

# A Novel Music Retrieval System with Relevance Feedback

Gang Chen<sup>1</sup> Tian-jiang Wang<sup>1</sup> Perfecto Herrera<sup>2</sup>

<sup>1</sup>Computer Science Department, Huazhong University of Science and Technology  
Wuhan, Hubei, China

<sup>2</sup>Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain  
wt\_jiangmail@yahoo.com.cn

## Abstract

*Although various researches have been conducted in the area of content-based music retrieval, however, few works have been done using relevance feedback for improving the retrieval performance. In this paper we introduce a novel content-based music retrieval system with relevance feedback. It enables users to search favorite music files by introducing the user as a part of the retrieval loop. In our system, we used a radial basis function (RBF) based learning algorithm and a method exploited both positive and negative examples to re-weight feature components. Experiments evaluate the performance of the proposed approach and prove the effectiveness of our system.*

## 1. Introduction

Rapid increase in speed and capacity of computers and internet has created the enormous number of music database. Traditional organization of music files using song's name, singer and other textual tags is not enough for modern users.

In recent years, to overcome drawbacks recognized in text-based approach, a content-based music analysis is getting more attention and various researches have been conducted in the area of Content-Based Music Retrieval (CBMR). However, due to the complexity and variability of the music and users' searching intention, it is not realistic to expect a music retrieval system to achieve satisfactory performance consistently for all users' intentions. Therefore, there may be situations where users are unsatisfied with retrieval results of the system, and feel the need to provide additional information regarding their preferences. The most efficient way to implement this is to collect relevance feedback (RF) from the user, and estimate the user's intention based on RF.

In fact, RF was first introduced for the retrieval of text documents in [1] and [2]. During retrieval process, the user interacts with the system and selects the relevance examples of the retrieved results, according to his/her subjective judgment. RF technique has been widely used to improve the

performance of Content Based Image Retrieval (CBIR) systems. There are two main RF strategies in CBIR systems: query shifting [3] and feature re-weighting [4]. In query shifting strategy the query is iteratively moving both towards regions containing subjectively relevant examples and against regions containing irrelevant examples. In a feature re-weighting method the goal is to find most relevant features for describing the similarity between examples and to reinforce their influence on similarity measure.

Although efforts on CBMR have been presented in recent years, however, there have been rather few works which have used RF methods for improving their retrieval performance. In related work, Haoshi et al. [5] used relevance feedback for music retrieval which is based on the Tree-structured vector Quantization (TreeQ) method. The approach of the TreeQ method is to train a vector quantize instead of modeling the sound directly. Michael et al. [6] presented an active learning music retrieval system using SVM and used relevance feedback information to shrink the version space of SVM. Seungmin Rho et al. [7] incorporated user's relevance feedback with genetic algorithm to improve retrieval performance and developed a prototype system.

In this paper a novel RBF based CBMR system with RF is proposed. We combines the non-linear modeling capability of the radial basis function (RBF) [8] and a new feature re-weighting algorithm, exploited both positive and negative examples with relevance feedback.

The rest of the paper is organized as follows: Section 2 describes our proposed system in detail. Section 3 discusses several experimental results. Finally, we conclude the paper and show the direction for future work in Section 4.

## 2. System detail

The system has a structure like an image retrieval system based on RF. One music file is input as a seed each time. At first, the system search music files based on the Euclidean distances between feature vectors of the query and music files from database. Music files from database then are sorted in

increasing order regarding to their distances from a query. Those music pieces are displayed on the screen, from which user recognizes and selects subjectively the best matched candidates. Then the feature vectors, extracted from selected music pieces, modify the centers, widths and weights of RBF along dimensions. The updated RBF model and weights are then used to evaluate subjective similarity in a new search. The process is repeated until the user is satisfied with retrieved results.

## 2.1. Feature extraction

In order to estimate the user's intention more accurately and comprehensively, we try to extract as many features as possible, providing a very rich description of every music file in the database. We used a set of 41 features of different kind including Temporal, Spectral, Tonal and Rhythm features [10] [11] [12] [13]:

- 1) Temporal features : Zero Crossing Rate, LPC;
- 2) Spectral features: MFCC, HPCP, Spectral centre, Spectral crest, Spectral flux, Spectral roll-off, Spectral energy, Spectral skew-ness, Spectral Strong peak, Flatness-DB, Spectral Bark bands et al.
- 3) Tonal features: Mode (major or minor), Key and Key strength et al.
- 4) Rhythm features: Onset rate, BPM, Beat loudness et al.

