Query by Humming System Through Multiscale Music Entropy

Trisiladevi C. Nagavi and Nagappa U. Bhajantri

Abstract Query by humming (QBH) is one of the most active areas of research under music information retrieval (MIR) domain. QBH employs meticulous approaches for matching hummed query to music excerpts existing within the music database. This paper proposes QBH system based on the estimation of multiscale music entropy (MME). The proposed technique exploits the statistical reliability through the MME for music signals approximation. Further, the Kd tree is employed for indexing MME feature vectors of music database leading to reduced search space and retrieval time. Later, MME feature vectors are extracted from humming query for recognition and retrieval of the corresponding song from music database. The experimental results demonstrate that the proposed MME and Kd tree-based QBH system provides higher discrimination capability than the existing contemporary techniques.

Keywords Entropy · Kd tree · Multiscale music entropy (MME) · QBH

1 Introduction

The contemporary personalized ways of music processing techniques are changing the nature of music dissemination allowing ardent music lovers to get access to large music collections widespread over genres and artists. Thus, various researchers have contributed in distinct ways for the growth of music

T.C. Nagavi (✉)
Deptartment of Computer Science and Engineering,
Sri Jayachamarajendra College of Engineering, Mysore, India
e-mail: tnagavi@yahoo.com

N.U. Bhajantri
Department of Computer Science and Engineering,
Government Engineering College, Chamarajanagar, India
e-mail: bhajan3nu@gmail.com

dissemination leading to the invention of MIR systems. The MIR research is exploring new technique referred as QBH system for music database search using humming a short music excerpt.

So far, most of the QBH research efforts [1–7] have focused on building systems based on dynamic programming [7], hierarchical filtering [1], iterative deepening melody alignment [2, 6], time series alignment [3], dynamic time warping (DTW) [4], wavelet transforms, and envelope transforms [5] approaches. Although these techniques exhibit good experimental performance, there is a need of consistency estimation in understanding the nature of a music signal. Also to conquer the inadequacy of using big knowledge base, music indexing is being used.

The growing scientific and commercial interest in audio and music processing is seen recently in the works of various MIR applications. They are based on pitch [8, 9], rhythm [10], and note [5] estimation. In this paper, we are proposing new music melody estimation approach based on multiscale entropy (ME). Also we present, Kd tree-based music melody database indexing structure for efficient and reduced space searching.

This paper is organized as follows. In Sect. 2, comprehension of related contributions is presented. The brief view of the proposed QBH system is described in Sect. 3. The entropy estimation procedure, Kd tree-based indexing, query processing, and searching are presented in Sects. 4, 5, and 6, respectively. The results and discussions are portrayed in Sect. 7. Finally, we draw the conclusions in Sect. 8.

2 Comprehension of Related Contributions

The music signals often exhibit complex structure of regularity. Through the regularity estimation, it is possible to perceive the characteristics of the music signals. In the past, several applications are developed based on regularity estimation through entropy such as heart rate analysis [11], music beat tracking [12], traffic flow analysis [13], human gait dynamics [14], and measuring predictability in networks [15].

The effectiveness of these applications motivates us to exploit the potential of regularity estimation techniques for developing QBH systems. Several methods [12, 16–19] on entropy estimation have been developed to detect specific pattern in music signals. The task of beat tracking is popularly used mechanism in music information retrieval (MIR). Davies and Plumbley [12] have explored a new method for the evaluation of beat tracking systems.

Another work [16] focuses on music retrieval based on emotion recognition. The system enhances maximum entropy with Gaussian and Laplace priors to model features for music emotion recognition. Authors were successful in getting better results for maximum entropy with priors. An interdisciplinary research [17] was proposed to explore the concept of entropy in jazz improvisation. In order to

measure the improvisation, they adapted standard melodic, harmonic, and rhythmic entropy analysis techniques.

Subsequently, the research work [18] addresses matching of music performances of the same piece of music, making it suitable for monitoring applications. The entropy-based audio finger printing was employed to represent music in the form of symbolic strings. The system was able to correctly identify Pop music with DTW, edit distance (ED), and longest common subsequence (LCS) distance measures in less than a second per comparison.

