interspeech 2008 incorporating SST 08

Improving Searching Speed and Accuracy of Query by Humming System Based on Three Methods: Feature Fusion, Candidates Set Reduction and Multiple Similarity Measurement Rescoring

Lei Wang [1], *Shen Huang* [1], *Sheng Hu* [1], *Jiaen Liang* [1], *Bo Xu* [1,2]

[1]Digital Content Technology Research Center, Institute of Automation,
[2]National Lab of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing 100080, China
{leiwang, shenhuang, shu, jeliang, xubo}@hitic.ia.ac.cn

Abstract

In this paper, we present three methods for improving the searching speed and accuracy for query by humming (QBH) system with large melody database. 1) At the feature level, to minimize the inevitable errors caused by a single pitch extractor, three different pitch extraction algorithms are fused together to gain more credible and robust pitch sequence. 2) To speed up the matching process, a candidate set reduction method is firstly adopted to filter out the unlikely candidates by faster but less precise methods; then a more accurate but slower strategy is executed on the survival candidate set to perform a finer match. 3) At the decision level, we utilize these scores generated during the filtering stage and fine-matching stage to fusing together to get more accurate result. The proposed system achieved mean reciprocal rank of 0.929 for the corpus used in MIREX2006 [1] while cost an average of 0.58 seconds for one query. The results reveal the advantage of our system on speed and accuracy comparing to other system participated in that contest.

Index Terms: Query by Humming, Feature Fusion, Candidates Set Reduction, Multiple Similarity Scores Rescoring

1. Introduction

The query by humming (QBH) system not only provides a natural and convenient way for users to search in the large scale song database, but also offer an extra interface when users forget the song's name but only remember a part of the melody. This technology could be used in the following areas: In Karaoke application, QBH enable user to hum a small part of a melody to search the wanted song in large song database [3]; in the filed of mobile search, it provides more convenient interface with the acoustic querying than traditional keyword searching to download ringtone [4]; what is more, an online music community equipped with QBH has shown its superiority to attract users [5]. Due to its potential commercial values, this field has been focused by a large group of researchers, and there is a contest hold by MIREX every year since 2006 [1, 2].

Since the users' acoustic input is not exactly the same as templates stored in database but varies according to different users, voice qualities, user's singing states, and other factors, similarity matching scheme is needed in QBH systems. Among existing matching algorithms, note-based and frame-based matching are two commonly used methods. The former one measures the similarity between the transcribed notes sequence and the music score [6, 7]. It is much faster than frame-based matching. Nevertheless, note transcription is error-prone and has a very negative influence on the final result. To avoid the error caused by note transcription, frame-based matching [8, 9] is proposed to compute on pitch sequence directly and has showed its advantage in MIREX 2006. However, three problems of frame-based matching are still unsolved until now. 1) Even though frame-based feature produces less error than note-based feature, pitch tracking algorithms are still far from perfect. 2) Due to it needs more computation at frame-level, this method is time-consuming with large song database. 3) Traditional similarity match in QBH systems always focus on a single similarity measurement alone to make decision, however, there are some useful additional information available in producing the final result.

In this paper, we present our approach to solve the above problems in detail. First of all, a pitch-level fusion is used to produce a more robust and creditable feature sequence. Secondly, to speed up the searching process, we perform a candidate set reduction method. The aim is to reduce possible candidate set by faster scheme for further refined search, while still reserving the true answer in the candidate set. Finally, the final decision is made by the rescored result of these multi-similarity scores generated in the filtering and fine-matching stage. In addition, this rescoring mechanism needs no additional computation.

The outline of this paper is organized as follows. Section 2 offers an overview of our QBH system. In section 3, the proposed methods are introduced in detail. Experiment results are given in section 4. Conclusions are drawn in section 5.

2. Overview of Our QBH System

As shown in figure 1, our system is functionally divided into 3 parts: 1) music score processing module, taking music score file to generate song template database; 2) Acoustic processing module, taking user's humming/singing as acoustic input queries to extract features; 3) template matching module, computing the similarity between template and user's query using certain measurement and produce the final ranked candidates sorted by similarities.

2.1. Music Score Processing Module

Since MIDI contains the exact information of music scores and it is prevailing on internet, we take it as our music scores source to build up melody database. MIDI files usually contain main melody track and other accompany tracks. Only main melody track is useful to build melody database. To save human's laborious work, we first use a pattern recognition approach to extract melody automatically. Then, to make the main melody extraction result more credible,

manually check by human-being is performed on these extracted tracks with low confidence.