Most of these features are extracted using windowing. Afterward we compute statistics of these features (mean, variance, min, max, derivative mean, derivative variance) and normalize them.

## 2.2. Initial query

Given a data set  $X = \{x_1, x_2, \dots, x_m\}$  in  $\mathfrak{R}^N$ , here each  $x_i$  is the full feature vector of a music file. Each music file is represented by one vector  $x_i$  in  $\mathfrak{R}^N$ . At the beginning of a searching procedure user inputs a music file and the system drives its feature vector as the initial query vector as  $Q$ . Then, the similarity between music files from database and a query is objectively evaluated using Euclidian distance to return  $N$  music files most similar to the seed music file.

## 2.3. User's relevance feedback

From  $N$  retrieved music files, let the user select relevant music pieces, which are most similar to the user's query concept, while regarding rest of the returned music files as irrelevant. The query refinement mechanism is used to move the query point towards the middle of the cluster of relevant music files. We use the sets of relevant music files  $M_r$  and irrelevant music files  $M_{ir}$  specified by the user to calculate a new query point  $Q_{t+1}$  by applying Rocchio's formula [14]:

$$Q_{t+1} = \alpha \cdot Q_t + \beta \cdot \text{mean}(M_r) + \gamma \cdot \text{mean}(M_{ir}) \quad (1)$$

Where  $\alpha + \beta + \gamma = 1$  and  $\alpha, \beta, \gamma$  control the relative importance of the previous query, the relevant and the irrelevant music files respectively. In our system, we

set  $\alpha = 0.9, \beta = 0.8, \gamma = -0.7$ . This process is iterated across the RF iterations, progressively moving  $Q_t$  towards the optimal query point. Then, we use RBF network to evaluate the similarity in a new search. RBF networks possess an excellent nonlinear approximation capability. We utilize this property to design the system of locally tuned processing units to approximate the target nonlinear. In the current work, with radial-basis function in mind, we associate a 1-D Gaussian-shaped RBF with each component of the feature vectors as follows:

$$S(x, Q) = \sum_{i=1}^N W_i G(x_i, Q_i) \quad (2)$$

$$= \sum_{i=1}^N W_i \exp\left(-\frac{|x_i - Q_i|^2}{2\sigma_i^2}\right)$$

Where  $Q = [Q_1, Q_2, \dots, Q_i, \dots, Q_N]$  is the adjusted query position or the centre of the RBF function, and  $x = [x_1, x_2, \dots, x_i, \dots, x_N]$  is the feature vector associated with a music file in the database.  $\sigma_i$  is the standard deviation of the  $i$ th feature component in the form of RBF width, which is the inversely proportional to their density(Gaussian distribution):

$$\sigma_i = \exp\left(\left(\frac{1}{M-1} \sum_{k=1}^M (x_{ik} - \bar{x}_i)^2\right)^{\frac{1}{2}}\right) \quad (3)$$

Where  $M$  is the number of relevant music files. Each RBF unit implements a Gaussian transformation which constructs a local approximation to a non-linear input-output mapping. Based on our simulation study, the new single unit RBF networks is effective in learning and quickly converges for one-class-relevance classification using small volume of training sets.  $W_i$  is the weight of each feature component.

Unlike most feature re-weighting approaches which use only the standard deviation from relevant, we use a feature re-weighting method based on a set of statistical characteristics using relevant and irrelevant examples [9]. It uses the following formula to compute the discriminability of  $i$ th feature component as its weight, which is defined as:

$$W_i = 1 - \frac{m}{M}, m \in y_i \quad \text{and} \quad (4)$$

$$(Q_i - \sigma_i) \leq y_i \leq (Q_i + \sigma_i)$$

Where  $W_i$  denote the weight of the  $i$ th feature component.  $M$  represents the total number of relevant music files in database. The weight indicates the ratio of irrelevant music files located outside the range of RBF width, which represents the ability of this feature to separate relevant music from irrelevant ones. The method makes good use of irrelevant music files as well as relevant ones to calculate the weights of each feature component. Hence it only assigns large weights to the feature component which clusters all relevant music files together and scatters all irrelevant music files away from the relevant ones.