An information-based approach [19] was proposed for meaningful musical events analysis and recognition. The entropy derived from a predictive model of music corresponds to important feature of music. Also, neural network model was employed to estimate instantaneous entropy for music. Even the relationship between listener's subjective sense of tension and entropy is considered in the work.

The majority of the music processing systems have focused on developing holistic single sample entropy-based techniques for capturing regularity in the music signals. Nevertheless, they cannot capture the long-term structures and large amount of information present in the music signals. In regard to this disadvantage, we are proposing MME technique, which uses single sample entropies of a music signal at multiple scales to uncover several types of structure.

3 Proposed Query by Humming System

In order to illustrate the proposed system of QBH, let us consider distinguished training and testing phases. Each of these phases perform different operations on the input signal such as preprocessing, framing, entropy estimation, Kd tree-based indexing, and searching. Through the simplified block diagram of a general system as shown in Fig. 1, the proposed QBH system's steps are discussed.

3.1 Preprocessing

Preprocessing is applied on the MP3 songs database to remove nonessential information, for down sampling and converting to mono channel [20].

3.2 Framing

In framing, the music signal is split into several frames, such that each frame is analyzed in short time instead of analyzing the entire signal at once [20]. In our experiments, the frame length is set to 25 with 10 ms overlap between two adjacent frames.
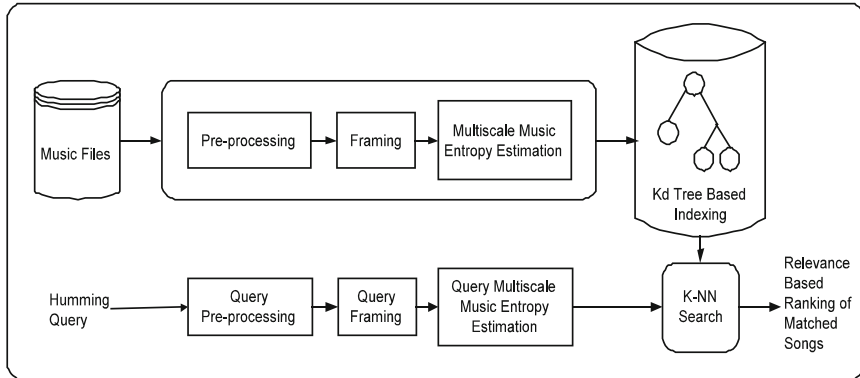
Fig. 1 Block diagram of the proposed QBH system

4 Entropy Estimation

The music signal data points often exhibit complex fluctuations containing information about the underlying regularity or consistency. Hence an important problem in music processing is determining whether the music signal has a regularity component. In this section, we discuss the entropy estimation procedure which can be quite effectively used to quantify the amount of regularity.

4.1 Shannon Entropy

Shannon's information theory defined a quantitative measure of information associated with an event A, which is a set of outcomes of some experiment. If $P(A)$ is the probability of an event A, then information associated with A is given by

$$i(A) = \log_b \frac{1}{P(A)} = -\log_b P(A) \qquad (1)$$

So, the amount of information associated with low probability event is more and vice versa. If we have a set of independent events A_i, which are sets of outcomes of experiment S, then the average information associated with the experiment is given by

$$H(S) = \sum_{i=1}^{N} P(A_i) i(A_i) = -\sum_{i=1}^{N} P(A_i) \log_b P(A_i) \qquad (2)$$

Here, the quantity $H(S)$ is an information entropy or Shannon entropy of S. So entropy is defined as measure of mean information content or information

consistency or information regularity [21]. The choice of the logarithm base in Eq. (2) determines the unit of information. In this work, we have used the bits units obtained by the use of logarithm base 2.

4.2 Multiscale Entropy

Conventional single scale entropy measures, purely quantify the regularity of data samples. Significant structural richness or complexity which assimilates correlations over multiple spatiotemporal scales can be derived through multiscale analysis [22]. So, ME is adopted as a new method for measuring the complexity of data samples.

