Ratanpara and Patel *EURASIP Journal on Audio, Speech, and Music Processing* (2015) 2015:16
DOI 10.1186/s13636-015-0062-9

EURASIP Journal on Audio, Speech, and Music Processing
a SpringerOpen Journal

RESEARCH Open Access

Singer identification using perceptual features and cepstral coefficients of an audio signal from Indian video songs

Tushar Ratanpara[1*] and Narendra Patel[2]

Abstract

Singer identification is a difficult topic in music information retrieval because background instrumental music is included with singing voice which reduces performance of a system. One of the main disadvantages of the existing system is vocals and instrumental are separated manually and only vocals are used to build training model. The research presented in this paper automatically recognize a singer without separating instrumental and singing sounds using audio features like timbre coefficients, pitch class, mel frequency cepstral coefficients (MFCC), linear predictive coding (LPC) coefficients, and loudness of an audio signal from Indian video songs (IVS). Initially, various IVS of distinct playback singers (PS) are collected. After that, 53 audio features (12 dimensional timbre audio feature vectors, 12 pitch classes, 13 MFCC coefficients, 13 LPC coefficients, and 3 loudness feature vector of an audio signal) are extracted from each segment. Dimension of extracted audio features is reduced using principal component analysis (PCA) method. Playback singer model (PSM) is trained using multiclass classification algorithms like back propagation, AdaBoost.M2, *k*-nearest neighbor (KNN) algorithm, naïve Bayes classifier (NBC), and Gaussian mixture model (GMM). The proposed approach is tested on various combinations of dataset and different combinations of audio feature vectors with various Indian male and female PS's songs.

Keywords: AdaBoost.M2; Singer; Music; Timbre; Pitch; Signal; Audio; MFCC; LPC; Loudness

1 Introduction

Indian music has become popular because of their playback singers. An Indian Hindi movie contains video songs which are sung by distinct playback singers. Viewer can upload/download video songs from internet and CD/DVDs. Indian video songs (IVS) can be extracted from Indian movies [1, 2] which increases collections rapidly. Indexing of such IVS requires information in a different dimension like playback singer of IVS and on-screen actor/actress performing on IVS. Currently, this information is attached manually as a textual caption with IVS. Textual caption is highly unreliable because IVS is uploaded by ordinary user. So viewer requires powerful functions for browsing [2, 3], searching [4], and indexing video content. Singing voice is one of the most important parameters in Indian video songs. The singing voice of a singer is the element of a song which attracts the listeners. Singing is a continuous speech. Therefore speech and synthesis analysis techniques are not the same for singing voice. There is no efficient algorithm which works fine on speech identification and singing voice characterization together. So information on the singer's voice is essential to organize, extract, and classify music collections [5]. Sometimes, a viewer is interested to hear Indian video songs based on their interest like favorite playback singer, actor, and actress. So there is a requirement to develop a system which provides the above features.

The proposed system can identify singing voice and recognize a singer from IVS. Significant accuracy can be achieved by extracting features of audio part from IVS. One of the usefulness of this system is famous Indian playback singer's video songs can be identified from a big database. It can be useful to learn singer voice characteristics by listening to songs of different genres. It can be useful in categorizing unlabeled IVS and copyright protection. IVS require information in different dimensions for efficient searching and indexing. So this system

* Correspondence: tushar.ratanpara@gmail.com
[1]C. U. Shah University, Wadhwan, Gujarat, India
Full list of author information is available at the end of the article

can be useful to index and efficient search for query-based IVS retrieval. Here Indian video songs are considered rather than Indian audio songs because this system can be extended to for visual clues also. Indian video songs are marketed by its music, actor, and actress. So it is necessary to index Indian video song using parameters like playback singers, actor, and actress for efficient search and retrieval.

2 Related work

A significant amount of research has been performed on speaker identification from digitized speech for applications such as verification of identity. These systems use features which are used in speech recognition and speaker recognition. Systems are trained on data without background noise and performance [6] tend to degrade in noisy environments. They are trained on spoken data in which it produces poor result for the singing voice input. Mel frequency cepstral coefficients (MFCCs) [7] are originally developed for automatic speech recognition applications and can be used for music modeling. Pitch and rhythm audio features are computed. MFCC feature vectors and artificial neural network classifier are used to identify playback singer [8] from a database. An accuracy of 70 % is achieved by this system using 10-artist database. Instrumental and singing sounds were not separated in the system. Singer's vibrato-based octave frequency cepstral coefficients (OFCC) [9] are used in singer identification. Experiments were performed using 84 popular songs only from 12-singer database. An average error rate of 16.2 % is achieved in segment level identification. In [10], composite transfer function-based features are extracted and polynomial classifier is used for classification. Self-recorded database for 12 female singers are used to build training model which produces 82 % accuracy. Music features are extracted for a musicological purpose using Echo Nest API [11]. In [12], spectrogram is an effective parameter in time-frequency feature which is used as input classification. Several classification techniques are compared such as feed forward network [13] and *k*-nearest neighbor. Energy function, zero crossing rate, and harmonic coefficients [14] are used for singer identification. One of the drawbacks of the above system is training model is generated manually. Singer voice is separated manually by removing instrumental music from audio songs and it is used to build training model. Sometimes self-recorded dataset is considered for singer identification. It increases complexity of a system and requires lots of execution time. In our proposed approach, training model is built automatically, not manually. Vocals and instrumental music are not separated out manually. Both are used to build training model. In other systems, only audio songs of singers are considered. But here, video songs are taken as input. The main advantage of this system is extension using visual clues. Actor and actress can be classified from video portion. It can merge with our proposed system because sometimes users are interested to watch IVS of their favorite actor or actress on screen and listen to their favorite singer in the background.

The rest of the paper is organized as follows: Section 3 describes proposed approach. Experimental setup is given in Section 4. Experimental results are explained in Section 5 followed by conclusion in Section 6.

3 Proposed approach

The abstract model of the proposed system for playback singer recognition using perceptual features and cepstral coefficients of an audio signal from Indian video songs is shown in Fig. 1. It comprises of six building blocks: (1) song collection, (2) segmentation, (3) feature extraction, (4) dimension reduction, (5) model generation, and (6) singer recognition.

Algorithm 1 represents singer recognition approach using different classifiers from IS.

Algorithm 1 Singer recognition approach

1. Collection of N Indian video songs of M singers.
2. Separate audio portion and video portion from each video song.
3. Compute x1, x2, x3 ……x53 audio feature vectors for each segment of audio portion. Where x1-x12 timbre audio feature vectors, x13-x24 pitch class, x25-x27 loudness, x28- x40 MFCC feature vectors, and x41-x53 LPC coefficients. These features are stored in S1 structure. Size of S1 structure is S x 53, where S is total number of segments. Total number of segments depend on length of audio portion and audio feature which are explained Section 3.2.
4. Mean removal technique is applied on S1 structure and result is stored in S2 structure.
5. Principal component analysis method is used on S2 structure to compute eigenvalue, eigenvector using single value decomposition (SVD) technique. Score is obtained. Result is stored in score structure which is divided into two parts (training dataset (80 %) and testing dataset (20 %)).

