World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:3, No:12, 2009

Practical Method for Digital Music Matching Robust to Various Sound Qualities

Bokyung Sung, Jungsoo Kim, Jinman Kwun, Junhyung Park, Jihye Ryeo, and Ilju Ko

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf

***Abstract*—**In this paper, we propose a practical digital music matching system that is robust to variation in sound qualities. The proposed system is subdivided into two parts: client and server. The client part consists of the input, preprocessing and feature extraction modules. The preprocessing module, including the music onset module, revises the value gap occurring on the time axis between identical songs of different formats. The proposed method uses delta-grouped Mel frequency cepstral coefficients (MFCCs) to extract music features that are robust to changes in sound quality. According to the number of sound quality formats (SQFs) used, a music server is constructed with a feature database (FD) that contains different sub feature databases (SFDs). When the proposed system receives a music file, the selection module selects an appropriate SFD from a feature database; the selected SFD is subsequently used by the matching module. In this study, we used 3,000 queries for matching experiments in three cases with different FDs. In each case, we used 1,000 queries constructed by mixing 8 SQFs and 125 songs. The success rate of music matching improved from 88.6% when using single a single SFD to 93.2% when using quadruple SFDs. By this experiment, we proved that the proposed method is robust to various sound qualities.

***Keywords*—**Digital Music, Music Matching, Variation in Sound Qualities, Robust Matching method.

I. Introduction

THE way in which music is distributed has changed from physical distribution in the form of record discs to online distribution in the form of digital music files. Furthermore, digital audio devices and interfaces have been fragmented and diversified under changing consumption patterns of music content. This indicates the time to require supplying digital music files with novel advanced service. The most important parts of advanced services are personalization, improved user serviceability, automation, and recommendation.

Music matching has been considered as a core technique for providing desired services. This technique has been researched and applied to recognize user music content through various methodologies. Thus far, the direction of related research has changed from text-based matching in an early stage to content-based matching. However, content-based music matching has a drawback of being susceptible to changing sound quality. In other words, music matching is currently an immature technique. Consequently, an improved technique of music matching that is robust to various sound qualities is required for overcoming the abovementioned drawback. In this paper, we propose a proved music matching method that is robust to various sound qualities.

The proposed method is a practical method for digital music matching and is robust to varying sound qualities in a music file. It consists of two parts: client and server.

The client part consists of the input, preprocessing, and feature extraction modules. First, the input module takes accepts MP3 files that are to be matched with identical music. Even though the inputted music files contain the same song, there exists a high possibility that each file has a different sound quality format (SQF). In such cases, the hashing technique, which is generally used in the string-matching field, cannot yield good music matching. Further, in the preprocessing module, the input music is converted to a standard format and its SQF is extracted simultaneously. Next, delta-grouped Mel frequency cepstral coefficients (MFCCs) are applied to it in order to extract consistent features, regardless of its changing SQF. This extraction algorithm embodies two steps: the grouping of MFCCs. These two steps facilitate simplification by dimensionality reduction and emphasizing the changing tendency. By these characteristics, delta-grouped MFCCs overcome the problems of generation of large amounts of data and susceptibility to changing sound quality.

The server part involves the following steps: parsing a query, selecting a sub feature database (SFD), and matching queries. A query from the client is divided into an SQF value and feature series. The SQF from the client query is compared with the sound format index of an FD for determining the SFD made by format of minimum gap of sound quality. The FD is constructed from individual databases of various SQFs. Therefore, it has a group of SFDs corresponding to each assisted format. The matching module browses the selected SFD by comparing FD with feature data of input query. As a comparison criterion, Euclidean distance (ED) is used. The content presenting the minimum ED (MED) is the final matched result.

The rest of this paper is organized as follows: In section 2, we present related studies on music matching. In section 3, we

B. Sung Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : ivsinger@ssu.ac.kr),
J. Kim Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : dotline@ssu.ac.kr),
J. Kwon Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : lovekpo@gmail.com),
J. Park Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : kaga@ssu.ac.kr),
J. Ryeo Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : hoya350@ssu.ac.kr),
I. Ko Corresponding Author is with the Department of Media, Soongsil University, Seoul, Korea (e-mail : andy@ssu.ac.kr).

