COMPARISON OF FEATURES FOR DP-MATCHING BASED QUERY-BY-HUMMING SYSTEM

Akinori Ito

Graduate School of Engineering
Tohoku University
Aoba 05, Aramaki,
Sendai, 980-8579 Japan
+81 22 217 7084
aito@makino.ecei.tohoku.jp

Sung-Phil Heo

Korea Telecom Research & Development Group
17 Umyeon-dong,
Seocho-gu, Seoul,
139-792 Korea
+82 31 702 9749
hsphil@hotmail.com

Motoyuki Suzuki

Graduate School of Engineering
Tohoku University
Aoba 05, Aramaki,
Sendai, 980-8579 Japan
+81 22 217 7112
moto@ecei.tohoku.ac.jp

Shozo Makino

Graduate School of Engineering
Tohoku University
Aoba 05, Aramaki,
Sendai, 980-8579 Japan
+81 22 217 7172
makino@ecei.tohoku.ac.jp

ABSTRACT

In this paper, we compared three kinds of similarity measures for DP-matching based query-by-humming music retrieval experiments. First, a DP matching-based algorithm is formulated using the similarity between a deltaPitch of an input humming and that of a song in the database. Then the three similarities are introduced: distance-based similarity, quantization-based similarity and fuzzy quantization-based similarity. The three similarities are compared by experiments. From the experimental results, the distance-based one gave the best recall rate. In addition, we examined the combination of distance-based and fuzzy-quantization-based similarities. The experimental result showed that the recall rate was improved by the combination.

1. INTRODUCTION

Rapid progress of hardware and software technologies makes it possible to manage and access large volumes of music data. To access the music contents more easily, content-based music information retrieval systems have been developed. Some of these systems use a user's humming as a key to information retrieval. The input humming is segmented into notes, and pitch frequencies are extracted. DeltaPitch and inter-onset interval (IOI) ratio are often used as features of the humming. Then the input is matched with the music in the database. The matching method involves two aspects: the similarity measure and the matching algorithm. Ghias et al. used the deltaPitch quantized in three levels (up, down and same)[1] as a representation of a note, and the identity of the quantized codes was used as the similarity of the two notes. McNab et al.[2] employed a similar technique. Sonoda et al. used a similarity based on hierarchical quantization of deltaPitch and IOIratio[3]. For the matching algorithm, dynamic programming (DP) based matching algorithm is commonly used[1, 3]. The MIRACLE system[4] employed a two-level matching algorithm that utilized linear matching and DP matching.

In this paper, we compare several similarity measures between notes from the retrieval accuracy point of view. The quantization-based similarity is the most popular approach, but there seems to be a couple of other possibilities of similarity measures. The continuous DP matching is chosen as a matching algorithm. Then three kinds of similarity measures are compared.

The MIR system used in this work assumes that a music database contains information of musical pieces with monophonic melody, and heights and lengths of the notes in the database are taken from MIDI data. When an input humming is given, the feature sequence is extracted from the input[5]. First, the input signal is segmented using a band-pass filter and power threshold. Then pitch frequencies are extracted from the humming. The sequence of the deltaPitch is calculated from the extracted pitch frequency. Here, a deltaPitch value is expressed as *cent*, i.e. if we have contiguous notes of f_1(Hz) and f_2(Hz), the deltaPitch value Δf is

$$\Delta f = 1200 \log_2 \frac{f_2}{f_1}. \qquad (1)$$

2. DP-MATCHING BASED MUSIC INFORMATION RETRIEVAL

DP matching is a popular approach to measure the distance between an input humming and a song in the database. The DP matching is a matching algorithm that considers insertions and deletions of notes in a humming input. As there are many variations of the DP-based matching, we employ a continuous-DP[6] based algorithm.

Let a deltaPitch sequence of the input humming be $h(1),\ldots,h(J)$, and that of m-th song in the database be $d_m(1),\ldots,d_m(I)$. Let the similarity between the i-th note of the m-th song in the database and the j-th note in the input humming be $S_m(i,j)$. This similarity is defined in several ways later. Here, $S_m(i,j)$ for $i \leq 0$ or $j \leq 0$ is defined as $-\infty$.