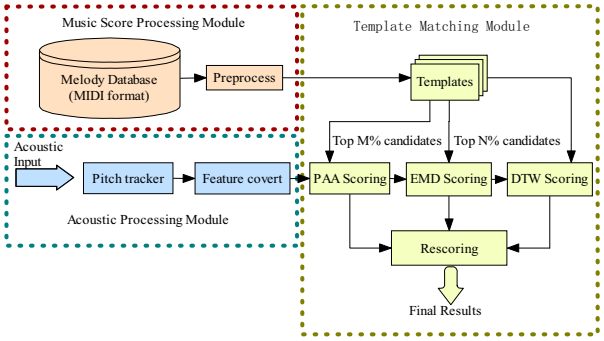

Figure 1: *Overview of Our QBH System*

To support humming/singing from anywhere to anywhere of a specified song, phrase segmentation method is adopted to locate entry point of a melody. According to the statistical analysis of our song queries set, 98.5% people hum from the beginning of a music phrase, while 94.1% from the start or the tidemark of the song. Therefore, it is possible to select the starting point of a music phrase as entry point. In our system, we use our hybrid rules including Max Repeated String (MRS) algorithm, pause, long note, theme pattern length to locate the potential entry point of a melody.

2.2. Acoustic Processing Module

We do not make any constraint on input methods, users could sing lyrics or humming a tune of a specific melody or even could query with meaningless syllables. When the acoustic input is obtained, the pitch trackers [10, 11, 12] extract the pitch sequence individually.

Since no pitch extraction method is perfect, errors might be introduced during this stage. There are mainly two kinds of pitch errors: 1) double or half of the normal frequency; 2) small pitch fluctuation caused by jitter. To mitigate the impact caused by the two error source, a five-point median filter is adopted to generate smoother pitch sequence. Furthermore, to take advantage of these complementary extracted pitch results, pitch fusion is used to generate the more robust pitch contour. Detail fusion methods are discussed in section 3.

2.3. Template Matching Module

The template matching module is in a multi-stage scheme. The first stage is based on Piecewise Aggregate Approximation [13, 14], a dimensional reduction method for fast filtering unlikely candidates. Earth Mover's Distance, another fast filtering scheme, further reduces the candidates set greatly. Then, Dynamic Time Warping, a frame-based and more accurate algorithm, is used for finer matching on the surviving candidate set. Finally, a rescoring method is applied to integrate these scores gained from the three-stage matching on the surviving candidates set to generate final re-ranked list. The matching methods and their fusion are also introduced in detail in section 3.

3. The Proposed Methods

3.1. Pitch Fusion for More Robust Pitch Sequence

Fundamental frequency (F0) or pitch is an important feature in many speech relevant applications. Numerous studies [10, 11, 12] have tried different ways to reduce pitch estimation error rate, but there always be a limitation of one specific pitch estimation algorithm. To obtain a more robust pitch sequence, we propose a multiple pitch extractors fusion method to achieve better precision by taking advantage of complementary pitch estimation methods together.

The reason why this mechanism can improve pitch's robustness is simple: None of the pitch extractors is perfect, and each of the individual pitch extractors may produce some errors. But, if one extractor produces error at some frames, the possibility of all of the other pitch extractors produce wrong pitch value at the same frame is small. Therefore, the fusion result of well-selected pitch extractors that are complementary to each other would reduce overall estimation error rate.

To get the pitch fusion result, three different pitch estimation algorithms are used to extract pitch sequence independently. Then, two methods of fusion are used: median-value fusion (MVF), and dynamic programming fusion (DPF). The former one is to take median value of pitch candidates generated by these three pitch trackers of one frame as the fused result. The latter one is to find a globally optimum path from these pitch candidates based on dynamic programming. These two fusion methods are effective to reduce the pitch estimation error rate.

3.2. Candidate Set Reduction for Speeding up the Matching Process

One big challenge of frame-based matching is to seek a way to speed up the searching process. Since it is hard to build index on high-dimension time series database, the filter-and-refine principles is often used [3, 8, 9]. In the filtering step, a large subset of candidates that obviously has nothing to do with the answer is filtered out. In the refining step, the smaller candidate set is checked in more detail based on finer matching algorithm. In our system, we use PAA and EMD as the filter and DTW as the detailed match method.