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf

present the detailed structure of our proposed digital music matching method. In section 4, we describe our experiments and results. Finally, in section 5, we provide conclusions and suggest future studies.

II. Related Studies

Research on music matching has been mainly progressing in two directions: One approach is to use music related text information: the other approach is to use information extracted from the music signal itself.

The research of text-based music matching is as follows. This methodology was originally used in text-based information retrieval fields [1], [2]. At present, it is being adopted and applied to music matching. Further, the error rate of music matching has been increasing due to the growing complexity of the music domain. This implies a technical limitation in providing various desired user services. As an example, text information with wrong filled information is one of the main causes of matching errors.

Content-based music matching [3], [4] has been proposed to overcome the problems of text-based music matching.

This field emerged as a result of on music information retrieval (MIR) [5]-[7] techniques. Low level signal processing techniques have been researched in the MIR field for acquiring abstract information from a music signal. Research on music matching has been staying these.

The fundamental steps of content-based music matching are as follows: preprocessing, extracting features by various algorithms, and retrieving a database.

Preprocessing implies minimal compensation against transforming errors that can occur in various digital environments. Some studies on this aspect aim to solve problems such as added noise and degradation of sound quality. Noise filtering [8], [9] has been studied in order to control noise easily. Normalization [10], [11] is carried out so as to minimize statistical errors in handling a large volume of data. In addition, decoding has been carried out with a focus on optimized content convertibility for end-user applications. Onset [12], [13] has been studying on fields need correct segmentation.

Feature extraction algorithms can be subdivided into time-based algorithms, frequency-based algorithms, and time-and-frequency-based. Examples of time-based algorithms are the zero-crossing rate (ZCR) algorithm [14], [15], root mean square (RMS) algorithm [16], etc. Examples of frequency-based algorithms are the linear predictive coding (LPC) algorithm [17]-[19], MFCC algorithm [20]-[23]; and algorithms based on the spectral centroid [24], spectral flux [25], and spectral roll-off [26]. Recently, a combined algorithm has also been studied as a complementary approach to the time-and-frequency-based method. A concise definition of each of these algorithms is provided in the following sections.

The ZCR is the rate of sign-changes along a signal. This feature has been used extensively in both speech recognition and MIR.

The RMS is a statistical measure of the magnitude of a varying quantity. It is particularly useful when a variant assumes positive and negative values. Further, it can be calculated for a series of discrete values or for a continuously varying function.

LPC is a tool used mostly in audio signal processing and speech processing for representing the spectral envelope of a digital signal of speech in a compressed form, using the information obtained using a linear predictive model. It is one of the most powerful speech analysis techniques and one of the most useful methods for encoding good-quality speech at a low bit rate, and it provides extremely accurate estimates of speech parameters.

The spectral centroid is a measure used in digital signal processing to characterize a spectrum. It indicates where the “center of mass” of the spectrum is. Perceptually, it has a robust connection with the impression of “brightness” of sound. Further, it is calculated as the weighted mean of the frequencies present in a signal, with their magnitudes as the weights.

Spectral roll-off is one of the measures of a spectral shape. The Frequency below which a particular percentage (about 85%) of the power spectrum is located is considered a s the roll-off point.

Spectral flux indicates the changes in a spectral shape. It is usually calculated as the 2-norm between two spectra. It can be used to determine the timbre of an audio signal, or in onset detection, among other applications.

III. Practical Digital Music Matching

A. Proposed System

Fig. 1 shows the overall structure of a practical digital music matching system that is robust to various sound qualities. The system flow is as follows: The input module receives MP3 files. The SQF and a series of delta-grouped MFCCs are extracted from the inputted music by performing the preprocessing and extracting feature steps. Then, these SQF and MFCCs are constructed as a query and sent to the server. Further, the parsing module breaks down the query into the original data. Next, the SFD of the closest sound quality is selected by using the SQF. The matching module browses selected SFD with a feature series of the query. Then, the matched result is sent and displayed to the client.