Now DP-score $g_m(i,j)$ are calculated as follows.
for $j=1$

$$g_m(i,1) = S_m(i,1) \tag{2}$$

for $j=2$

$$g_m(i,2) = \max \left\{ \begin{array}{l} g_m(i-1,1) + S_m(i,2) \\ g_m(i-2,1) + S_m(i,2) + \beta \end{array} \right. \tag{3}$$

for $j \geq 3$

$$g_m(i,j) = \max \left\{ \begin{array}{l} g_m(i-1,j-2) + (S_m(i,j)+\beta)/2 \\ g_m(i-1,j-1) + S_m(i,j) \\ g_m(i-2,j-1) + S_m(i,j) + \beta \end{array} \right. \tag{4}$$

Here, β is a penalty value for insertion and deletion errors. This algorithm assumes that the insertion errors or the deletion errors do not occur successively. Now $g_m(i,J)$ is an optimum score between the input humming $h(1),\ldots,h(J)$ and the m-th song assuming that the note $h(J)$ corresponds to the note $d_m(i)$.

Using $g_m(i,j)$, the score of song m is calculated as

$$V_m = \max_i g_m(i,J). \tag{5}$$

Finally, top-N songs that have the highest V_m are chosen as the retrieval result.

To evaluate a retrieval result, we employ the top-10 recall rate that is the ratio of the queries for which the correct song is listed within the top-10 candidates. Let N_Q be number of the queries and r_i be the rank of the correct song in the retrieval result of the i-th query. Let the rank-hit function $h_k(n)$ be

$$h_k(n) = \left\{ \begin{array}{ll} 1 & \text{if } n \leq k \\ 0 & \text{otherwise} \end{array} \right. \tag{6}$$

Then the top-10 recall rate R_{10} is calculated as

$$R_{10} = \frac{1}{N_Q} \sum_{i=1}^{N_Q} h_{10}(r_i). \tag{7}$$

3. DISTANCE-BASED MATCHING

3.1. Distance-based similarity

The most straightforward way to calculate the similarity $S_m(i,j)$ is to observe the difference between $d_m(i)$ and $h(j)$. If they are similar, then the difference between them is nearly zero. Then we can define $S_m(i,j)$ as

$$S_m(i,j) = -|d_m(i) - h(j)|. \tag{8}$$

If the two deltaPitches are identical, then the maximum similarity of zero is obtained. When they are different, the similarity value gets smaller.

music database	number of musical pieces	children's song: 155 generated: 10,000 total: 10,155
	number of average notes	57.8
humming data	singer	1 male
	number of humming	67
	number of average notes	14.0

Table 1. Large-scale database.

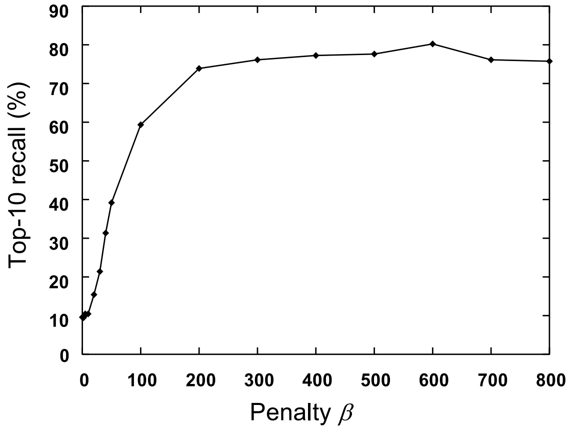

Figure 1. Retrieval result by distance-based similarity.

3.2. Experiment

In order to investigate the performance of the proposed method, a music retrieval experiment was carried out. The large-scale music database used for this experiment is shown in Table 1. The large-scale MIR system has 10,155 songs that consist of 155 children's songs and 10,000 pieces automatically generated by the trigram probabilities. The generation was performed as follows. Let the pitch and the length of two contiguous notes be $(p_{i-1},\ell_{i-1}),(p_i,\ell_i)$. Then the pitch p and the length ℓ of the next note are generated according to the trigram probabilities $P(p|p_{i-1},p_i)$ and $P(\ell|\ell_{i-1},\ell_i)$ respectively. The trigram probabilities was estimated from the 155 real songs. Therefore, the generated 10,000 songs are similar to the 155 songs, from which target songs are chosen.

The top-10 recall rates for various β are shown in Figure 1. The best result of 80.2% was obtained when $\beta = 600$.

4. QUANTIZATION-BASED MATCHING

4.1. Quantization-based similarity

Quantization-based matching (also known as contour matching) algorithm involves string-matching based algorithm employed by QBH[1]. This algorithm converts a deltaPitch sequence of input humming and songs in the database into sequences of quantized codes. Then the code sequence of the input humming is matched with that of songs in the database using an approximate string matching algorithm.