## 2.4. Feature selection

In order to identify a set of relevant features according to the user query while at the same time maintaining a small size feature vector to attain a good matching and lower complexity. To this end, the music description is modified during retrieval by removing the least significant features and better specifying the most significant ones [16].

We start from a subset in extracted feature space only containing the mean of each feature. After relevance feedback and calculating the weights associated to each feature, the following replacement rules are conducted:

- Removal: less discriminating feature (weight below a lower threshold  $T_L$  are discarded by setting the weights to zero);
  - Refinement: highly discriminating features (weight above a higher threshold  $T_H$  are replaced by a more detailed description (mean, variance, min, max, derivative mean, derivative variance));
  - Preservation: Other parameters are left unchanged
- In our system, we set  $T_L = 0.05$  and  $T_H = 0.55$  respectively.

## 3. Experiments

We present experiments to show the performance of our proposed system. We select two different music collections. These collections include a large number of pre-categorized music files, where the category can be used as ground truth. The first one is a genre database from Tzanetakis [15]. The genre database has the following classes: classical, country, disco, hiphop, jazz, rock, blues, reggae, pop and metal. The other one is a mood database tagged subjectively. The mood database includes: happy, aggressive, sad and relax. See Table 1 for a list of the number of music files in each category.

Table 1. Genre Database and Mood database

Genre	Number	Mood	Number
Blues	100	Happy	112
Classical	100	Aggressive	133
Country	100	Sad	134
Disco	100	Relax	228
Hiphop	100		
Jazz	100		
Metal	100		
Pop	100		
Reggae	100		
Rock	100		
Total	1000		607

The procedure of the experiment is described as follows:

- Treat the  $i$ th music category as the positive and the remaining as negative;
- Select an example from the positive category as a query to launch retrieval, and perform initial query;
- Sort the music files in the database according to the distance;

- According to the ground truth, label the top  $N$  music files to simulate user feedback. After that, these labelled music files are added into the current training data set;
- Based on the available training data, use the proposed method to re-weight the features;
- Redo steps 3 to 5 four times to simulate user feedbacks;
- To accumulate statistics, redo steps for each music files in the category. The obtained retrieval performance is averaged respectively;
- Redo steps 1 to 7 for each music category in the database and the obtained retrieval performance is averaged.

The system performance is estimated in terms of average retrieval precision (AR) as:

$$AP = \frac{R}{N} \quad (5)$$

$R$  is the number of relevant music files declared by the user and  $N$  is the number of music files from which user's feedback is performed. We set  $N = 20$  in the experiments. The results after 4 iterations can be seen in Table 2 and Table 3.

Table 2. Average precision of genre database

Genre	Average Precision			
	Iteration1	Iteration2	Iteration3	Iteration4
Blues	0.4025	0.6105	0.7125	0.7655
Classical	0.5263	0.8602	0.9409	0.9677
Country	0.4670	0.7170	0.8090	0.8475
Disco	0.4005	0.6305	0.7395	0.7875
Hiphop	0.6250	0.8790	0.9540	0.9720
Jazz	0.5330	0.8535	0.9460	0.9825
Metal	0.6715	0.9310	0.880	0.9955
Pop	0.4410	0.7500	0.8750	0.9345
Reggae	0.3395	0.6005	0.7480	0.8045
Rock	0.3010	0.4690	0.5510	0.6050
Average	0.4707	0.7301	0.8156	0.8662

As shown in Table 2, after 4 iterations, the average top 20 precision of 86.62% for ten musical genres is achieved. Rock music gives the worst retrieval precision at 60.50%. Rock is easily confused with other genres because of its broad nature. Metal music has the best precision at about 100% because of its special nature. The same result is reported in [15]. It can be seen from Table 3, the sad music has the lowest average precision at 74.59% because of a possible semantic overlap with other moods, and aggressive music is the most predictable.

Table 3. average precision of mood database

Mood	Average Precision			
	Iteration1	Iteration2	Iteration3	Iteration4
Happy	0.4161	0.6942	0.8304	0.9067
Aggressive	0.7586	0.9489	0.9812	0.9932
Sad	0.2869	0.4866	0.6407	0.7459
Relax	0.3862	0.6072	0.7500	0.8463
Average	0.4620	0.6842	0.8001	0.8730

We also compared our approach of our system with approach in MARS-1 [4] on the two databases. As always, the retrieval is performed over a substantial number of music files and the results are averaged. For each query, 10 feedback iterations are run. The result is shown in Figure 1. As we can see our approach outperforms in all 10 iterations.