In this study, we adopt our modified MFCCs as a feature extraction algorithm. Further, MFCCs have been popularly used in analyzing and recognizing various audio signals because they have shown high performance. However, they have mainly two problems, which are discussed below.

First, MFCCs are sensitive to change in sound quality. The features extracted from identical songs with different SQFs have a value gap. The value gap becomes severe with increasing difference in sound qualities. In order to solve this problem, we build an FD as a set of SFDs. A previously developed matching system employs the FD of a single SFD. Length of sound quality of section considers characteristics of application domain to choose appropriate formats.

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf

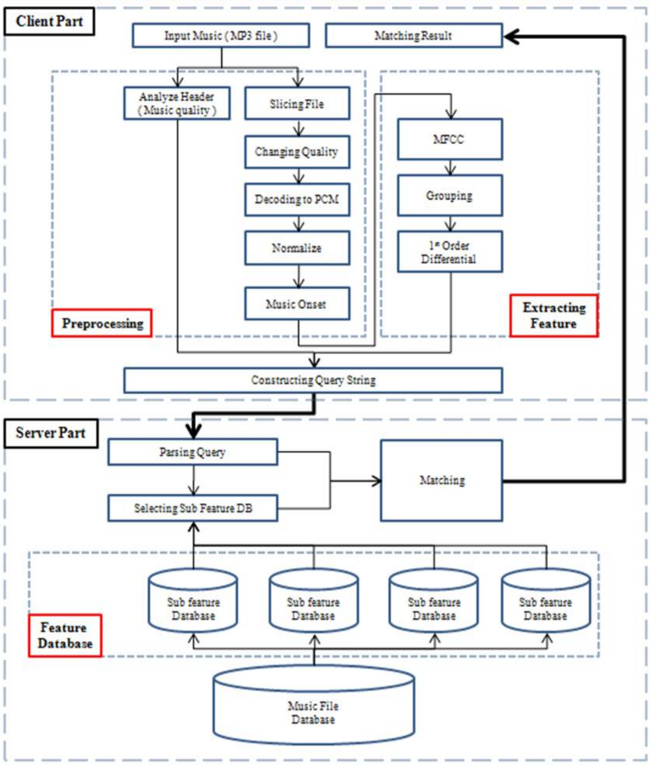

Fig. 1 Proposed structure for digital music matching robust to various sound qualities. It has two main parts: client and server level. The client part consists of the preprocessing, feature extraction, and query string construction modules. The server part consists of the query parsing, SFD selection, and matching modules.

Second, the volume of data generated increases rapidly if a large number of input signals are used for achieving high accuracy. Extending the window size and reducing the hop size are general approaches to solve this problem. It is more effective to first extract a large number of feature series from short frames and then simplify the extracted series than simultaneously extracting and simplifying a small number of feature series from long frames. This is because in the case of the latter approach, the loss rate of the available information is higher than that in the case of the former approach. In the proposed system, a series of original MFCCs is simplified by grouping and first order differential.

B. *Client Part*

The client part consists of two modules: a preprocessing module and a feature extraction module.

The preprocessing module obtains the SQF from the header of a music file and revises the music signal. Preprocessing includes slicing the input music file, changing the quality of the input music, decoding the input music to PCM, and then normalizing it. The slicing step extracts necessary parts from music signals. The step “changing quality” converts the input music to one of the SQFs in order to minimize the quality loss. The step “decoding to PCM” transforms the compressed music into decompressed music signals. In this study, we initialize the decoding format with 44100Hz sampling, 16bit quantizing, and mono channels. Here, normalization is applied to each short frame. Through this, the contrast attribute of short frames is emphasized. Further, the music onset module removes the first silence signals from the input music. Silence signals are frequently inserted during the digitization process. The length of the first silence signal changes whenever music is converted from one format to another. Then changed amount is accumulated by randomizing form. The difference between the first silence signals of identical song files is one of the main factors adversely affecting the matching success rate. Therefore, it is necessary to apply the music onset to solve this problem. The transition point between a silence signal and a music signal is detected by analyzing patterns of calculated series of the ZCR value.