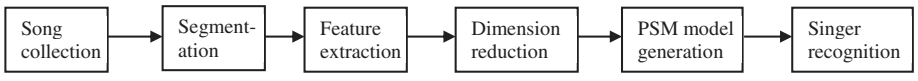

Fig. 1 Abstract model of our proposed system

6. PSM is obtained using back propagation and AdaBoost.M2 algorithm using neural network. PSM is also obtained using KNN, GMM, and naïve Bayes classifier.
7. Compute probability of each song for M singers from test sample song dataset.
8. Return a recognize singer name which contains maximum probability among M singers.

3.1 Song collection

In the proposed approach, dataset is generated by own because there is no standard database that is available for IVS of different singers. Six famous singers are selected from Indian Bollywood industry. For each singer, video songs are downloaded from the Internet. Then audio and video portion is separated out from IVS. Audio portion is divided into various segments to compute various audio features which is explained in Section 3.2.

3.2 Segmentation

Each collected video song is divided into segments. Audio channels are extracted from IVS which leads to segmentation. Fig. 2 represents segmentation process from IVS. Echo Nest timbre, Echo Nest pitches (ENP), and Echo Nest loudness (ENL) are computed using Echo Nest API [15]. Segments are characterized by their perceptual onsets of an Indian song in Echo Nest API automatically. It is observed that frame size is around 220–260 ms for an Indian song. Frame size of MFCC feature vectors and LPC coefficients is as follows. Audio portion is divided into different segments using Eq. 1.

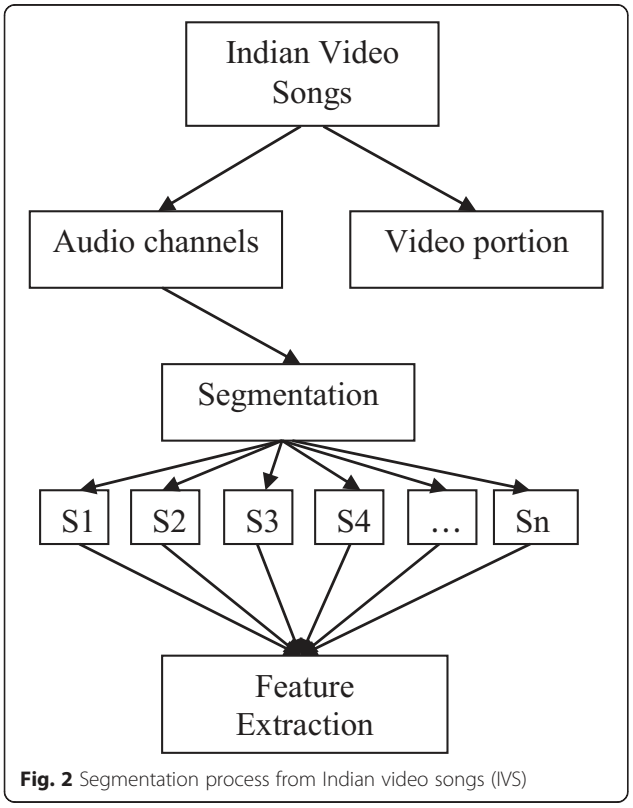

Fig. 2 Segmentation process from Indian video songs (IVS)

$$\mathrm{TNF} = \mathrm{TS\ (seconds)} / 5 \tag{1}$$

where TNF is total number of segments and TS total number of seconds of audio portion. Normally in other manuscripts, training model is generally built for a whole song. But in our proposed system, only the first 180 s are used to recognize a singer from IVS because it is assumed that vocals of singer are sufficient to recognize a singer. So the value of TS is 180 s and TNS is 36 segments. Frame size is 5 s (5000 ms) for each segment in an IVS. Feature extraction is done for each segment which is explained in Section 3.3.

3.3 Feature extraction

Music researchers have started a company named the Echo Nest [15] in 2005. It gives free analysis of music via API. Users can retrieve information regarding artists, blogs [16], and songs. A song has been uploaded by users which lead to unique song id to extract song features like tempo, timbre, loudness, and pitches. It can also collect socially oriented information like blogs, social networks, and web pages. Echo Nest version 4.2 is used in the procedure. Fifty-three audio feature vectors (×1, ×2, ×3 …×53) are computed from each segment. The following audio features are extracted using Echo Nest API: (1) Echo Nest timbre (ENT), (2) Echo Nest pitches, and (3) echo nest loudness.

3.3.1 Echo Nest timbre

When distinct singers are playing or singing with the same pitch at that time, we may not identify the difference between them using pitch. This difference in the quality of the pitch is defined as timbre [16]. Timbre is also referred as tone color or tone quality. It depends on the original shape of the wave form. The Echo Nest analyzer's timbre [11] feature is a vector that includes 12 unbounded values roughly centered on 0. Those values are high-level abstractions of the spectral surface ordered by degree of importance. It distinguishes different types of musical instruments or voices. The first dimension represents the average loudness of the segment, the second emphasizes brightness, the third is more closely correlated to the flatness of a sound, the fourth to sounds with a stronger attack, etc. Fig. 3 represents 12 basis functions (i.e., template segments).

The actual timbre of the segment is best described as a linear combination of these 12 basis functions weighted by the coefficient values:

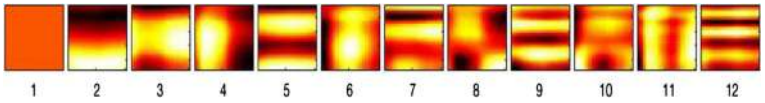

Fig. 3 Twelve basis functions [16] for the timbre vector

$$\text{timbre} = c1 \times b1 + c2 \times b2 + \ldots + c12 \times b12 \quad (2)$$

where $c1$ to $c12$ represent the 12 coefficients and $b1$ to $b12$ the 12 basis functions as displayed in Fig. 3. Twelve timbre audio features (×1 to ×12) are extracted using the 12 basis functions of timbre vector (Fig. 3) for each segment of an IVS.

3.3.2 Echo Nest pitches

Pitch is an auditory sensation in which a listener assigns musical tones to relative positions. It is measured in hertz. Pitch [16, 17] content is given by a "chroma" vector, corresponding to the 12 pitch classes C, C#, D to B, with values ranging from 0 to 1 that describe the relative dominance of every pitch in the chromatic scale [18]. Twelve audio features (×13 to ×24) are computed for each segment by 12 pitch classes.

3.3.3 Echo Nest loudness

Loudness [16] is defined as quality of a sound which is a primary psychological correlate of physical strength (amplitude) which is measured in decibels (DB) [11]. The following three audio features (×25, ×26, and ×27) of loudness are extracted for each segment in the proposed approach: (a) loudness start (b) loudness max time, and (c) loudness max.

