 87% retrieval accuracy and 90% high retrieval accuracy rate with the tree-based quantization.

*Key-Words:* - content based audio retrieval, K-means clustering, Tree-based quantization, MFCC

## 1 Introduction

The traditional information retrieval technology is based on text. Yahoo and Google are two typical text-based web search engines. The classic text-based information retrieval problem is to locate the desired text documents using a search query consisting of a number of keywords. Typically, matching documents are found by locating query keywords within them. If a document has a high number of query terms and it is regarded as being more "relevant" to the query than others documents with fewer or no query terms, it is regarded as being more "relevant" to the query than other presented to the user for further exploration, as the web search engines do. Besides text-based files, audio and other multimedia files with text-based definitions of attributes such as id, format, artist can be retrieved by this method. However, in most software applications, digital audio frequency is always managed as an opaque stream. There is no defined word or text-based object for comparison. Thus, we need a more effective and efficient approach to manage the audio retrieval.

Content Based Music Retrieval (CBMR) [1][2] refers to searching audio data of music through the database to retrieve a small number of music files that possess similarities to the user input. The user can input a music file instead of the traditional query options, keywords of text. In order to achieve content-based music retrieval, automated methods are developed to identify the composition and content of the music files and to search the music data based on the identified content of the query samples. Compared to a text-based music retrieval system, CBMRS [1][2] is rather general more than biased. Text-based music retrieval relies on the human operators to input the descriptive text for music files, which is based on the human perception of the music files. The CBMRS is more flexible in music retrieval and readily adaptable to query variations.

In the aspect of audio feature extraction, the MFCCs are coefficients that represent audio signal based on perception. They are extracted from music files by a feature extraction process. In this paper we used 12-dimentional mathematical coefficients of MFCC feature vectors for sound modeling. We have tested two quantization algorithms: clustering and binary tree techniques for the distance measure. These two quantization alogrithms have been used to group MFCCs with similar values together. Based on the quantization, the histogram of each music file is constructed. In distance computation, Euclidean and City Block distance are employed for calculating the similarity between histograms.

In this paper, a new CBMR system based on the quantization technique has been proposed and implemented. The system can search out similar music files using the query sample in a short time. The retrieval accuracy of tree-based quantization algorithm can achieve 90%.

Section 2 gives the introduction to the music retrieval system. Section 3 describes the proposed

Content Based Music Retrieval (CBMR) system. Sections 4 and 5 present the experimental results and conclusion, respectively.

# 2 Introduction to Music Retrieval

## *2.1 Multimedia Music Retrieval*

Music retrieval refers to searching music through the database to retrieve a small number of music files that possess similarities to the user input. The input from the user can be either a text or a music file. There are two types of music retrieval methods. One is text-based music retrieval and the other is content-based music retrieval. One of the important factor that affects the results from the music retrieval system is the quantization implementation. A good quantization implementation can increase the efficiency in searching music files and increase the accuracy of the retrieval results. In this paper we have proposed and implemented a tree-based quantization for the CBMR system. The result of the tree-based CBMR system shows in section 4.

### 2.1.1 Text-based Music Retrieval

Searching music files using text-based music retrieval is the most common technique. The Yahoo website is currently using this method for users to search music files. In this algorithm, music files, together with some text-based attributes like description or keywords, are stored into the database by human operations. Music retrievals are then performed by matching query texts with those descriptive keywords [3]. There are several problems inherent in such systems. First, a music retrieval query will fail if a user forms a query based on a set of keywords that are absolutely or partially different from the stored keywords of the music contents that the user is referring to. Second, because of the competition between human operators and enormous volumes of data, the process is significantly time consuming for getting exactly accurate results. If there are no acoustical properties of the underlying data employed, it is preferable to describe the acoustical properties with the text. But it is difficult or nearly impossible to describe them in text.

### 2.1.2 Content-based Music Retrieval

While information retrieval for text relies on simple text queries, the structure for a query for music is not so obvious. Though textual descriptions can be assigned to music, they are not always apparent or indeed well-defined. The content-based retrieval applications can avoid the problem somewhat by using music query samples. Some recent work improves on this by allowing the user to sing or whistle a desired tune. Some of the others work by uploading a music file as a query example.

Depending on music data types used in queries and the underlying database, music retrieval systems can be classified as follows [4]:

Symbolic query on a symbolic database: Both the query and the underlying database are in symbolic formats, such as MIDI and Humdrum. Retrieval problems on such symbolic data typically can be addressed by methods derived from text searching techniques. Several systems of this type have been implemented, including the Themefinder project (http://www.themefinder.org), where the symbolic database can be searched using pitch sequences, intervals, approximate contours, etc.