The Fig. 2a describes the pictorial representation of coarse graining baseline procedure for scale 1, 2, and 3. For a given data set, $X = \{x_1, x_2, x_3, \ldots, x_N\}$, multiple coarse grained sequences are constructed by averaging the data points within nonoverlapping windows of increasing length. Each element of the coarse grained sequences, y_j^δ, is generated according to the equation:

$$y_j^\delta = \frac{1}{\delta} \sum_{i=(j-1)\delta+1}^{j\delta} x_i \tag{3}$$

where δ represents scale level 1–l and $1 <= j < = N/\delta$. The length of the coarse grained sequence is $M = N/\delta$. For scale level $\delta = 1$, the coarse grained sequence is same as the original sequence x_i where $i = 1$–N.

We then render the entropy measurement e_δ of each coarse grained sequence y_j^δ of length M for scale level $\delta = 1$–l as defined in the following equation:
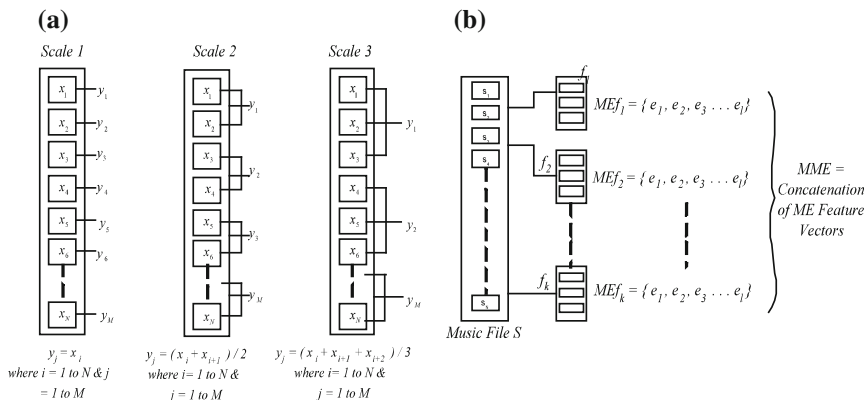

Fig. 2 **a** Schematic illustration of the coarse graining procedure for *scale 1, 2,* and *3.* **b** MME estimation procedure

$$e_\delta = -\sum_{j=1}^{M} P\left(y_j^\delta\right) \log_b P\left(y_j^\delta\right) \tag{4}$$

where $P\left(y_j^\delta\right)$ indicates probability of data point y_j at scale level δ and b is logarithm base. In this work, we have used the bits units obtained by the use of logarithm base 10. Now, ME is defined as the feature vector of entropies e_δ for scale levels $\delta = 1\text{--}l$ as shown in Eq. (5):

$$\text{ME}(X, \delta) = \overset{t}{\underset{\delta=1}{II}} e_\delta \tag{5}$$

where X is an original data set.

4.3 Multiscale Music Entropy

In this section, we reproduce the analysis of ME with reference to music signals. Given a music signal database

$$D = \{S_1, S_2, \ldots S_R\} \tag{6}$$

and a query Q, the aim is to find all the music signals in D that contain the specified query Q as well as feature vector similar to that of Q. One dimensional music signal S is defined as a sequence of values:

$$S = [s_1, s_2, \ldots s_N] \tag{7}$$

where N is the number of samples in S. Primary objective is to represent music melody in the form of ME.

Each music signal S is split into several frames, as defined in Eq. (8), so that it is possible to analyze each frame in short time.

$$S = [f_1, f_2, \ldots f_k] \tag{8}$$

where $f_1 = [s_1, s_2, \ldots s_{250}]$, $f_2 = [s_{251}, s_{252}, \ldots s_{500}]$, $\ldots f_k$. So, for every frame f, $\text{MEf}_{i,\ \text{feature}}$ vector of entropies e_δ for scale levels $\delta = 1\text{--}l$ are constructed by following the procedure discussed in Sect. 4.2. Same is defined in Eq. (9):

$$\text{MEf}_i = \overset{t}{\underset{\delta=1}{II}} e_\delta \tag{9}$$

where $i = 1\text{--}k$.