- Loudness start: provides loudness level at the start of the segment
- Loudness max time: maximum loudness value within the segment
- Loudness max: highest loudness value within the segment

Mel frequency cepstral coefficients and linear predictive coding coefficients are computed using the following methods.

3.3.4 Mel frequency cepstral coefficients MFCC are the most useful coefficients which are used for speech recognition because of their ability to represent speech amplitude spectrum in a compact form. Figure 3 shows the process of creating MFCC features [7, 19]. Speech signal is divided into frames by applying a hamming windowing function at fixed intervals. Cepstral feature vectors are generated using each frame (Fig. 4).

The final step is to compute the discrete cosine transform (DCT) of the log filter bank energies in MFCC. But only 12 of the 26 DCT coefficients are kept because the higher DCT coefficients represent fast changes in the filter bank energies which reduce the performance of system. So it gives small improvement by dropping them. Thirteen MFCC coefficients (×28 to ×40) are used by our proposed approach which are extracted for each segment from IVS.

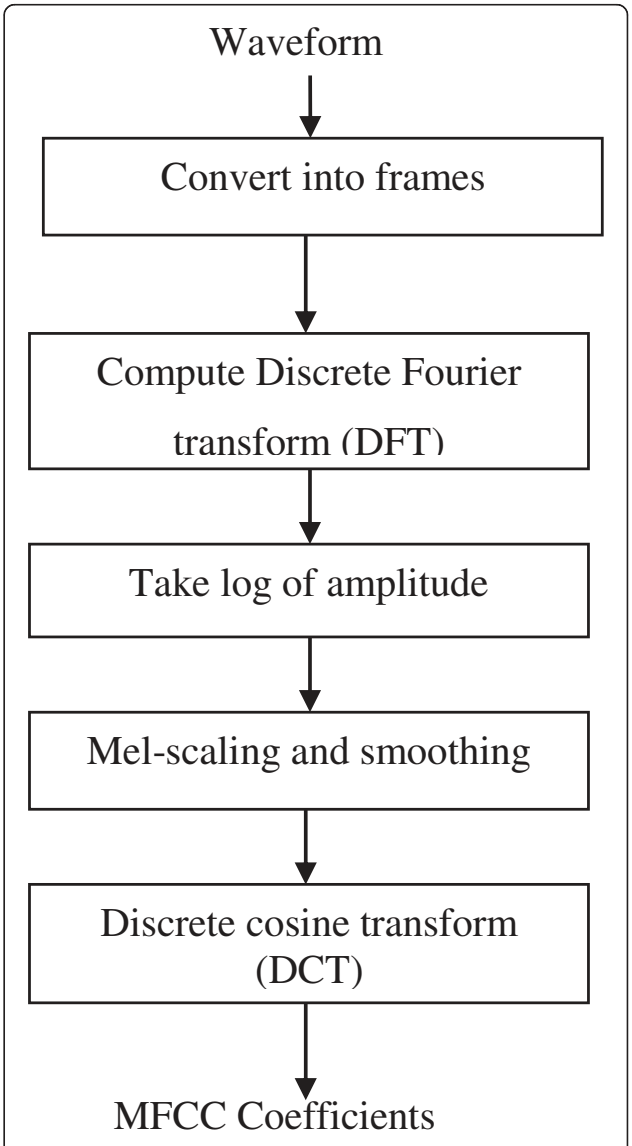

Fig. 4 The process of creating MFCC [7] coefficients

3.3.5 Linear predictive coding coefficients Linear predictive coding (LPC) [20] can provide a very accurate spectral representation for speech sound. LPC coefficients are computed by using following equation.

$$\mathrm{X(n)} = -a(2)X(n-1) - a(3)\mathrm{X}(n\text{-}2) \ldots\ldots -a(P+1)X(n-P) \tag{3}$$

where p is the order of the polynomial, a = [1 a (2) ... A (P + 1)]. In the proposed approach, 13 LPC coefficients (×41 to ×53) are calculated. The value of P is 12 (12th-order polynomial).

Indian video song length is around 5–7 min. Each segment calculates 53 audio features. A typical problem in this system is a large number of audio features are extracted from each Indian video song. It requires large computational time for training database which takes more execution time for song queries. Therefore, there is a need to reduce number of feature vector without losing important information of audio features. Dimension is reduced using principal component analysis (PCA) method which is discussed in the following section.

3.4 Dimension reduction

Principal component analysis method [21, 22] is used to compute principal component which reduces the dimension of extracted audio features. It retains as much as possible variance in the audio features. Principal components are extracted by a linear transformation to a new set of features which are uncorrelated and are ordered according to their importance. Principal component is computed using singular value decomposition algorithm.

PCA [23, 24] consists of five main steps which are explained in Algorithm 2.

Algorithm 2 Principal component analysis method

1. Subtract the mean from each audio feature vectors. It produces audio feature vectors whose mean is zero.
2. Calculate the covariance matrix.
3. Compute the eigenvalues and eigenvectors of the covariance matrix.
4. Sort eigenvalue in descending order. The number of eigenvectors represents the number of dimensions of the audio feature vectors.
5. Derive the new audio feature vectors. Take the transpose of the feature vector and multiply it on the left of the original data set.

$$\text{Final Feature Vectors} = \mathrm{RFE} * \mathrm{RDM} \tag{4}$$

where RFE is the matrix with the eigenvectors in the columns transposed and RDM is the matrix mean-adjusted data transposed. Final feature vectors which contain principal component score are divided into training and testing dataset. Playback singer models are generated by using training dataset which is discussed in Section 3.5.

3.5 Model generation

In the proposed approach, playback singer model (PSM) is obtained using a different classification [25, 26] algorithm. The following models are generated: (1) Gaussian mixture model, (2) k-nearest neighbor model, (3) naïve Bayes classifier (NBC), (4) back propagation algorithm using neural network (BPNN), and (5) AdaBoost.M2 model.

3.5.1 Gaussian mixture model

Playback singer recognition algorithm involves Gaussian mixture model [15, 27] distribution over an audio feature space like loudness, timbre, MFCC, LPC, and pitch. Mean and covariance of feature vector is computed for each song in our training dataset. Average unnormalized negative log-likelihood (average-UNLL) is calculated for a song given a Gaussian distribution which leads to prediction of singer.

For each audio feature vector, the UNLL is computed using Eq. 5.

$$\begin{aligned}\text{Unnormalized negative loglikehood (UNLL)} &= (X1\text{−mean\_singer}) * \text{inverse}(\text{cov\_singer}) \\ &\quad * \text{transpose}(X1\text{−mean\_singer})\end{aligned} \tag{5}$$

where $X1$ is the $1 \times F_n$ audio feature vectors, mean_singer is the $1 \times F_n$ mean vectors, and inverse of cov_singer is the $F_n \times F_n$ matrix which is computed by taking inverse of the covariance matrix. F_n is the total number of audio feature vectors which is used to generate Gaussian mixture model (GMM).