Conventional systems use quantized code such as ‘U’ (up), ‘D’ (down) and ‘S’ (same). In this paper, we express the quantized code as integer values $0, 1, \ldots, K-1$. Let center values of quantization intervals be $\mu_0, \ldots, \mu_{K-1}$. Now the quantization function $Q(x)$ is defined as follows.

$$Q(x) = \underset{k}{\operatorname{argmin}} |x - \mu_k| \tag{9}$$

Then the similarity $S_m(i,j)$ can be defined as follows.

$$S_m(i,j) = \begin{cases} 1 & \text{if } Q(h(j)) = Q(d_m(i)) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

The advantage of quantization-based method is its robustness, as it is not affected by small fluctuation of pitch frequency. The drawback of this method is that the performance of this method is greatly affected by quantization error.

4.2. Experiment

An experiment was carried out to measure the performance of quantization-based similarity. The experimental condition is the same as described in Section 3.2. For a certain K, the center value is chosen as follows.

$$\mu_i = D\left(i - \frac{K-1}{2}\right) \quad (i = 0, 1, \ldots, K-1) \tag{11}$$

D is a quantization interval. Figure 2 shows the results for quantization level $K = 3, \ldots, 11$, penalty $\beta = -1 \sim 0$ and $D = 100$. From this result, $K = 11$ and $\beta = -0.8$ are optimum and 53.3% of top-10 recall rate was obtained. Compared to the distance-based similarity (Figure 1), the quantization-based method was not effective.

Figure 3 shows the results for various D. If D is large, the number of quantization error becomes lower, but the number of ‘synonyms’ in the database becomes large. This result showed that the highest performance of 56.3% was obtained for $K = 7$ and $D = 300$.

There can be a couple of reasons that degrades the quantization-based matching. One reason is quantization errors around the quantization boundary, and the other one is octave errors caused by pitch extraction errors.

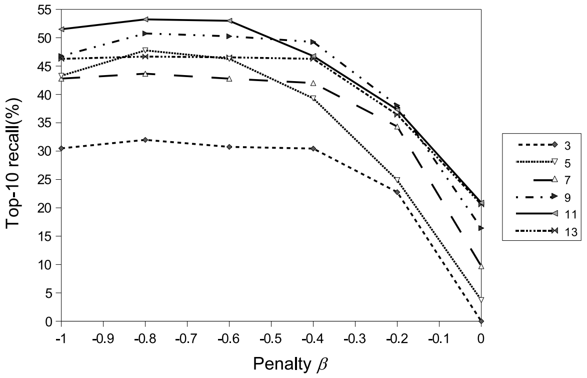

Figure 2. Retrieval result by quantization-based similarity.

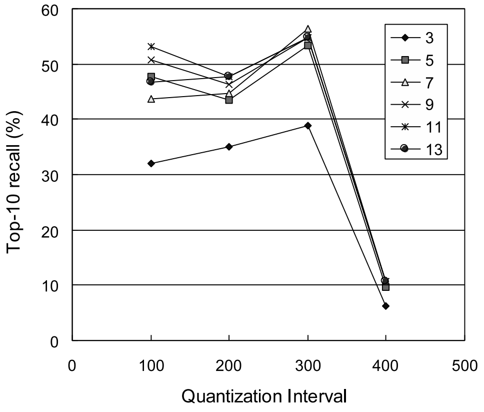

Figure 3. Retrieval result for various quantization intervals.

5. FUZZY-QUANTIZATION-BASED MATCHING

5.1. Observation of deltaPitch difference between humming and database

The pitch error between a query humming and a database was further investigated. First, correspondences between deltaPitches of an input query and that in the database were determined using DP matching. After searching for the optimal correspondences, deltaPitch differences were calculated. The number of humming that contained 863 notes was 67. Eight hundred seventy notes were detected from the query by the automatic note segmentation, and 21 insertions and 14 deletion of notes were observed.