The feature extraction module executes three steps and produces a series of delta-grouped MFCCs. This module involves extracting MFCC sequences, grouping MFCCs, and first order differential between groups. Grouping and delta help simplify of feature sequences and emphasize the simplified patterns, respectively. The MFCC algorithm is a feature extraction algorithm designed on the basis of a model of the human hearing system. Its standard version is defined as the following sequence of four steps: First, a signal from the time domain is transformed into the frequency with the standard window size and hop size. Second, a Mel-filter bank (MFB) is applied to frequency data for increasing the analysis density at frequencies less than 1 KHz. This is because the human ear is relatively more sensitive to sounds with frequencies less than 1 KHz. Third, filtered data takes a log-like human sound recognition form. Fourth, the logged data takes DCT to minimize the correlation between output nodes. In this study, the window size and hop size were set to 5 ms and 2.5 ms, respectively. We designed an MFB to obtain 12 features from 1 to 12 orders and fixed the block size of grouping to 50 frames. Then, with these, we extracted 80 delta-grouped MFCCs from one music file. Fig. 2 shows that the MFCCs are simplified and emphasized by grouping MFCCs and calculating delta.

A query string constructs a set of bit-rate indices, the sampling frequency, BPS, and 80 delta-grouped MFCCs.

C. *Server Part*

The server part consists of three modules: query parsing, SFD selection, and matching. A query from the client is divided into an SQF and a feature series by the parsing module. Each one is inputted to selecting SFD module and matching module.

The SFD selection module determines an appropriate SFD in the FD by using the parsed SQF of the client query. The closest SQF is a judgment rule to choose appropriate SFD. The SFD determined by the selected SQF is used by the matching module.

The extracted feature series are retrieved in the selected SFD. The retrieval criterion is ED. A candidate group of high-rank records that have a small ED is selected by using the select syntax. The final matched record that has the MED is determined by comparing the ED between candidate records.

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf

At this time, the MED must satisfy the initial critical states. Feature from contents are not distributing limited space and changing range of it is broad. That is, it is difficult to determine the absolute values of critical states. The pattern of ED differences from the first to the third candidate is used as a judgment rule of critical states.

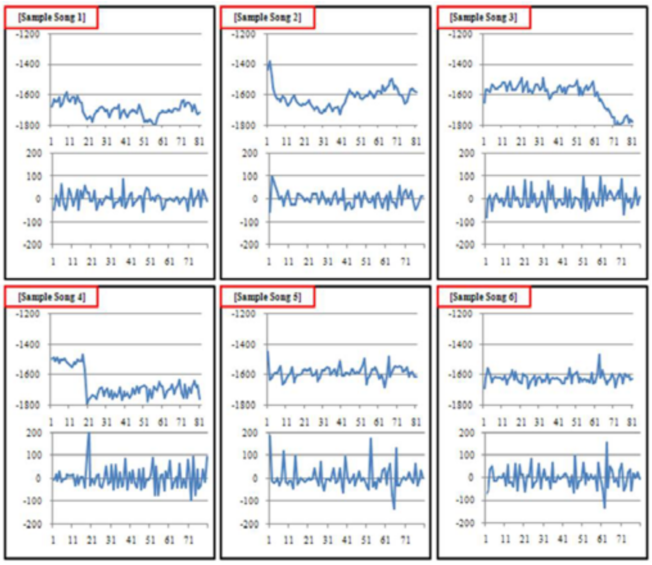

Fig. 2 Comparison of feature tendency of feature extracting steps. Each panel contains graphs of grouped MFCCs and delta-grouped MFCCs of 6 sample songs. The upper and lower graphs of each sample song show groped MFCCs and delta-grouped MFCCs extracted from the sample, respectively. The song from sample song1 to sample song 6 is "She said she said" by the Beatles, "If i ain't got you" by the Alicia Keys, "Into the new world" by the Girls generation, "Hungry spider" by the Makihara noriyuki, "Rak tae doo lae mai dee" by the Potato and "Je suis venue te dire que je m'en vais" by the Monica Nonueira, respectively.