3.5.2 K-nearest neighbor model

K-nearest neighbor model [15] is used to predict singer using values of K (K = 3). The training samples are audio feature vectors which are distributed in a multidimensional feature space. Each training sample contains a class label. Feature vectors and class labels of training samples are stored in the training phase. Euclidean distance is computed for each test sample. Test samples are classified by assigning the class labels using k nearest training samples.

3.5.3 Naïve Bayes classifier

NBC [28] is highly scalable which requires linear number of parameters. Maximum likelihood training is computed by evaluating equation which takes linear time while other classifiers take expensive iterative approximation.

$$\text{Posterior probablility of } X \text{ being SCi} = \text{Prior probability of SCi} * \text{Likelihood of } X \text{ given SCi} \tag{6}$$

where X is a feature vector of a test sample. SCi represents the ith singer class. Prior probability of SCi and likelihood of X given SCi are computed using the following formulas.

$$\text{Prior probablility of SCi} = \frac{\text{number of sample songs of } SCi}{\text{total number of sample songs}} \tag{7}$$

$$\text{Likelihood of } X \text{ given SCi} = \frac{\text{number of sample songs of } SCi \text{ in the vicinity of } X}{\text{total number of sample songs of SCi}} \tag{8}$$

3.5.4 Back propagation algorithm using neural network

Back propagation neural network [29] model is a supervised learning model used in many applications.

It is based on gradient descent method. It calculates gradient of error function with respect to all weights in the neural networks which is used to update the weights in an attempt to minimize the error function. The following algorithm is used for a three-layer network (one input layer, one hidden layer, and one output layer). Number of neurons in the hidden layer is 50.

```
Initialize neural network synaptic weights ()
  Do
  For each training sample s
    Desired output = neural net output (neural network, s)
    Expected output = output(s)
    Compute error (desired output – actual output) at the output layer neurons
    Compute ΔWh for all weights from hidden layer to output layer
    Compute ΔWi for all weights from input layer to hidden layer
    Update neural network weights
Until all test samples classified correctly or another stopping criteria satisfied
Return the neural network
```

3.5.5 AdaBoost.M2 model

AdaBoost.M2 is a very popular algorithm for binary classification. AdaBoost.M2 [30, 31] is an extension of AdaBoost.M1 for multiple classes. It is generated using extracted audio features. Algorithm trains learners sequentially. For every learner with index t, it computes pseudo-loss [32]. It uses weighted pseudo-loss for N samples and K classes to compute classification error.

$$\varepsilon t = \frac{1}{2}\sum_{(n=1)}^{N} \Sigma_{(k \neq Yn)}(1 - ht(Xn, Yn) + ht(Xn, k)) \; * \; \mathrm{d_{n,k,t}} \tag{9}$$

where $h_t(X_n,k)$ is the confidence of desired output (prediction) by learner at step t into class k. Output range is between 0 (not at all confident) to 1 (highly confident). $d_{n,k,t}$ are sample weights at step t for class k. y_n is the true class label taking one of the k values. The second summation is done on all classes other than the true class y_n.

Classification accuracy can be measured using pseudo-loss from any learner in the network. Then it increases weights for samples which are misclassified by learner t and reduces weights for samples which are correctly classified by t. The next learner $t + 1$ is then trained on the data using updated weights $d_{n,(t+1)}$.

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x) \tag{10}$$

where

$$\alpha_t = \frac{1}{2}\frac{\log(1-\varepsilon t)}{\varepsilon t} \tag{11}$$

are weights of the weak hypotheses in the network.

3.6 Singer recognition

In our proposed approach, singer recognition is carried out using trained models. Indian video songs (IVS) which are not used in trained models are given for testing using various classification algorithms which leads to recognition of a singer.

4 Experimental setup

A novel music database is prepared for Indian Hindi video songs of six famous Indian playback singers from Bollywood movies and albums which are publically available in CDs/DVDs. Each singer contains 50 songs. Proposed technique is evaluated using 300 popular songs. Our dataset consists of video songs of singers whose contribution in Bollywood industry is over a period of 20 to 25 years. To maintain uniformity of our database, we have converted each song to 256 kbps bit rate. All the experiments have been performed in MATLAB, on a standard PC (Intel Core i3, 2.30 GHz, 4-GB RAM). Table 1 represents the list of singers whose songs are selected for our database.

Table 1 List of singers used in our database

ID	Singer name	Gender
S1	Rahat Fateh khan	Male
S2	Sunidhi Chauhan	Female
S3	Sonu nigam	Male
S4	Mohit Chauhan	Male
S5	Shreya Ghoshal	Female
S6	Atif Aslam	Male

Table 2 Various types of division for database

Phase	Dataset1	Dataset2	Dataset3
	Number of songs used		
Training	120	192	240
Testing	30	48	60

5 Experimental results

Normally playback singer identification system is divided into two parts: training phase and testing phase. PSM is generated using known songs of singers which are used in testing phase to test unknown song of singers and score is computed. As for the query set, 20 % dataset is used from each singer and the remaining 80 % dataset is used for training. So training samples are selected automatically, not manually, from the dataset.

To compute accuracy of our system, we have used 5-fold random cross validation method [33] in each playback singer model. Accuracy of the proposed system is computed by the following equation.

$$\text{Accuracy } (\%) = \text{SIC/TC} \tag{12}$$

Table 3 Different types of combinations of audio features

Audio features	TL	PL	TPL	MLP	MT	MLPT	MLPTPL
Timbre (T)	Yes	No	Yes	No	Yes	Yes	Yes
Loudness (L)	Yes	Yes	Yes	No	No	No	Yes
Pitches (P)	No	Yes	Yes	No	No	No	Yes
MFCC (M)	No	No	No	Yes	Yes	Yes	Yes
LPC (LP)	No	No	No	Yes	No	Yes	Yes

where SIC is number of songs in which singers are identified correctly and TC is total number of songs which are used for testing. To check diversity of database, it is divided into various parts as shown in Table 2.

Figure 5 shows performance of various divisions of datasets using AdaBoost.M2, *k*-nearest neighbor (KNN), GMM, BPNN, and NBC model. AdaBoost.M2 model performs better than KNN, GMM, BPNN, and NBC model. AdaBoost.M2 model is trained with 5000 number of learning cycles and an MLPTPL feature set (Table 3). Dataset2 and Dataset3 give less accuracy than Dataset1 because of over fitting [34]. Over fitting generally occurs because of complex model and too many parameters relative to number of samples.