Finally feature vectors, MEf, of frames, $f_1 - f_k$, are concatenated to form the feature vector of one music signal referred as MME, expressed through equation as:

$$\text{MME} = \overset{t}{\underset{i=1}{II}} \text{MEf}_i \qquad (10)$$

So, this process is repeated for all music signals and knowledge base of MME is built during training phase. MME construction system for database D is depicted in Fig. 2b and steps are shown in Algorithm 1.

Algorithm 1 : Procedure to construct Multiscale Music Entropy (MME) for music database

---

Input: a music database D, number of music signals R, number of entropy scale levels $\delta = 1$ to l and number of samples in each music signal N

Output: Multiscale Music Entropy MME_D data set

/* **Framing** */
$for\ i = 1 : R\ do$
split music signal S_i of database D into several frames
$S_i = [f_1, f_2, ... f_k]$ /* k= number of frames */
end

/* **Generating coarse grained sequence & estimating entropy for each sequence** */
$for\ z = 1 : R\ do$
$for\ t = 1 : k\ do$
$for\ \delta = 1 : l\ do$
$for\ j = 1 : N/\delta\ do$
multiple coarse grained sequences are generated for frame, f_t, of each music signal
$y_j^\delta = \frac{1}{\delta} \sum_{i=(j-1)\delta+1}^{j\delta} x_i$
/* x= original sequence in frame, f_t */
end
then render the entropy measurement of each coarse grained sequence
$e_\delta = - \sum_{j=1}^{M} P(y_j^\delta) \log_b P(y_j^\delta)$
where $P(y_j^\delta)$ indicates probability of data point y_j at scale level δ, $M = \frac{N}{\delta}$ and b is logarithm base
end

/* **Concatenation of all entropies of coarse grained sequence to form MEf for every frame** */
$MEf_t = \overset{l}{\underset{\delta=1}{\|}} e_\delta$
end

/* **Concatenation of all MEf of every frame to form MME for every music signal** */
$MME = \overset{k}{\underset{i=1}{\|}} MEf_i$
insert generated MME to the result data set MME_D
end
return the result data set MME_D

---

5 Kd Tree-Based Indexing

The Kd tree is a binary tree, which iteratively splits the entire input space into partitions. Each internal node represents splitting condition and leaves of this node denote subsets of the partition. Thus, the root node represents the entire input space, while the leaf nodes explicitly record the data points of smallest partitions [23]. The tree construction process becomes accustomed to the local characteristics of input space and the sizes of partitions at the same phase are not necessarily equal.

Our implementation employs the multiscale music entropy-balanced (MMEB) Kd tree for music database indexing. The tree indexing utilizes the MME statistical

properties of music data for efficient and accurate searches. The MMEB Kd tree is an extension of statistical Kd tree which optimizes the multidimensional decision tree based on balancing split decisions. This balancing policy is adopted during tree building process so that the average and worst case search depth is reduced.

To construct a tree from a MME knowledge base, top down recursive procedure is employed. The common variation of splitting a subset of partition is the median point μ of the dimension d. This method of splitting guarantees a balanced tree and leads to reduced search depth. The cost of building a tree from N data points is O (NlogN). The steps of building MMEB Kd tree are shown in Algorithm 2.

Algorithm 2 : Procedure to construct Multiscale Music Entropy Balanced (MMEB) Kd tree for indexing music database

---

Input: number of music signals R in music database and Multiscale Music Entropy feature vectors $\{ MME_i \} \epsilon$ Knowledge Base, where $i = 1$ to R

Output: A MMEB Kd Tree $\{K\}$. Every internal node has a split condition, $\{d, \mu\}$
   where d is the dimension and μ is the median to split on.
   Here all feature vectors, $\{ MME_i \} \epsilon$ Knowledge Base, where $i = 1$ to R,
   belong to the left or right child based on split condition, $\{ d, \mu \}$.
   The leaf nodes have a list of indices to the features that ended up in that node