3.2.1. Piecewise Aggregate Approximation

According to stable property of speech signal in a short-time, pitch value would not vary greatly in a small period of time. Such local correlation of pitch value has the potential to reduce feature dimensionality by means of using the sequence of the average value of a short segmentation of time to representation the shape of pitch contour. The so-called Piecewise Aggregate Approximation (PAA) algorithm is proposed independently by Yi and Keogh [13, 14].

As shown in Fig.2, the original pitch sequence is generated by pitch fusion method and each pitch point corresponds to 50ms. So, an 8-second length query has got 160 point in total. The pitch sequence is divided into 8 equal-sized segments and the average value is computed in each segment. With PAA transformation, the pitch sequence could be approximately represented with much shorter PAA vector. To compensate the error caused by the variation of humming speed, we perform a linear scaling during PAA matching.

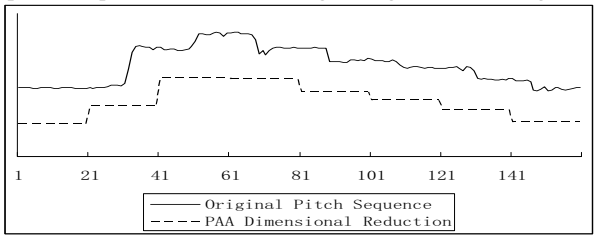

Figure 2: Dimensional Reduction by PAA

Empirically, the PAA method seems promising: since it needs a very low computational cost, even a linear search over database items could be tolerated. In our experiment, only 0.05 second is needed to perform PAA matching with a database of 2048 songs.

3.2.2. Earth Mover's Distance

Suppose one point set as "supplier" which is a number of hillocks, with which weight represented as the volume occupied. Another set as "demander" can be imaged as a mass of holes with a certain amount of capacities. EMD measures the minimum flow work needed to transport earth of these hillocks to fill the total amount of holes, with which the cost considering both capacities and two dimensional positions. Such kinds of problem can be transformed into the linear programming [7].

EMD has robust properties to errors brought about in the front end. Note segmentation mistakes to fragment note into two separate parts. In string matching, this action will induce an insertion cost and otherwise a deletion cost from query to template. But in EMD, Such error is trivial because EMD is in a top-down scale of view, the contribution of query notes is approximate equal to the notes of template, no matter how much fragmented or consolidated errors segmentation brings. As for insertion or deletion error, the influence of string matching is two units cost, but EMD is little, either. Moreover, since EMD needs much less computational cost than DTW, to use it as a filtering scheme could also reduce overall searching time significantly.

3.2.3. Dynamic Time Warping

Traditional similarity measure method, such as Euclidean distance, is very sensitive to missing a data point, or having additional noise points. This is particularly the case for the query by humming system. The user's input may be performed faster or slower comparing to the template stored in melody database, thus, the Euclidean distance would produce significant error. To overcome this problem, dynamic time warping (DTW) algorithm is adopted to fill the gap caused by tempo variation between each individual user's acoustic input and music score template. The DTW algorithm is widely used for comparing time series with non-linear distortions in the time axis. This method exhibits good performance in isolate word speech recognition and query by humming [8]. As the finer matching method in our system, DTW algorithm searches the path with the least global distance between query and reference. However, it needs much more computational time than PAA and EMD.

3.3. Multi-Similarity Measurement Fusions to Achieve More Credible Result

During the experiment, although we noticed that although DTW outperforms other methods, the wrong decision made by the other matching methods would not necessarily be the same. This property suggests that these scores generated through filtering stage could offer complementary information which could improve the performance of the final decision. Actually, the combination of different similarity measurement is a way of uncertainty reduction. Many fusion methods are proposed to integrate these multiple information together to get better results, such as dynamic information selection, multi-information structuring and grouping, hierarchical mixture of experts, candidate set reduction, parallel voting strategy and so on. Among these methods, voting fusion is the straightest way to take advantage of the multi-source information in this task.