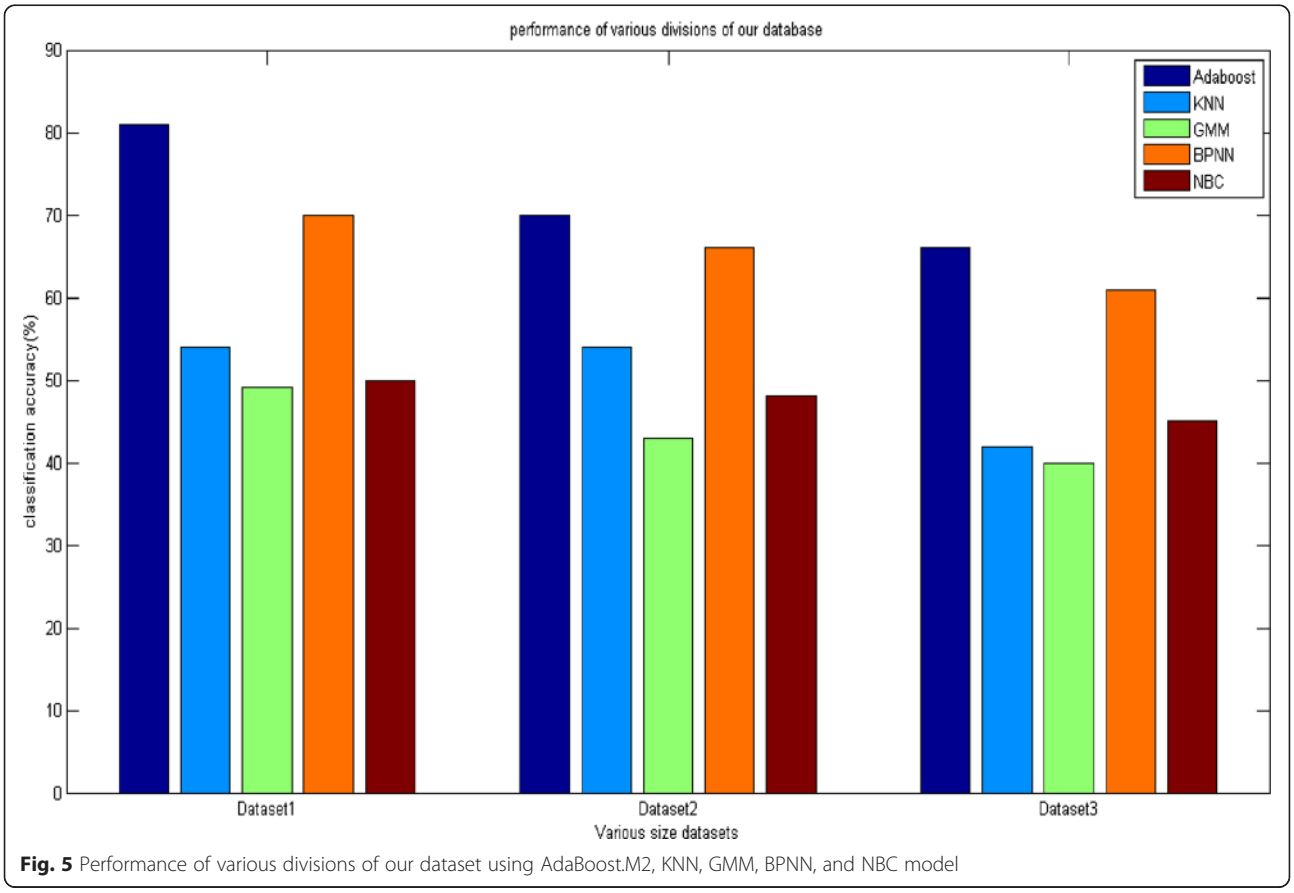

Fig. 5 Performance of various divisions of our dataset using AdaBoost.M2, KNN, GMM, BPNN, and NBC model

It is necessary to check the performance of AdaBoost.M2 model with different numbers of learning cycles which is shown in Fig. 6. It shows that accuracy is increased in AdaBoost.M2 model when the number of learning cycles increases. It produces 80 % or more accuracy using 3000 or more numbers of learning cycles.

Redistribution error is misclassification error of the class labels in training data. Redistribution error is computed after each learning cycle. Performance of redistribution error using different numbers of learning cycles is shown in Fig. 7. Redistribution error is more in 50 and 500 numbers of learning cycles compared to 1000 and 5000 numbers of learning cycles.

Confusion matrix is a specific table layout which gives visualization performance of the proposed approach. Figure 8 represents confusion matrix of AdaBoost.M2 model with MLPTPL feature set and 5000 numbers of learning cycles. On the x-axis target class is plotted and y-axis contains output class of various singers' songs.

Receiver operating characteristic (ROC) [35] is a graphical plot that gives the performance of a binary classifier system. The curve is created by plotting the true positive rate against the false positive rate as shown in Fig. 9. Youden's index [36] is often used in conjunction with ROC analysis. Youden's index is a single statistic that captures the performance of a diagnostic test. The index is used for summarizing the performance of a classifier. Its value ranges from 0 to 1. Zero value indicates that classifier gives the same proportion of positive results for groups with and without the features. So the test is useless. A value of 1 indicates that there are no false positives or false negatives which means the test is perfect. The index gives equal weight to false positive and false negative values, so all tests with the same value of the index give the same proportion of total misclassified results. Here sensitivity and specificity is 1. So the value of Youden's index is 1.

$$J = \text{Sensitivity} + \text{Specificity} - 1 \tag{13}$$

Performance of AdaBoost.M2 model is tested using various performance measures. So it is concluded that AdaBoost.M2 model performs better than the other models. The proposed approach is also tested with different types of combinations of audio feature set as shown in Table 3. Now PSM model is built using various combinations of feature set. Performance curve is plotted in Figs. 10 and 11.

Figure 10 represents that TPL features is better than the TL and PL features. An accuracy of 80 % is achieved by AdaBoost.M2 model. An accuracy of 71 % is achieved by BPNN model using TPL features. Figure 11 represents performance of MLP, MT, MLPT, and MLPTPL feature sets using different classification models.

It shows that 81 % accuracy is achieved by MLPTPL feature set using AdaBoost.M2 model. AdaBoost.M2 model performs better rather than other playback singer models using other combination of feature set (Table 3). It is also observed that when MFCC coefficients are combined with LPC and timbre coefficients in AdaBoost.M2 model, then 70 % or more accuracy is achieved.