Monophonic acoustic query on a symbolic database: This problem, also known as Query by Humming or QBH, has received considerable attention during the past few years. In such systems, the input query is typically given as a user-hummed melody through a microphone, and the melody is analyzed and matched against a symbolic database. Human-hummed tunes are monophonic melodies and can be automatically transcribed into pitches with reasonable accuracy. In order to compensate for possible inaccuracies of human-hummed tunes, some systems use approximate contour information (up, down, etc.) or beat information to aid the retrieval process.

Polyphonic acoustic query on a polyphonic acoustic database: Input queries are acoustic, typically short sound clips, and the goal is to find similar acoustic pieces from the database. In this paper we will focus on this problem.

# 3 Introduction to the Proposed CBMR System

In the proposed Content Based Music Retrieval (CBMR) system, music files are characterized by histograms derived from a vector quantizer. Music similarity can be measured by comparing histograms.

The basic operation of the proposed retrieval system is as follows: first, a suitable corpus of music examples from a database must be accumulated and parameterized into feature vectors. The corpus must contain examples of the classes of music files to be discriminated between, e.g, different music types, different instruments. Next, a quantizer is constructed using the quantization methods based on the feature vectors. This quantizer transforms the feature vectors of training data into histograms. To

retrieve music by similarity, a histogram is constructed for the query file, as well. Comparing the query histogram with corpus histograms will yield a similarity measure for each music file in the corpus. These can be sorted by similarity and the results presented as a ranked list as in conventional text retrieval. Fig.1 represents the framework of the proposed music retrieval system.

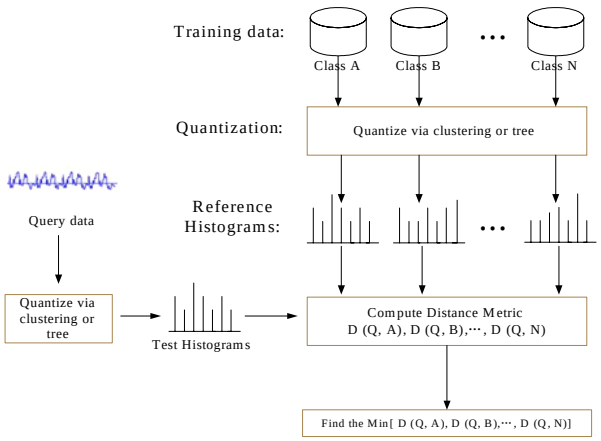


Fig.1 Framework of the proposed Music Retrieval

The proposed CBMR system comprises three components: audio feature extraction process, quantization process and searching process. Sections 3.1 to 3.3 explain more details about them.

## 3.1 Audio Feature Extraction Process

The audio feature extraction process parameterized the music files into MFCCs [5]. First, the music is Hamming-windowed in overlapping steps. Generally, each window is 25mS wide and these windows are overlapped so that there are 500 windows, hence feature vectors, in a second of music audio. For each window, the log of the power spectrum is computed using a discrete Fourier transform (DFT). The log spectral coefficients are perceptually weighted by a non-linear map of the frequency scale. This operation, called Mel-scaling, emphasizes mid-frequency bands in proportion to their perceptual importance. The final stage is to further transform the Mel-weighted spectrum (using another DFT) into "cepstral" coefficients. This results in features that are reasonably dimensionally uncorrelated. Thus, the final DFT is a good approximation of the Karhunen-Loeve transformation for the Mel spectra. Fig.2 represents the framework of MFCCs feature extraction.

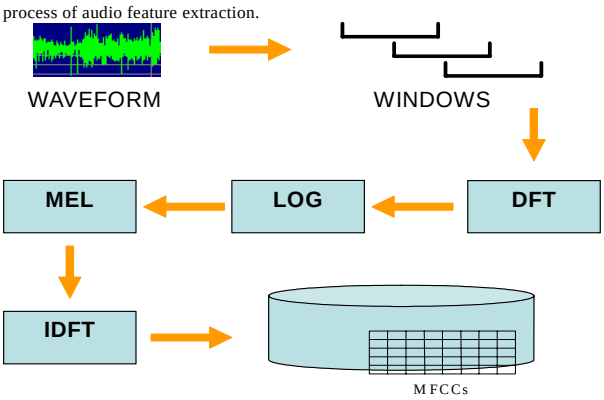


Fig.2 Framework of MFCCs feature extraction

In the CBMRS, the results of feature extraction are stored in the text files.