If only ranked candidate set is available, the voting fusion is often used as a decision level fusion. Nevertheless, in this method all the similarity measurements are considered equally important so that a less reliable one may disturb the final decision significantly. A possible way of overcoming this drawback is that of including a measure of the importance of each similarity scores in the fusion rules. The matching method with higher accuracy is more likely to be correct and should be given more voting weight. Thus, the weighted voting fusion can be expected a high performing solutions by assigning appropriate degrees of weight on the scores to vote. The weighted voting function is

$$score(t_j) = \sum_{i=1}^{M} weight(c_i)\, prob(c_i, t_j)$$

Where $weight(c_i)$ is the weight of similarity measurement c_i and $prob(c_i, t_j)$ is the similarity obtained when c_i runs on candidate t_j, and the candidate with the highest voting scoring is assigned as the winner:

$$t_j^* = \arg\max_{j} score(t_j)$$

Furthermore, this fusion mechanism does not need additional computation cost but only utilize the byproduct generated by the PAA, EMD and DTW process. As shown in figure 1, since the PAA and EMD runs much faster than DTW, we perform PAA at first, and only reserve top M% of candidate with highest PAA scores. Then EMD is performed with the same scheme for further reducing the survival candidate set. During the filtering stage, the score of PAA and EMD is saved. Therefore, when performing filtering stage, we achieve two objectives with a single effort: not only the candidate set is reduced, but the scores of the two similarity measurement are also saved for the rescoring stage.

4. Experiments

4.1. Database

To make our result comparable, the corpus used in MIREX 2006 Query by Singing/Humming Contest provided by Roger Jang is taken to test the proposed methods. This corpus [16] contains 48 monophonic MIDI files plus 2000 non-target MIDI files as melody database and 2797 sung queries from 118 persons as the acoustic input. The non-target files are selected from ESSAN collection [15]. Since the contest does not provide the exact file list of these non-target files, to make our result comparable, we select 2000 files randomly from the ESSAN collection. These sung queries were humming or singing from the beginning of the song and last for exactly 8 seconds. The way for singers to query is various: some of these song queries are sung with lyrics, some are hummed nasal voice, even some are queried with nonsense syllables. All the querying files are digitized at 8 kHz, 8 Bit, PCM format. The performance is evaluated in three measurements: Mean Reciprocal Rank (MRR), Top-1 rate, and Top-20 rate. MRR is calculated by the reciprocal of the rank of the first correctly identified cover for each query (1/rank) [1]. Top-1 rate is the percentage that the correct melody is ranked in the first place of all returns. Top-20 rate is the rate that the correct melody is ranked within the top 20 of all returns.

Table 1. *Comparison of pitch extracting methods*

Method	TOP 1	TOP 20	MRR
Pitch1	89.91%	95.17%	91.40%
Pitch2	89.56%	95.53%	91.26%
Pitch3	90.52%	95.96%	92.10%
MF	91.34%	95.99%	92.63%
DPF	91.49%	96.38%	92.91%

Table 2. *Speed comparison of matching methods*

Method	Computation Time
PAA (alone)	0.05 seconds
EMD (alone)	0.72 seconds
DTW (alone)	9.2 seconds
PAA + EMD + DTW	0.58 seconds

Table 3. *Accuracy comparison of matching methods*

Method	TOP 1	TOP 20	MRR
PAA (alone)	33.14%	71.93%	43.36%
EMD (alone)	71.89%	80.10%	74.98%
DTW (alone)	86.35%	94.74%	88.88%
Rescoring	91.49%	96.38%	92.91%

4.2. Evaluation on the Proposed Methods

Without loss of generality, for comparison, we tested the three individual pitch extractors [10, 11, 12], the Median Fusion (MF) and the Dynamic Programming Fusion (DPF) with the same matching strategy: rescoring with the result of PAA, EMD and DTW's scores. From the results shown in table 1, both the fusion methods are performs better than using one pitch extractor alone. DPF integrates the pitch sequences' global information of, the result is even better.

When evaluating the performance of candidate set reduction, we test our system on the computer with Intel Pentium4 3.0GHZ, 1G memory, which is very similar to the computer used in MIREX2006 [1]. The recall rates of PAA and EMD according to different surviving rate are shown in figure 3. The average computation time is given in table 2. Since PAA and EMD run much faster than DTW, using PAA and EMD as filters could improve the searching speed significantly. It only cost 0.58 seconds to get a final result when PAA and EMD surviving rate are set to be 30% and 10% respectively. The result is very promising. It speeds up for about 15 times than using DTW alone to perform a linear search over the whole database.