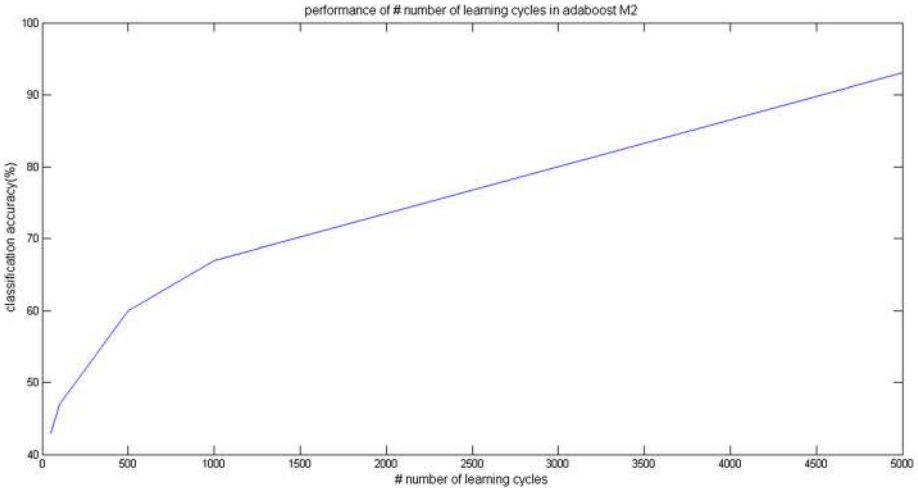

Fig. 6 Performance of AdaBoost.M2 model using different numbers of learning cycles

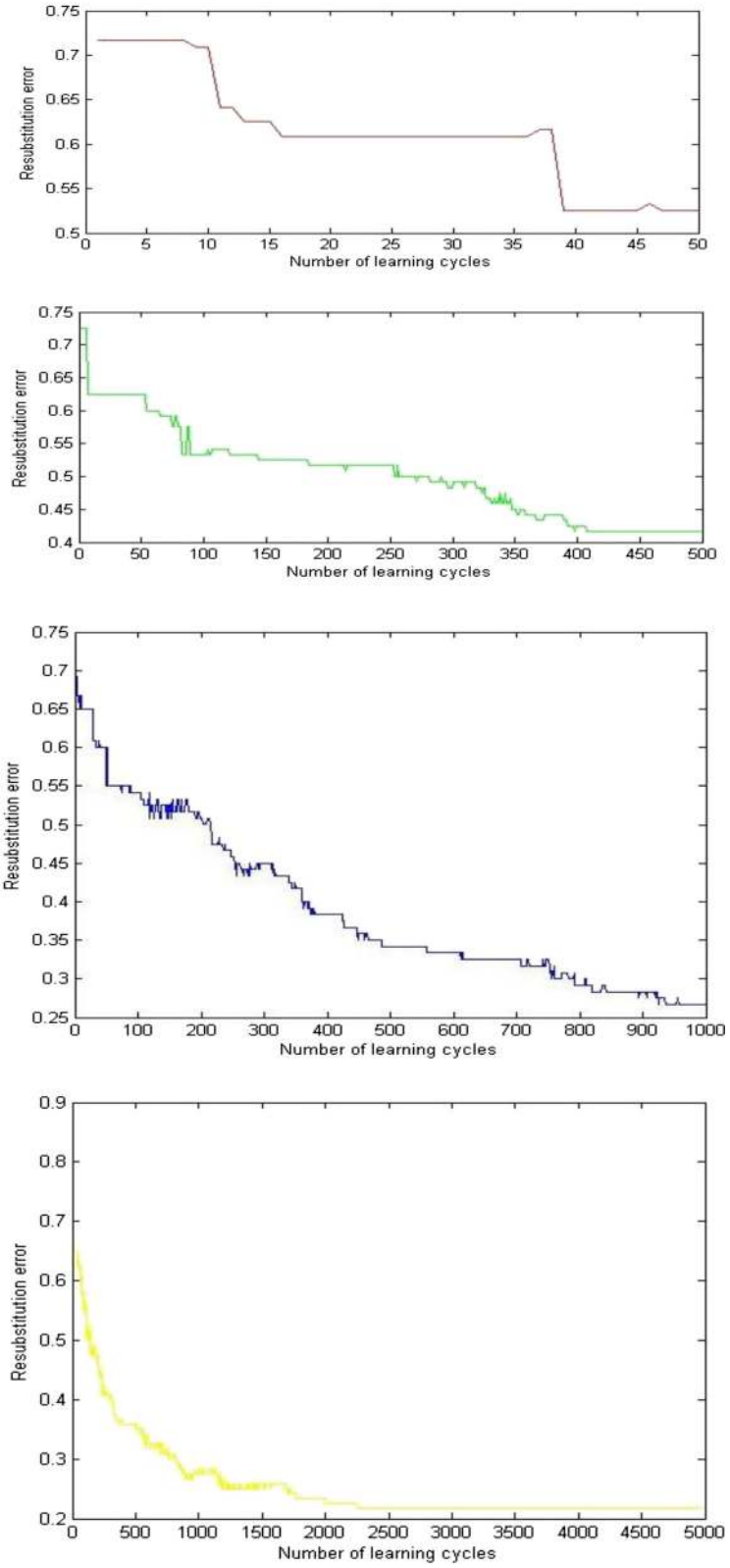

Fig. 7 50, 500, 1000, and 5000 numbers of learning cycles in AdaBoost.M2 model vs. redistribution error

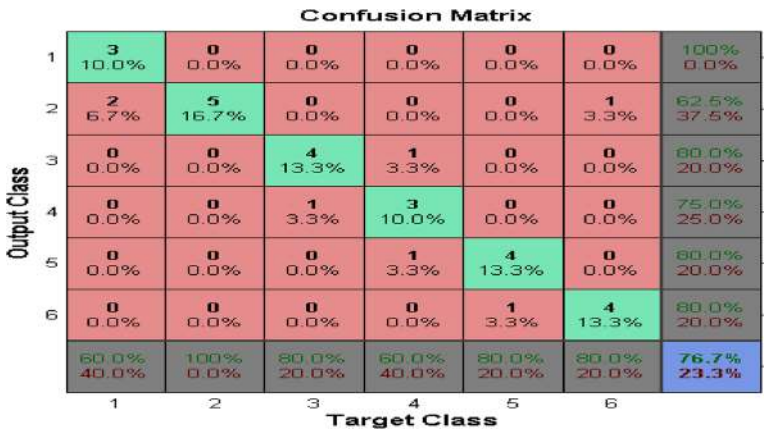

Fig. 8 Confusion matrix of AdaBoost.M2 model for a single pass

6 Conclusion

In this paper, playback singer recognition technique is proposed using perceptual features of an audio signal and cepstral coefficients from Indian Video songs. The proposed scheme first use PCA method to reduce dimensionality of audio feature vectors. Then five models (GMM, KNN, AdaBoost.M2, BPNN, and NBC) are generated using extracted audio feature vectors. An experimental result shows MLPTPL features with AdaBoost.M2 model gives more accuracy than other feature set. It is observed that AdaBoost.M2 is more efficient than GMM, BPNN, NBC, and KNN. Accuracy of AdaBoost.M2 model is increased when numbers of learning cycles are increased from 50 to 5000. Redistribution error is decreased when numbers of learning cycles are increased. The proposed system can be extended using visual clues from video portion to identify actor or actress. Various performance measures are plotted to show accuracy of our proposed approach.