## 3.2 Quantization Process

Once data has been parameterized, a vector quantizer is grown off-line using as much training data as practical. Through the quantization, the training data are transformed into reference histograms. Fig.3 shows the vector quantization in the CBMRS. In the CBMRS, a user can choose to use binary tree or clustering to generate the vector quantizer.

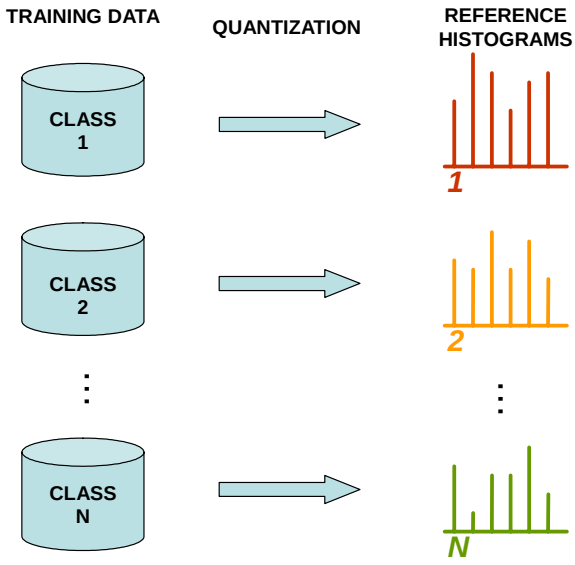


Fig.3 Vector Quantization in the proposed CBMR system

### 3.2.1 Tree-based Quantization in the Proposed CBMR System

In the proposed CBMR system, the tree-based

quantization works as follows: At first, the text-format MFCCs are converted into the binary fff format. Based on the fff-format MFCCs of training data, a tree is generated by the growtree application. The tree file is stored as tree.tfl. After that, the probtree application takes the tree file and the information of training data and generates a histogram file for each training data. These histograms are stored as tab files. These histogram files will be used as reference data for comparing the similarity between the query sample and each training data of the music file in the database.

### 3.2.2 Clustering-based Quantization in the Proposed CBMR System

In the clustering-based quantization, there are four types of information that have to be provided before the clustering process can start. They are (1) the chosen seeds, (2) the constructed MFCCs, (3) the dimension size of the MFCCs, and (4) the number of clusters to be created.

The results are stored in the text files. One text file contains the centroids of the clusters and n text files are created for n clusters formed where each text file contains the feature vectors information of the training data for each cluster.

In the proposed CBMR system, three algorithms are employed to cluster MFCCs. They are Forgy, K-means and K-Harmonic Means algorithms. They are partitional clustering algorithms, which means that samples are grouped into clusters based on similarity. After the quantization, histograms of the training data are generated. The histograms of each data are stored in the text files.

## 3.3 Searching Process

The searching process [6] is based on the distance calculation between the query sample's histogram and each reference histogram. Two distance metrics, the Euclidean distance [6] and the City Block distance [6], are employed in the proposed CMBR system. By searching out the minimum distance between query and reference samples, the results presented as a ranked list as in conventional text retrieval. Fig.4 illustrates the searching process of the proposed CBMR system.

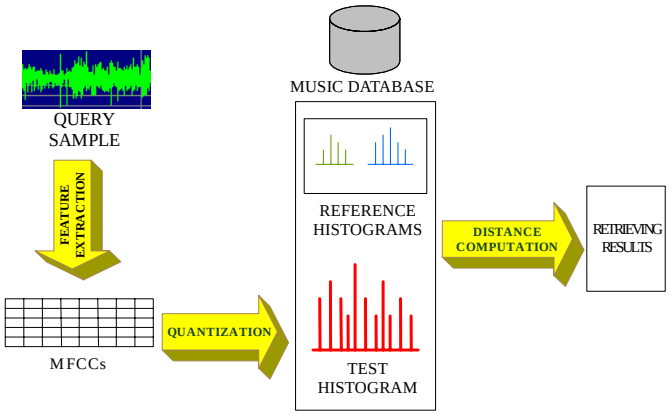


Fig.4 Searching Process of the proposed CBMR system

# 4 Testing Results

The retrieval performance assessment rule is as follows: each music file has been chopped into 5 clips of 4-second length and 1 clip of 3-second length. These clips are partially overlapping. Because these clips are from the exact one music file, they are considered similar with each other and relevant to that music file only (the clips of 1 music file are in the exactly same class only). In the feature extraction process, all of the music clips are transformed into 12-dimensional window feature vectors.