/* Build Function */
1. assign all the feature vectors, $\{ MME_i \} \epsilon$ Knowledge Base, where $i = 1$ to R to the root node of K
2. every node in the tree is visited using *Breadth First Order*
3. *for* for all the feature vectors, $\{ MME_i \} \epsilon$ Knowledge Base, where $i = 1$ to R, dimension $d = 1, 2, 3, ...Z$ *do*
compute its *median* μ from the points in that node
choose a *dimension* d at random
choose the split condition $\{ d, \mu \}$
for for all points that belong to this node: *do*
if $\{ MME_i[d] \} \leq median$ μ
  assign $\{ MME_i[d] \}$ to *left [node]*
  else
  assign $\{ MME_i[d] \}$ to *right [node]*
end if
end
end
4. step 3 is performed recursively until the subdivision depth exceeds threshold

---

6 Query Processing and Searching

Query processing basically involves query preprocessing, framing, and entropy estimation as discussed in Sects. 3 and 4. The melody database is a set of MME feature vectors indexed by MMEB Kd tree. The user hums different proportions of the query melody which is segmented into frames of 25 ms. Subsequently, each frame is resolved into query multiscale music entropy (QMME) feature vector.

Query vector QMME is searched in MMEB Kd tree using K-Nearest Neighbor (K-NN) algorithm to find those music signals whose MME feature vectors are close to QMME. Tree is recursively searched from the root to leaves. In concurrence with neighborhood range, search is carried out to locate those branches, which are close to

the branches where the query resides. The cost of searching MMEB Kd tree with N data points is O (logN). The steps of K-NN searching are shown in Algorithm 3.

7 Results and Discussions

In order to evaluate the proposed method, several experiments are conducted with datasets and interesting trends are observed in the performance. The performance evaluation is done through standard yardsticks such as mean reciprocal rank (MRR), mean of average (MoA), and Top X Hit Rate.

The training database chosen for the experiments consists of 1,000 Indian devotional monophonic songs. These are further segmented to 1,350 fragments. Then a subset of 100, 200, 500, 800, 1,000, and 1,350 fragments are employed during training phase. The query database consists of total 200 hums from ten participants. The humming query is searched in the music database and QBH system's performance analysis is evaluated with different metrics such as MRR, MoA, and Top X Hit Rate [6, 7, 24].

Algorithm 3 : Procedure to search Multiscale Music Entropy Balanced (MMEB) Kd tree using K-NN

Input: MMEB Kd tree $\{K\}$, number of music signals R in music database, Multiscale Music Entropy feature vectors $\{MME_i\} \in Knowledge\ Base$, $where\ i = 1\ to\ R$ used to build tree, a query feature vector, $QMME$ and k to decide number of nearest neighbors

Output: A set of relevant music signals $\{A_i\}$ $where\ i = 1\ to\ k$ which are nearest neighbors with their distances $\{B_i\}$ $where\ i = 1\ to\ k$ to the query feature vector, $QMME$