IV. Experiments & Results

For confirming the robustness of the proposed system, the used music server was constructed and query matching was carried out as follows. The music server was constructed with 1,000 randomly selected MP3 songs that had a bit-rate of 320 kbps. For constructing queries, we used 125 songs exist in music server. Next, we generated 1,000 MP3 files for query by converting these 125 songs into 8 MP3 formats (64 kbps, 80 kbps, 96 kbps, 128 kbps, 160 kbps, 192 kbps, 256 kbps and 320 kbps).

We performed an experimented to confirm that the proposed system is effective in increasing the success rate of matching. We considered three cases: single SFD structure, double SFD structure, and quadruple SFD structure. Then, we measured the success rates and speeds of matching for the three cases by using the same query set.

In the case of a single SFD, the total contents are used in only one format 64Kbps. The double SFD structure employed a bit-rate of 6 kbps for input files having bit-rates less than 128 kbps and 160 kbps for input having bit-rates exceeding 160 kbps. The quadruple SFD structure employed a bit-rate of 64 kbps for input files having bit-rates of less than 96 kbps, 96 kbps for input files having bit-rates between 159 kbps and 96 kbps, 160 kbps for input files having bit-rates between 255 kbps and 160 kbps, and 256 kbps for input files having bit-rates exceeding 256 kbps. Here, 1,000 query music files obtained by mixing 8 BPS formats and 125 songs were used for each experimental case. Thus, the entire experiment involved 3,000 instances of music matching.

Table 1 shows experimental results for the 3 cases. These results indicate that sound quality of input music does not always vary directly with the success rate of matching. Query amount of this experiment is not absolutely huge that regards result ratio as the convergent point. Therefore, it is more accurate to compare the results of average matched rates obtained in each case than to consider the correlation among the elements. From this point of view, use of the FD having an SFD structure causes an increase in the success rate of matching.

Fig. 3 shows that each matching time is similar with average time, except that for 80 kbps in case 2. At the bit-rate 80 kbps of case 2, the processing time of the step "changing quality" is longer than that at other bit-rates. Yet, this is not a high correlated fact to core algorithm of proposed method. This implies that this fact can be considered as an exceptional situation. Thus, the average matching time of the entire process is approximately 0.925s.

TABLE I
Matching success rate For three cases

Query BPS	Case 1	Case 2	Case 3
64	84.8 %	84.8 %	84.8 %
80	81.6 %	81.6 %	81.6 %
96	79.2 %	79.2 %	84.8 %
128	99.2 %	99.2 %	100 %
160	90.4 %	97.6 %	97.6 %
192	92.0 %	99.2 %	99.2 %
256	90.4 %	97.6 %	98.4 %
320	91.2 %	98.4 %	99.2 %
Average	**88.60 %**	**92.2 %**	**93.2 %**

The total processing time is divided into preprocessing time, feature extraction time, and matching time. Fig. 4 shows comparing the ratio of the processing time of each step to the total processing time. We can observe that each step has a steady time rate. This implies that the proposed system guarantees consistent processing time, regardless of query states.

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf

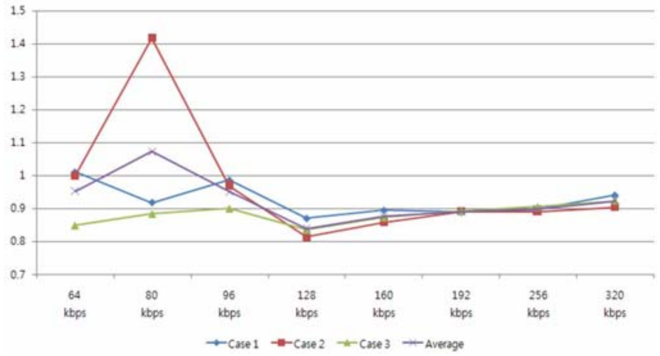

Fig. 3 Average matching time required by the entire process with 8 sound quality formats. Unit of Y-axis is s. Each point indicates the average matching time of 125 songs of the same format. Average means average time from case 1 to case 3.