For the retrieval test in section 4.1, the Percentage of Retrieval Accuracy Method (PRAM) is employed to calculate the precision. PRAM is used to determine the percentage of accuracy results. The accuracy result of each query sample (the 3-sec clip of the music file), $a_i$ is measured by the number of training samples (the 4-sec clips of the music file) returned with quantization out of the total number of 4-sec clips chopped from the music file. PRAM is determined as shown below:

$$\frac{\sum_{i=1}^{K} a_i}{K} \times 100\%$$ , where $K$ is the total number of query samples.

The function of PRAM is computed by adding the accuracy results of all query samples performed and the percentage is obtained by using the mean accuracy result multiplied by100%.

### 4.1 Test 1: Retrieval Accuracy Based on Quantization

The purpose of this test is to determine which quantization method can obtain the best music retrieval accuracy rate. For the clustering-based quantization, we had tested a different number of clusters and found out which number of clusters can give the best music retrieval accuracy rate. We have compared different numbers of clusters: 16, 32, 64 and 128 for the total number of the parts of the music file in the same class. This test uses 20 query samples. Each query will return the top 5 results. This test is performed with 12-dimensional MFCCs on a total of 1,275 samples in 255 classes. Table 1 summarized the retrieval accuracy results based on the quantization methods and the number of clusters.

| **Quantization Methods** | | 16 clusters | 32 clusters | 64 clusters | 128 clusters |
|---|---|---|---|---|---|
| **Clustering-based Quantization** | Forgy | 78% | 82% | 82% | 86% |
| | K-means | 79% | 83% | 84% | 87% |
| | KHM | 79% | 82% | 83% | 86% |
| **Tree-based Quantization** | | 90% | | | |

Table 1 Retrieval Accuracy for the proposed CBMR system

From the results in Table 1, we found that the best number of clusters used is 128 clusters. As the number of clusters used increases, the accuracy of the results becomes higher.

Derived from the results in 128 clusters, we found that the K-means clustering algorithm can obtain the best result compared to other clustering algorithms. All of the algorithms employed have produced good results.

Compared with the clustering algorithms, tree-based algorithm can obtain the best retrieval result with a 90% accuracy rate. This shows that the retrieval precision depends on the dimension of histograms. In this experiment, for 128-clustering quantization, the histograms are 128-dimensional; on the other hand, for tree-based quantization, a 500-leaf tree is constructed to discriminate between classes. In this way, for each sample that is quantized by the tree, the dimension of the generated histogram is 500.

## 5 Conclusion

In this paper a new Content Based Music Retrieval (CBMR) system based on tree-quantization has been proposed and implemented. Compared with the text-based method for music retrieving, the proposed new CBMR system is more flexible and readily adaptable to query variations because it does not rely only on the human operators to input the descriptive text for music files, which is based on the human perception of the music files. Compared with other audio retrieval systems and MIDI music retrieval systems, the proposed CBMR system with quantization can handle polyphonic acoustic query on a polyphonic acoustic database. A high retrieval accuracy can be achieved by using a clustering-based quantizer with the retrieval accuracy 87%. It can obtain 90% high retrieval accuracy by using tree-based quantization.

*References:*

[1] Erling Wold, Thom Blum, Douglas Keislar, and James Wheaton, "Classifcation, Search and Retrieval of Audio" Muscle Fish LLC, 2550 Ninth Street, Suite 207 B, Berkeley, CA 94710, USA, 1997.

[2] J.T.Foote, "Content-based retrieval of music and audio," In C.-C. J.Kuo et al., editor, *Multimedia Storage and Archiving Systems II, Proc. Of SPIE*, Vol. 3229, pp.138-147, 1997. http://svr-w.emg.cam.ac.uk/jtf/papers/spie97-abs.html.

[3] J.T.Foote, "An Overview of Audio Information Retrieval," 18 December 1997

[4] Cheng Yang, "Peer-to-Peer Architecture for Content-Based Music Retrieval on Acoustic Data", Stanford University Department of Computer Science, Stanford, CA 94305, USA, May 2003

[5] Nina Ewerlaf, "Evaluation of support vector machines for content-based information retrieval of digital music data based on user preference," *Uppsala Universitetsbibliotek*, Uppsala, Sweden, 2003. http://www.uth.uu.se/tekniskfysik/exjobb/xjobb/ewerlof.doc

[6] J.T.Foote and H.F.Silverman, "A model distance measure for talker clustering and identification," In *Proc. ICASSP '94*, Vol.S1, pp.317-32, IEEE, (Adelaide, Australia), Apr. 1994.