Next, differences of the deltaPitch between the database and humming were observed. Here, if the expected deltaPitch is equal to the observed deltaPitch, the error is zero. Table 2 shows the distribution of the deltaPitch difference. From this result, it was found that almost 40% of the notes have a difference of more than 50 cents from the notes in the database. Differences more than 1000cent are 1.4% of all notes, which seem to be caused by pitch extraction error.

deltaPitch difference (cent)	#notes	ratio(%)
-50	516	60.8
50-100	216	25.4
100-200	79	9.3
200-300	12	1.4
300-400	7	0.8
400-500	1	0.1
500-600	1	0.1
600-700	2	0.2
700-800	1	0.1
800-900	0	0.0
900-1000	2	0.2
1000-	12	1.4

Table 2. Distribution of deltaPitch difference

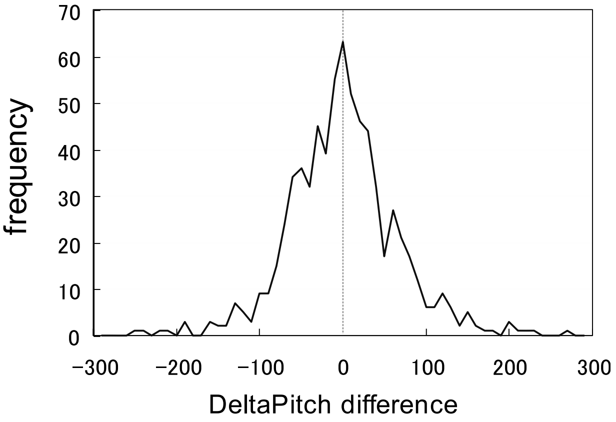

Figure 4. Histogram of deltaPitch error.

Figure 4 shows a histogram of the deltaPitch difference. Here, the y-axis represents the frequency of occurrence of deltaPitche errors whose values fall into 10 cent bin.

From these results, it is clear that it is important to deal with quantization error in order to raise the accuracy of the quantization-based method.

5.2. Membership function and fuzzy quantization

There are several ways to avoid the effect of a quantization error. The most popular method is fuzzy quantization[7]. The basic idea of fuzzy quantization is to change the feature function of the quantization into a continuous function.

First, the ordinary quantization is formulated from the feature function point of view. Let the number of quantization levels (clusters) be K, and the k-th cluster be C_k $(0 \le k \le K-1)$. The feature function of the quantization $f(x,k)$ is defined as

$$f(x,k) \equiv \begin{cases} 1 & \text{if } x \in C_k \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Using $f(x,k)$, continuous value x is categorized into one of the quantization levels. On the other hand, the fuzzy quantization method uses a membership function instead of a feature function. A membership function $R(x,k)$ maps the input x into a continuous value from 0 to 1.

There are many possibilities to construct $R(x,k)$. In this paper, the membership function is constructed using a probabilistic framework. Let us assume that the deltaPitches that corresponds to the level C_k in an input humming follows a certain distribution $\phi_k(x)$. Now, the membership function $R(x,k)$ is calculated as follows.

$$R(x,k) \equiv \frac{\phi_k(x)}{\sum_{i=0}^{K} \phi_i(x)} \tag{13}$$

Here, $R(x,k)$ is equivalent to *a posteriori* probability of quantization level C_k given x under an assumption that the occurrence probability of C_k is uniform.

Next, the distribution function $\phi_k(x)$ has to be decided. If we assume that the distribution function is independent from k except the mean value, $\phi_k(x)$ can be calculated as

$$\phi_k(x) = \phi(x - \mu_k) \tag{14}$$

where $\phi(x)$ is a distribution function whose mean is zero. As $\phi(x)$ is independent from k, it becomes optimal when the distribution function properly models the distribution shown in Figure 4. From an observation of the distribution in Figure 4 that the center is sharp and around the edge is smooth, a distribution of Figure 4 seems to be modeled by Laplace distribution rather than Gaussian distribution.

The density function of the Laplace distribution is a typical supergaussian distribution. The density function of Laplace distribution is given as follows.

$$\phi(x) \equiv \frac{1}{2v} \exp\left\{ -\frac{|x|}{v} \right\} \tag{15}$$

where v is a parameter that is related to the variance of the distribution.

To generalize Gaussian and Laplace distributions, we introduce another parameter γ into the density function to control the kurtosis of the distribution. The distribution function is

$$\phi(x) \equiv \frac{\gamma}{2\Gamma\left(\frac{1}{\gamma}\right) v^{\frac{1}{\gamma}}} \exp\left\{ -\frac{|x|^\gamma}{v} \right\} \tag{16}$$

where γ is related to the kurtosis of the distribution. Figure 5 shows some examples of density functions for various v and γ. When $\gamma = 1$, this distribution function is identical to Laplace distribution. On the other hand, when $\gamma = 2$ we obtain Gaussian distribution.

From this distribution function, the membership function R is calculated according to formula (13). Figure 6 shows examples of membership functions when $K = 9, \gamma = 1$ and $v = 20$.