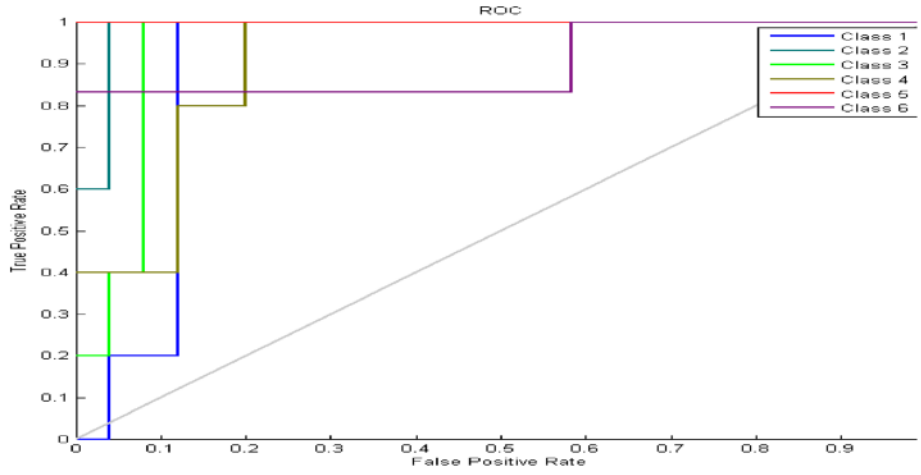

Fig. 9 ROC curve using AdaBoost.M2 model for a single pass

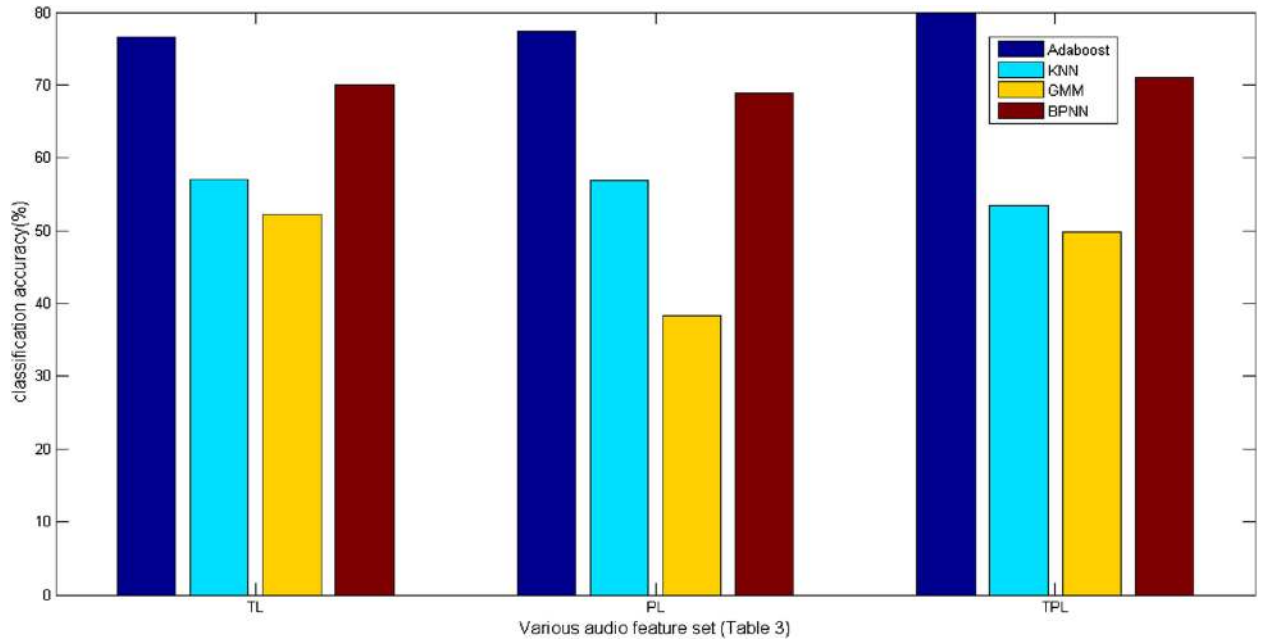

Fig. 10 Performance of different feature sets (TL, PL, and TPL; Table 3) using classification models

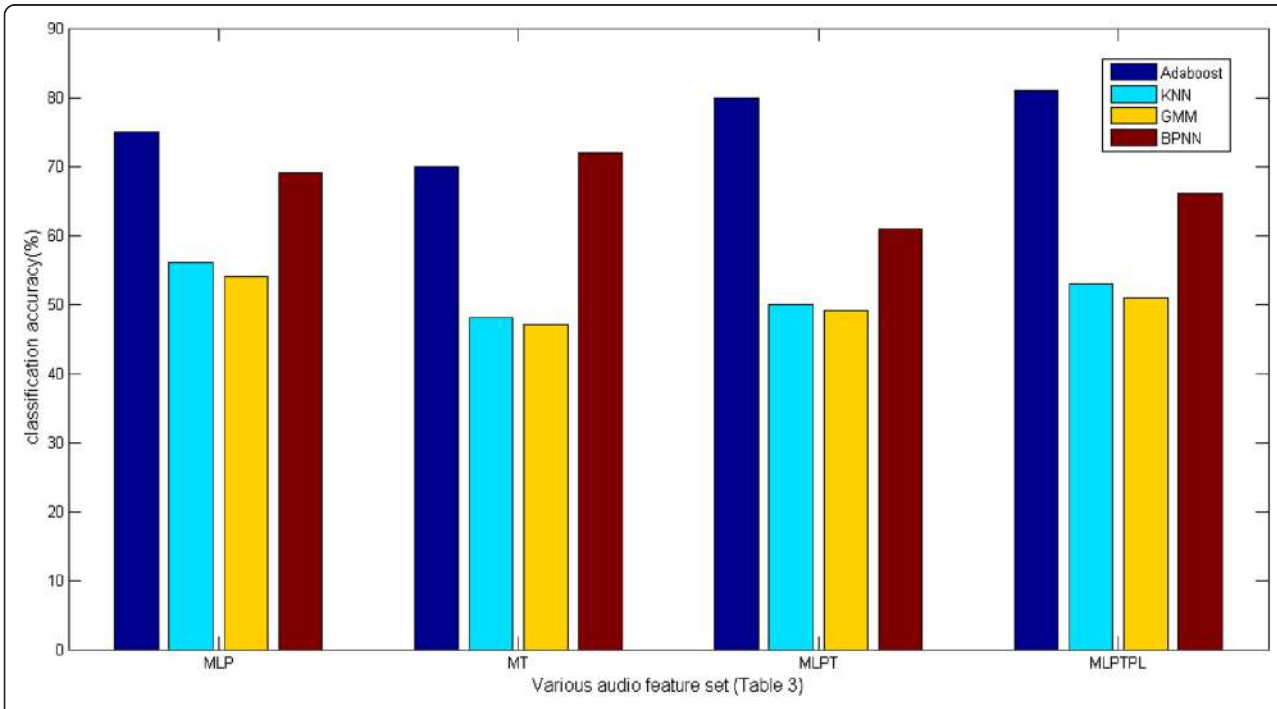

Fig. 11 Performance of different feature sets (MLP, MT, MLPT, and MLPTPL; Table 3) using classification models

Competing interests
The authors declare that they have no competing interests.