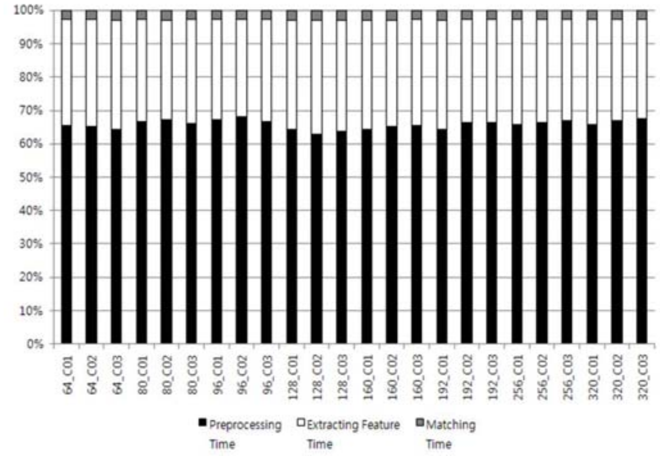

Fig. 4 Processing time ratio of sub-steps in each case. The Y-axis shows the accumulated ratio from 64 kbps in case 1 (indicated as 64_C1 in figure) to 320 kbps in case 3 (320_C3). Each element of the X-axis is obtained from 24 query states by mixing 8 formats in 3 cases.

V. Conclusion

In this paper, we proposed a practical method for digital music matching that is robust to various sound qualities of music. For extracting feature data from music, we used delta-grouped MFCCs. Further, we constructed an FD from a set of SFDs. We applied these elements to the proposed system and confirmed that it has a high matching success rate.

With regard to future studies, we have two study plans for developing of the proposed system. The first plan is to conduct a new experiment with large music files. We believe that this extended experiment will clearly confirm the improvement in the performance of the proposed system. The second plan is to conduct research on indexing of the FD. At present, the retrieval time is proportional to the volume of music contents. We expect to decrease the retrieval time.

Acknowledgment

The authors thank Changhwan Choi (MOG Interactive, Korea), who gives us a great experimental environment: music samples and music server.

The work described in this paper was funded by the Brain Korea 21 Project of Media department of Soongsil University in 2009

References

[1] G. Skobeltsy, T. Luu, I. P. Zarko, M.

[2] Rajman and K. aberer, “Query-Driven Indexing for Peer-toPeer Text Retrieval,” Proc. 16th International World Wide Web Conference, Canada, 2007, pp. 1185-1186.

[3] B. Stein, “Fuzzy-Fingerprints for Text-Based Information Retrieval,” Proc. I-KNOW 05th, 2005, pp. 572-579.

[4] Y. Peng, C.W. Ngo, C. Fang, X. Chen and J. Xiao, “Audio Similarity Measure by Graph Modeling and Matching,” Proc. 14th annual ACM international conference on Multimedia, USA, 2006, pp. 603-606.

[5] F. Kurth and M. Muller, “Efficient Index-Based Audio Matching,” IEEE Trans. Audio, Speech and Language processing, vol. 16, no. 2, pp. 382-395, Feb. 2008.

[6] P. Roos and B. Manaris, “A Music Information Retrieval Approach Based on Power Laws,” Proc. 19th IEEE ICTAI, Greece, Oct. 2007, pp. 29-31.

[7] Z. W. Ras, X. Zhang and R. Lewis, “MIRAI: Multi-hierarchical, FS-Tree Based Music Information Retrieval System,” LNAI 4585, pp. 80-89, 2007.

[8] M. I. Mandel, D. P. W. Ellis, “Multiple-Instance Learning for Music Information Retrieval,” Proc. ISMIR 2008.

[9] S. Wabnik, G. Schuller, J. Hirschfeld and U. Kraemer, “Different quantization noise shaping methods for predictive audio coding,” Proc. IEEE International Conference on Acoustics, Speech and Signal processing, France, 2006.