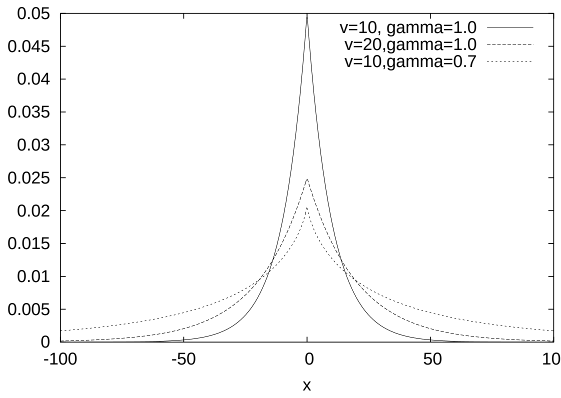

Figure 5. Examples of distributions for various v and γ.

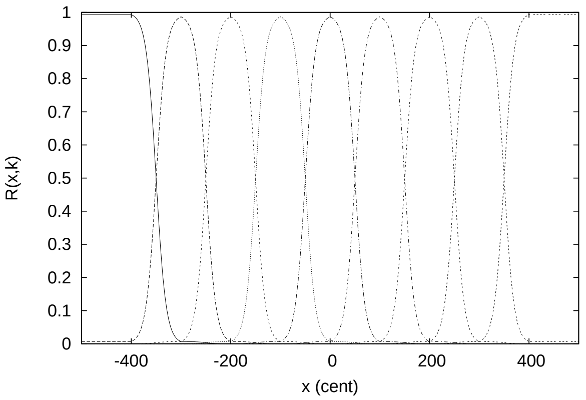

Figure 6. An example of membership functions.

Finally, the value of the membership function that corresponds to $h(j)$ become a similarity.

$$S_m(i,j) = R(h(j), Q(d_m(i))) \qquad (17)$$

5.3. Experiment

The above-mentioned similarity is examined by the retrieval experiment. The experimental condition is the same as described in 3.2. In this experiment, quantization level K was set to 9 and quantization interval D is set to 100. First, penalty value β was optimized. Figure 7 shows the result for $v = 10$ and $\gamma = 0.5$. The optimum β was around -0.2 in this result. Therefore $\beta = -0.2$ was used hereafter. Figure 8 shows the top-10 recall rate for various v and γ. From this result, it is found that the parameters are optimum around $v = 50$ and $\gamma = 0.8$. At the optimum point, a recall rate of 76.1% was obtained.

6. EXPERIMENT WITH FIVE USERS

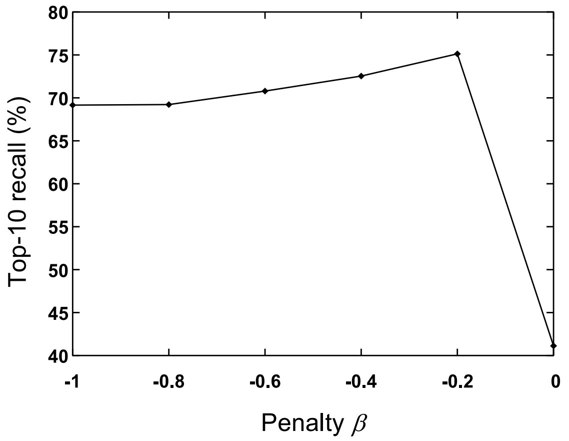

Figure 7. Retrieval result by fuzzy-quantization-based similarity.

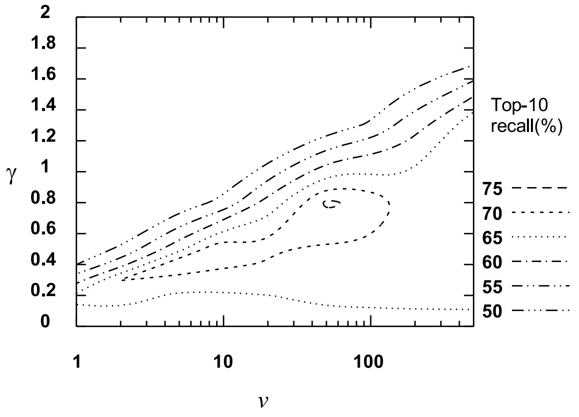

Figure 8. Recall rates for various v and γ.

Next, the above three similarities are compared using the humming data sung by five users. The experimental conditions are shown in Table 3. Various parameters are set to the optimal values obtained in the previous sections. For the distance-based method, β was set to 600. For the quantization-based method, D was set to 300, K was set to 7 and β was set to -0.8. For the fuzzy quantization based method, D was set to 100, K was 9, β was -0.2, v was 50 and γ was 0.8.