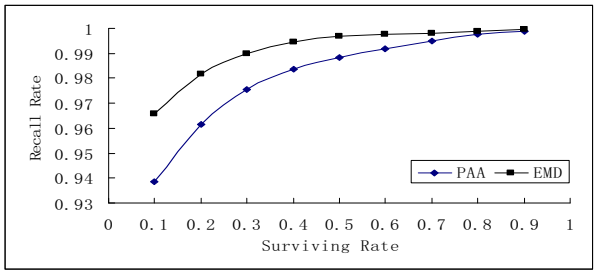

Figure 3: *recall rate over different survival rate*

When testing the weighted rescoring method, the PAA and EMD surviving rate are set to be 30% and 10% respectively while the weights of these three scores are also set to be 10%, 30% and 60%. Table 3 shows that rescoring methods performs much better than using single score alone.

Table 4. *Overall system performance comparison*

Method	MIREX2006		MIREX2007	
	MRR	Speed	MRR	Speed
System I	92.6%	0.89 s	92.5%	NONE
System II	88.3%	9.16 s	87.2%	NONE
Our System	92.9%	0.58s	92.9%	0.58s

As shown in table 4, we compared with the two best systems [9, 8] in Query-to-Database, QBSH task in both MIREX 2006 and MIREX 2007. The dataset used in these two contests is the same. The experimental results demonstrate that the proposed system outperforms these systems in both accuracy and speed. In addition, since there is a trade-off between speed and accuracy, the speed performance could be even better if sacrifice some accuracy.

5. ACKNOWLEDGEMENT

This work was supported by the National High-Tech Research and Development Plan of China under Grant 2006AA010103.

6. Conclusion

In this paper, we propose three methods for improving the searching speed and accuracy for query by humming system: pitch fusion at the feature level, candidate set reduction to quicken up the matching process, and multiple scores fusion at the decision level. The proposed methods are evaluated on the dataset used in the *Mirex2006 Query by Singing/Humming Contest*. Experimental results show that our methods could achieve 91.49% top-1 accuracy and 92.91% MRR on this dataset while only cost an average of 0.58 seconds for a query.

7. References

[1] http://www.music-ir.org/mirex/2006/index.php/
[2] http://www.music-ir.org/mirex/2007/index.php/
[3] Roger Jang, "A General Framework of Progressive Filtering and Its Application to Query by Singing/Humming", IEEE Trans. Speech, Audio and language Proc., 2(16):350-358, 2008.
[4] Lie Lu, "Mobile RingTone Search through Query by Humming," in Proc. ICASSP2008, pp.2157-2160.
[5] http://www.midomi.com/
[6] E. Lopez y M. Rocamora. "Tararira: Sistema de búsqueda de música por melodía cantada". X Brazilian Symposium on Computer Music. 2005.
[7] R.Typke. "Using transportation distances for measuring melodic similarity". ISMIR. 2003.
[8] J.-S. Roger Jang, Nien-Jung Lee, and Chao-Ling Hsu, "Simple But Effective Methods for QBSH at MIREX 2006", in Proc. Int. Symp. on Music Inf. Retrieval 2006, Victoria, Canada, Oct 2006.
[9] X. Wu and M. Li, "A top-down approach to melody match in pitch contour for query by humming," in Proc. 5th Int. Symp. Chinese Spoken Lang. Process., Singapore, 2006
[10] L. Hui, B.-q. Dai, and L. Wei, "A pitch detection algorithm based on AMDF and ACF," in Proc. ICASSP2006 pp.377-380.
[11] P. Boersma, "Accurate short term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound". Proc. Institute of Phonetic Sciences 17. Univ. Amsterdam. 1993.
[12] A. de Cheveigne, "YIN, a fundamental frequency estimator for speech and music," J. Acoust. Soc. Am., 2001.
[13] B.-K. Yi and C. Faloutsos. "Fast time sequence indexing for arbitrary lp norms". In VLDB, pages 385-394, 2000.
[14] E. Keogh, "Dimensional Reduction for Fast Similarity Search in Large Time Series Databases", Journal of Knowledge and Information Systems, pp. 263-286, Match, 2001.
[15] http://www.esac-data.org/
[16] http://neural.cs.nthu.edu.tw/jang2/dataSet/childSong4public/QBSH-corpus