/* K-NN Search Function */

1. given a humming query Q, extract its $QMME$ features
2. search the Kd Tree K to find the nearest neighbors $\{A_i\}$ $where\ i = 1\ to\ k$ with their distances $\{B_i\}$ $where\ i = 1\ to\ k$
3. initialize a priority queue, PQ, with the root nodes of the K tree by appending
   $branch = (node, val)\ with\ val = 0$
4. the priority queue PQ is indexed by $val\ [branch]$ and it returns the branch with smallest val
5. initialize $counter = 0, vector = [\ ]$
6. *for counter* $<=$ *max no of backtracking steps do*
   retrieve the top branch from PQ
   descend the tree defined by branch till leaf by adding unexplored branches on the way to PQ
for $i = 1$ to length ($QMME$ feature vector) do
   if $QMME(i) == vector(i)$ then /*Matching condition*/
      $result \leftarrow result \cup QMME(i)$
   end if
   if $vector(i) == *$ then /*Partial query matching */
      perform match with $K.left, vector\ and\ result$
      perform match with $K.right, vector\ and\ result$
   else
      if $vector(i) < QMME(i)$ then
      perform match with $K.left, vector\ and\ result$
      else
      perform match with $K.right, vector\ and\ result$
      end if
   end if
end
end
7. locate the k nearest neighbors to PQ in vector and return the sorted vector
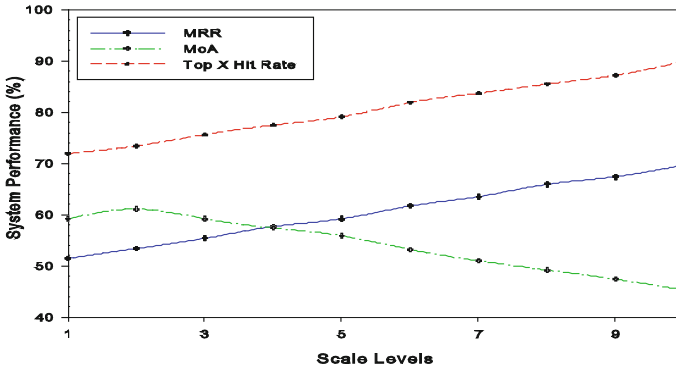 $\{A_i\}$ $where\ i = 1\ to\ k$ and their distances $\{B_i\}$ $where\ i = 1\ to\ k$

Fig. 3 QBH system's performance analysis with *MRR*, *MoA,* and *Top X Hit Rate*

7.1 Performance Analysis

MRR metric estimates any system which generates list of possible responses to a query. The probability of the target song coming in first position is estimated through MRR [20]. In the proposed strategy, MRR is obtained in the range 51.50–69.72 % for different scale levels. Also the performance of the system in terms of MRR goes on increasing with an increase in the scale level as depicted in Fig. 3. This is an indication that system's potential ability of discriminating songs increases with higher scale levels of MME estimation.

Similarly mean of accuracy (MoA) metric estimates the mean or average rank at which desired song is found for every query. The proposed strategy yields MoA in the range 59.23–45.32 % with scale levels one to ten. Experimental results demonstrate that MoA decreases with higher scale levels of MME estimation as depicted in Fig. 3. This is an indication that mean or average rank of the retrieved song decreases with increase in scale levels of MME estimation.

The third performance measure the Top X Hit Rate is employed to calculate approximately the percentage of successful queries. The different scale level of MME estimation procedure has significant effect on Top X Hit Rate and same is portrayed in Fig. 3. The Top X Hit Rate varies between 71.91 to 89.90 % for X value 10 and 1–10 scale levels of MME estimation. More humming queries return the desired song in the top 10 list when the scale level of MME estimation is increased from 1 to 10. The proposed QBH system based on MME exhibits significant performance in terms of accuracy, MRR and MoA metrics.

8 Conclusions

This paper is an attempt to exploit the advantages of Multiscale Entropy Analysis technique for estimating regularity and complex structure of music signals. Here, MME feature vectors are employed to represent music melody for music search

through QBH system. Also, music melody is indexed through MMEB Kd tree to support fast and accurate search in the database to produce effective music retrievals. Proposed approach yields sensible performance by returning the desired song within the top 10 hits 89.90 % of the time and as the top hit 69.72 % of the time on a database with 1,000 songs and 1,350 fragments. Exhaustive exploration of the possibility of combining Multiscale Entropy Analysis techniques is to be considered in future work.