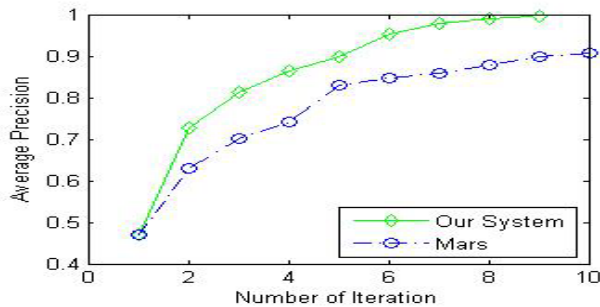


Figure 1. Comparison with Mars System

#### 4. Conclusions

When performing content-based music retrieval, it is very important to take into account the user's needs and specificities, which can be identified via relevance feedback. In this paper, we have proposed a novel content-based music retrieval system with relevance feedback. We validated our system by testing it on two different databases and conduct some performance evaluations. The results are quite comparable with those already reported in literature. Further research will be addressed to improvements in retrieval procedure, for instance, with more feature vector components and the use of different criterions for similarity computation.

#### 5. Acknowledgment

This work has been partially supported by a grant from National High Technology Research and Development Program of China (863 Program) (No. 2007AA01Z161). The work reported here was done in the Music Technology Group (MTG), Universitat Pompeu Fabra. We want to thank Emila Gomez, Joan Serra, Cyril Laurier and Ferdinand Fuhrmann for their helpful suggestions. We also want to thank the MTG human and technical infrastructure for providing support for this research.

#### References

[1] C. Rijsbergen, *Information retrieval*. Butterworths, London, 1979.

[2] G. Salton, *Automatic Information Organization and Retrieval*. McGraw-Hill, 1968.

[3] S. M, Y. Rui and T. Huang, "Content-based image retrieval with relevance feedback", in Proc. *IEEE Int Conf on Image Processing*, 1997, pp. 815–818.

[4] T. H. and Y. Rui. "Relevance feedback: A power tool for interactive content-based image retrieval", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 644–655, 1998.

[5] N. I. Keiichiro Hoashi and Kazunori Matsumoto, "Personalization of user profiles for content-based music retrieval based on relevance feedback", in Proc. *11 ACM international conference on Multimedia*, 2003, pp. 110–119.

[6] M.I. Mandel, G.E. Poliner, and D.P.W.Ellis, "Support vector machine active learning for music retrieval", *Multimedia Systems*, pp. 3–13, 2006.

[7] M. K, Seungmin Rho and Eenjun Hwang, "Music information retrieval using a ga-based relevance feedback", in Proc. *International conference on multimedia and ubiquitous engineering*, 2007, pp. 739–744.

[8] L. G. P and Muneesa wang, "An internative approach for CBIR using a network of radial basis functions", *IEEE Trans Multimedia*, vol 6, pp.703–716, 2004.

[9] A. Z. and Y. Wu, "A feature-reweighting approach for relevance feedback in image retrieval", in Proc. *IEEE Int. Conf. Image Processing ICIP2002*, 2002, pp. 581–584.

[10] S. Rho, E. Hwang, and M. Kim, "Music information retrieval using a GA-based relevance feedback", in Proc. *International conference on multimedia and ubiquitous engineering*, 2007, pp 739–744.

[11] S. Skovborg and E. Nielsen, "Evaluation of different loudness models with music and speech material", in Proc. *AES 117th Convention*, 2004.

[12] Streich S., "Music complexity: a multi-faceted description of audio content", Ph.D. Thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2007.

[13] Gomez Emilia. Tonal description of music audio signals. Ph.D. Thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006.

[14] J.J. Rochhio, "Relevance feedback in information retrieval". *The SMART Retrieval System: Experiments in Automatic Document Processing*, pp. 313–323, 1971.

[15] C. Tzanetakis and Essl, "Automatic musical genre classification of audio signals", in Proc. *ISMIR 2001*, 2001, pp. 205–210.

[16] Anelia G., Francesco G.B. N. Charlie D. and Thomas H., "Content-based image retrieval by feature adaptation and relevance feedback", *IEEE Transaction on Multimedia*, Vol. 9, pp. 1183-1192, Oct. 2007.