Figure 9 shows the experimental results of five users. The recall rate was different from user to user. Distance-based method showed the best performance and fuzzy-quantization-based method was the next. The average recall rate for top-10 candidates was 65% by the distance-based method.

To improve the recall rate, we tried to combine the distance-based and fuzzy-quantization-based similarities. In this experiment, similarity between the input humming and the m-th song in the database was calculated as

$$V_m = \lambda V_m^{dist} + (1-\lambda) V_m^{FQ} \qquad (18)$$

where V_m^{dist} and V_m^{FQ} are similarities obtained by the distance-based and fuzzy-quantization-based method

music database	number of musical pieces	children's song: 155 generated: 10,000 total: 10,155
	number of average notes	57.8
humming data	singer	5 males
	number of humming	320
	number of average notes	11.7

Table 3. Large-scale database.

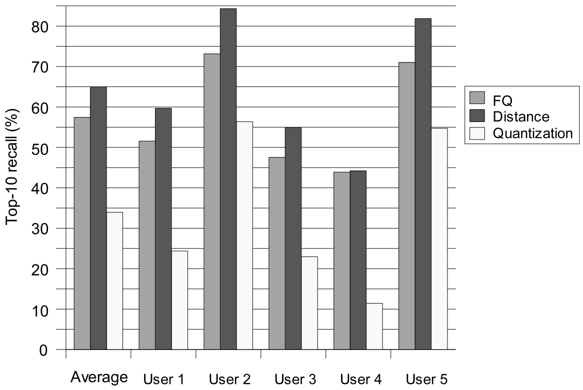

Figure 9. Retrieval result for five users.

respectively. Figure 10 shows the recall rate for various λ. This result is an average recall rate for five users. By combining these scores, about a 1 point improvement was obtained.

7. CONCLUSION

In this paper, we compared three kinds of similarity measures through DP-matching based query-by-humming music retrieval. The compared representations are distance-based, quantization-based and fuzzy-quantization based representations. From the experimental result, the distance-based method gave the best recall rate. Besides, we examined the combination of distance-based and fuzzy-quantization-based similarity. The experimental result showed that the recall rate was improved by the combination.

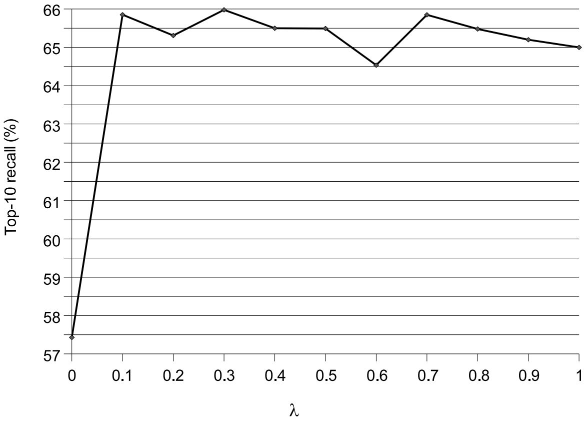

Figure 10. Retrieval results of the combined score.

8. REFERENCES

[1] A. Ghias, J. Logan, D. Chamberlin and B. C. Smith, *Query by Humming: Musical Information Retrieval in an Audio Database*, Proc. ACM Multimedia, 1995.

[2] R. J. McNab, L. A. Smith, D. Bainbridge and I. H. Witten, *The New Zealand Digital Library MELody inDEX,* D-Lib Magazine, May, 1997.

[3] T. Sonoda and Y. Muraoka, *A WWW-based Melody Retrieval System–An Indexing Method for A Large Database–*, Proc. ICMC, 2000.

[4] J.-S. Roger Jang, J.-C. Chen and M.-Y. Kao, *MIRACLE: A Music Information Retrieval System with Clustered Computing Engines,* Proc. ISMIR, 2001.

[5] S.-P. Heo, M. Suzuki, A. Ito, S. Makino and H. Chung, *Multiple pitch candidate based music information retrieval method for query-by-humming*, Proc. Int. Workshop on Adaptive Multimedia Retrieval, 189–200, 2003.

[6] S. Nakagawa, *Connected Spoken Word Recognition Algorithm by Constant Time Delay DP, O(n) DP and Augmented Continuous DP Matching,* Information Sciences , 33 , 63–85, 1984.

[7] L. A. Zadeh , ”Fuzzy sets,” Inform. Contr., Vol. 8, pp. 338-353, 1965