[10] M. Park, H. R. Kim and S. H. Yang, “Frequency-Temporal Filtering for a Robust Audio Fingerprinting Scheme in Real-Noise Environments,” ETRI Journal, vol. 28, no. 4, pp. 509-512, Aug. 2006.

[11] S. Hamawaki, S. Funasawa, J. Katto, H. Ishizaki, K. Hoashi and Y. Takishima, “Feature Analysis and Normalization Approach for Robust Content-Based Music Retrieval to Encoded Audio with Different Bit Rates,” LNCS 5371, 2008.

[12] D. Giuliani, M. Gerosa and F. Brugnara, “Improved automatic speech recognition through speaker normalization,” Computer Speech and Language, vol. 20, pp. 107-123, 2006.

[13] C. C. Toh, B. Zhang and Y. Wang, “Multiple feature fusion based onset detection for solo singing voice,” Proc. ISMIR 2008.

[14] R. Zhou and J. D. Reiss, “Music Onset detection combining energy-based and pitch-based approaches,” Proc. MIREX Audio Onset Detection Contest, 2007.

[15] W. Pan, Y. Yao, Z. Liu and W. Huang, “Audio Classification in a Weighted SVM,” Proc. ISCIT07, 2007.

[16] A. J. eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho and J. Huopaniemi, “Audio-Based Contest Recognition,” IEEE Trans. Audio, Speech and Language processing, vol. 14, no. 1, pp. 321-329, Jan. 2006.

[17] A. Farina, “Assessment of Hearing Damage when listening to music through a personal digital audio player,” Journal of the Acoustical Society of America, 2008.

[18] J. E. M. Exposito, S. G. Galan, N. R. Reyes and P. V. Candeas, “Adaptive network-based fuzzy inference system vs. other classification algorithms for warped LPC-based speech/music discrimination,” Engineering Applications of Artificial Intelligence Journal, vol. 20, pp. 783-793, 2007.

[19] M. K. S. Khan and W. G. A. Khatib, “Machine-learning based classification of speech and music,” Multimedia Systems Journal, vol. 12, no. 1, pp. 55-67, 2006.

[20] H. Zhou, A. Sadka and R. M. Jiang, “Feature Extraction for Speech and Music Discrimination,” Proc. 6th International Workshop on Content-Based Multimedia Indexing, UK, Jun. 2008.

[21] G. J. A. Hunter and K. Zienowicz and A.I Shihab, “The Use of Mel Cepstral Coefficients and Markov Models for the Audomatic Identification, Classification and Sequence Modeling of Salient Sound

Events Occurring During Tennis Matches," Journal of the Acoustical Society of America, vol. 123, issue. 5, pp. 3431, 2008.

[22] K. M. Indrebo, R. J. Povinelli and M. T. Johnson, "Minimum Mean-Squared Error Estimation Mel-Frequency Cepstral Coefficients Using a Novel Distortion Model," IEEE Trans. Audio, Speech and Language processing, vol. 16, no. 8, pp. 1654-1661, Nov. 2008.

[23] A. H. Nour-Eldin and P. Kabal, "Mel-Frequency Cepstral Coefficient-Based Bandwidth Extension of Narrowband Speech," Proc. InterSpeech, Brisbane, 2008.

[24] N. Sato and Y. Obuchi, "Emotion Recognition using Mel-Frequency Cepstral Coefficients," Journal of Natural Language Processing, vol. 14, no. 4, pp. 83-96, 2007.

[25] J. Bergstra and N. Casagrande, "Aggregate features and Adaboost for music classification," Machine Learning Journal, vol. 65, no. 2-3, pp. 473-484, Dec. 2006.

[26] E. Schubert and J. Wolfe, "Does Timbral Brightness Scale with Frequency and Spectral Centroid?," ACTA Acoustica United with Acoustica, vol. 92, pp. 820-825, 2006.

[27] T. Li and M. Ogihara, "Content-based music similarity search and emotion detection," Proc. IEEE International Conference on Acoustic, Speech and Signal processing, France, 2006.

Open Science Index, Computer and Information Engineering Vol:3, No:12, 2009 publications.waset.org/9344.pdf