References

1. Addis, A., Armano, G., Vargiu, E.: Using the progressive filtering approach to deal with input imbalance in large-scale taxonomies. In: Proceedings of LSHC Workshop of ECIR (2010)
2. Jang, J.S.R., Lee, H.R.: Hierarchical filtering method for content based music retrieval via acoustic input. In: Proceedings of the 9th ACM Multimedia Conference, Canada, pp. 401–410 (2001)
3. Adams, N.H., Bartsch, M.A., Shifrin, J.B., Wakefileld, G.H.: Time series alignment for music information retrieval. In: Proceedings of 5th ISMIR, pp. 303–311 (2004)
4. Selina, C., Eamonn, K., David, H., Michael, P.: Iterative deepening dynamic time warping for time series. In: Proceedings of 2nd SIAM International Conference on Data Mining (2002)
5. Zhu, Y., Shasha, D.: Warping indexes with envelope transforms for QBH. In: Proceedings of the ACMSIGMOD International Conference on Management of Data, California, pp. 181–192 (2003)
6. Adams, N., Marquez, D., Wakefield, G.: Iterative deepening for melody alignment and retrieval. In: Proceedings of ISMIR, pp. 199–206 (2005)
7. Jang, J.S.R., Lee, H.R.: An initial study on progressive filtering based on DP for QBSH. In: Proceedings of 7th IEEE Pacific-Rim Conference on Adv. in MIP, China, pp. 971–978 (2006)
8. Raju, M.A., Sundaram, B., Preeti Rao:. Tansen: a query-by-humming based music retrieval system. In: Proceedings of the National Conference on Communications (NCC) (2003)
9. Shifrin, J., Pardo, B., Meek, C., Birmingham, W.: HMM based musical query retrieval. In: Proceedings of 2nd ACM/IEEE-CS Joint Conference on DL, Oregon, USA, pp. 295–300 (2002)
10. Jeon, W., Ma, C.: Efficient search of music pitch contours using wavelet transforms and segmented DTW. In: Proceedings of IEEE Internationl Conference on ICASSP, Prague, pp. 2304–2307 (2011)
11. Thuraisingham, R.A., Gottwald, G.A.: On multiscale entropy analysis for physiological data. Technical report (2006)
12. Davies, M.E.P., Plumbley, M.D.: On the use of entropy for beat tracking evaluation. In: Proceedings of IEEE International Conference on ASSP, Honolulu, HI, pp. 1305–1308 (2007)
13. Yan, R.Y., Zheng, Q.H.: Multi-scale entropy based traffic analysis and anomaly detection. In: Proceedings of 8th International Conference on ISDA, Kaohsiung, Taiwan, pp. 151–157 (2008)
14. Costa, M., Peng, C.K., Goldberger, A.L., Hausdorff, J.M.: Multiscale entropy analysis of human gait dynamics. Phys. A Stat. Mech. Appl. Technical report (2003)
15. Riihijarvi, J., Wellens, M., Mahonen, P.: Measuring complexity and predictability in networks with MEA. In: Proceedings of IEEE INFOCOM, Rio de Janeiro, pp. 1107–1115 (2009)
16. He, H., Chen, B., Guo, J.: Emotion recognition of pop music based on maximum entropy with priors. In: Proceedings of 13th Pacific-Asia Conference, PAKDD, Thailand, pp. 788–795 (2009)

17. Simon, S.J.: Measuring Information in Jazz Improvisation. Technical report School of Library and Information Science, University of South Florida, South Florida (2007)
18. Ibarrola, A.C., Chavez, E.: On musical performances identification, entropy and string matching. In: Proceedings of MICAI, Springer, Advances in AI LNCS, pp. 952–962 (2006)
19. Cox, G.: On the relationship between entropy and meaning in music: an exploration with recurrent neural networks. In: Proceedings of the Annual Meeting of the Cognitive Science Society (2010)
20. Trisiladevi, C.N., Nagappa, U.B.: Perceptive analysis of QBS system through query excerption. In: Proceedings of the 2nd International Conference on CCSEIT, ACM, India, pp. 580–586 (2012)
21. Sayood, K.: Introduction to Data Compression, 3rd edn. Elsevier (2006)
22. Costa, M., Goldberger, A.L., Peng, C.K.: Multiscale entropy analysis of biological signals. Technical report (2005)
23. Aly, M., Munich, M., Perona, P.: Distributed Kd-trees for retrieval from very large image collections. In: Proceedings of BMVC, Dundee, UK (2011)
24. Jang, J.S.R., Lee, H.R.: A general framework of progressive filtering and its application to query by singing/humming **16**:350–358 (2008)