Author details
[1]C. U. Shah University, Wadhwan, Gujarat, India. [2]Birla Vishwakarma Vidhyalaya, v.v. nagar, India.

Received: 25 December 2014 Accepted: 17 June 2015
Published online: 25 June 2015

References

1. T Ratanpara, M Bhatt, A novel approach to retrieve video song using continuity of audio segments from Bollywood movies. Third International Conference on Computational Intelligence and Information Technology (CIIT), 87–92 (2013)
2. T Ratanpara, M Bhatt, P Panchal, A novel approach for video song detection using audio clues from Bollywood movies. Emerg. Res. Comput. Inf., Commun Appl **1**, 649–656 (2013)
3. Y Fukazawa, J Ota, Automatic task-based profile representation for content-based recommendation, IOS press. Int. J. Knowl. Based Intell. Eng. Syst. **16**, 247–260 (2012)
4. A Fanelli, L Caponetti, G Castellano, C Buscicchio, A hierarchical modular architecture for musical instrument classification, IOS press. Int. J. Knowl. Based Intell. Eng. Syst. **9**, 3 (2005). 173.182
5. L Regnier, G Peeters, Singer verification: singer model vs. song model, in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 437–440
6. R Mammone, J Rechard, X Zhang, R Ramachandran, Robust speaker recognition: a feature-based approach. IEEE Signal Process. Mag. **13**, 5 (1996)
7. B Logan, *Mel frequency cepstral coefficients for music modeling, in International Symposium Music Information Retrieval*, 2000
8. B Whitman, G Flake, S Lawrence, Artist detection in music with minnow match, in *Proceeding of the 2001 IEEE Workshop on Neural Networks for Signal Processing*, 2001, pp. 559–568
9. TL Nwe, H Li, Exploring vibrato-motivated acoustic features for singer identification. IEEE Trans. Audio Speech Lang. Process. **15**(2), 519–530 (2007)
10. M Bartsch, G Wakefield, Singing voice identification using spectral envelope estimation. IEEE Trans Speech Audio Process. **12**(2), 100–109 (2004)
11. J Andersen, Using the Echo Nest's automatically extracted music features for a musicological purpose. Cognitive Information Processing (CIP) 4th International Workshop, 1-6 (2014)
12. P Doungpaisan, Singer identification using time-frequency audio feature. Advances in Neural Networks – ISSN 2011 6676, 486–495 (2011)
13. L Gomez, H Sossa, R Barron, J Jimenez, A new methodology for music retrieval based on dynamic neural networks, IOS press. Int. J. Hybrid Intell. Syst. **9**, 1–11 (2012)
14. T Zhang, *System and method for automatic singer identification, IEEE International Conference on Multimedia and Expo*, 2003
15. Tingle, Derek, YE Kim, D Turnbull, Exploring automatic music annotation with "Acoustically-Objective" Tags. Proceedings of the international conference on Multimedia information retrieval, ACM, 55-62 (2010)
16. T Jehan, D DesRoches, *The Echo Nest Analyzer documentation*, 2014
17. B Marshall, Aggregating music recommendation Web APIs by artist, IEEE conference on Information Reuse and Integration (IRI), 75-79 (2010)
18. J Sun, H Li, L Ma, A music key detection method based on pitch class distribution theory, IOS press. Int. J. Knowl. Based Intell. Eng. Syst. **15**(3), 165–175 (2011)
19. T Ratanpara, N Patel, Singer identification using MFCC and LPC coefficients from Indian video songs, Emerging ICT for Bridging the Future Proceedings 49th Annual Convention Computer Society India (CSI) Volume 1 **337**, 275–282 (2015)
20. LB Jackson, *Digital Filters and Signal Processing*, 2nd edn. (Kluwer Academic Publishers, Boston, 1989), pp. 255–257
21. V Panagiotou, N Mitianoudis, *PCA summarization for audio song identification using Gaussian mixture models, in DSP*. 2013 18th international conference on digital singal processing (DSP), 1-6 (2013)
22. M. Zaki, J. Mohammed, W. Meira, Data mining and analysis: fundamentals of data mining algorithms, Cambridge University press, (2014)
23. M Diez, A Varona, on the projection of PLLRs for unbounded feature distributions in spoken language recognition, signal processing letters. IEEE **21**(9), 1073–1077 (2014)
24. S Shum, N Dehak, R Dehak, J Glass, *Unsupervised methods for speaker diarization: an integrated and iterative approach, IEEE Transactions on Audio, Speech, and Language Processing 21.10*, 2013, pp. 2015–2028
25. A Khan, A Majid, A Mirza, Combination and optimization of classifiers in gender classification using genetic programming, IOS press. Int. J. Knowl. Based Intell. Eng. Syst. **9**, 1–11 (2005)
26. A Lampropoulos, P Lampropoulou, G Tsihrintzis, Music genre classification based on ensemble of signals produced by source separation methods, IOS press. Intell. Decis. Technol. **4**, 229–237 (2010)
27. K Rao, S Koolagudi, Recognition of emotions from video using acoustic and facial features. Journal of Signal Image & Video processing, 1-17 (2013)
28. H Maniya, M Hasan, Comparative study of naïve Bayes classifier and KNN for tuberculosis, in *International Conference on Web Services Computing (ICWSC)*, 2011
29. MAW Saduf, Comparative study of back propagation learning algorithms for neural networks. Int J. Adv. Res. Comp. Sci. Softw. Eng. **3**(12), 1151–1156 (2013)
30. O Martinez, M Cyrill, S Burgard, *Supervised learning of places from range data using Adaboost. Proceeding of the 2005 IEEE international conference on robotics and automation*, 2005, pp. 1730–1735
31. Y Jizheng, X Mao, Y Xue, Compare, Facial expression recognition based on t-SNE and AdaboostM2. IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, GreenCom-iThings-CPSCom, 1744-1749, (2013)
32. Y Freund, R Schapiro, A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst Sci. **55**(1), 119–139 (1997)
33. J Rodrı´guez, A Pe´rez, J Lozano, Sensitivity analysis of k-fold cross validation in prediction error estimation. Pattern Anal. Mach. Intell., IEEE Trans. **32**(3), 569–575 (2010)
34. I Junejo, A Bhutta, H Foroosh, Single-class SVM for dynamic scene modeling. J. Sig. Image Video Process. **7**(1), 45–52 (2011)
35. G Williams, Instantaneous receiver operating characteristic (ROC) performance of multi-gain-stage APD photoreceivers. IEEE J Electr Dev. Soc. **1**(6), 145–153 (2013)
36. J Youden, Index for rating diagnostic tests. Cancer **3**, 32–35 (1950)