I J C T A, 9(37) 2016, pp. 361-368

Automatic Video Annotation of Singer in a Video Song using Spectral and Cepstral Features

S. Metilda Florence*

Abstract: Automatic Video Annotation refers to the extraction of information about the video contents automatically. The extracted information can serve as the first step for different data access methods such as surfing, searching, comparison, and classification. It is worth mentioning that annotating music information in a video is an emerging task and was not much covered in past research papers. Massive amount of video songs reachable to the public calls for developing tools to efficiently retrieve and manage the music of interest to the end users. Thus, the Automatic Video Annotation of Singer in a Video Song System enables the user to search for their favorite Singer's video song. Music classification in a video song can be categorized as Genre Classification (Rock, Pop, Classical etc.), Mood Classification (Happy, Sad, Angry etc.), Instrument Recognition (Piano, Violin etc.) and Artist Identification (Singer identification). The proposed System focus on Artist Identification and the work on other classifications are published in proceedings and Journal by the same author. The Proposed System performs the search in a video store by comparing the content of the video and not the user's textual query and tags associated with the videos. An effort is taken for identifying Singer in the video songs by mining their audio features like Spectral and Cepstral features. Analysis of various classification algorithms to study, train and check the model representing the singer of a video song are presented. The experimental outcomes show that the user can retrieve the video songs on their interest.

Index Terms: Content Based Search, Video annotation, Artist Identification, Spectral Features, Mel Frequency Cepstral Coefficients.

1. INTRODUCTION

A recent statistics of YouTube states that it has more than 1 billion users. Each day people watch hundreds of millions of hours of videos on YouTube. In near future watching online videos will increase in huge amount. In order to reuse the materiala vailable in a video store, there is a requirement to annotate the accessible material. In several video production companies, this task is still executed manually. The proposed technique will annotate the video automatically from the audio information. The main contribution of this paper is the use of music to annotate video, which is a much less explored problem. Music classification in a video song can be categorized as Genre Classification (Rock, Pop, Classical etc.), Mood Classification (Happy, Sad, Angry etc.), Instrument Recognition (Piano, Violin etc.) and Artist Identification (Singer identification). The proposed System focus on Artist Identification and the work on other classifications are published in proceedings and Journal by the same author [1, 2 and 3].For Singer classification, three legends namely S.P. Balasubramaniam (SPB), P. Susheela and Swarnalatha are selected. More contributing video clips of 10 seconds duration are collected from Internet and Video CDs. For each singer 100 video songs are taken. From each video song the vocal track alone extracted. Mathematical functions are applied to calculate the MFCC and Spectral features. These features are directly applied to standard classifiers for classification. Since, there is no single classification algorithm which is recognized to perform well for all applications. It is required to carry out a comparative study on the same set of signals to determine the best classifier. The classification accuracy is not only depends on the classifier, but also the strength of the features that are extracted.

* Assistant Professor, Department of Information Technology, SRM University, Kattankulathur, India

2. PROPOSED SYSTEM

The proposed Automatic Video Annotation of Singer in a Video Song System depicted in Figure 1. More contributing video clips from VCDs and Internet are collected. Each collected video file is processed by transforming and trimming it to 10 seconds duration. Audio track from video clips are extracted with a sampling rate of 44.1 kHz. From these extracted audio tracks the vocal are isolated and then processed by the feature extraction phase in order to extract the features. By using efficient classifiers the extracted feature set is classified based on Singers. Five different classifiers are used to train and test the dataset.

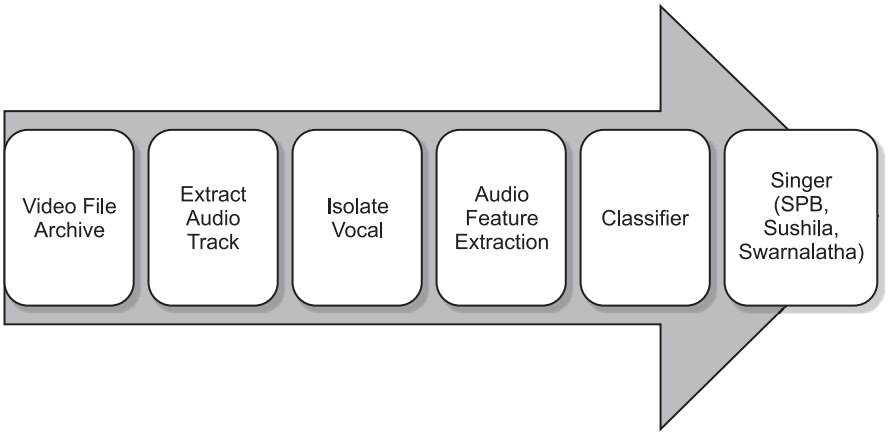

Figure 1: Block diagram of proposed System

A. Video File Archive and Audio Track Extraction

In the initial stage of this work, video files are collected from the internet and Video CDs. For each singer 100 songs are collected. Duration of all songs was equally set to 10 seconds. During this short period we can cut only the singing part of the song by avoiding intro, chorus, outro etc. This will help us to extract only the singer voice with back ground music. Using Matlab code audio track is extracted without opening each video files. This audio track contains both vocal and instrumental music in it.

B. Vocal Isolation

In this phase, vocal is isolated from the extracted audio track. The majority of energy in the singing voice falls between 200 Hz and 2000 Hz (there may be with certain deviation depending on the singer). Since frequency range of singing is concerned, a direct method would be used to detect energy within the frequencies bounded by the range of vocal energy. A simple method is to filter the audio signal with a band-pass filter which permits the vocal range to pass through while weakening other frequency areas. To achieve this Chebychev infinite-impulse response (IIR) digital filter of order 12 is used. This filter has the musical effect of suppress other instruments that fall outside of this frequency region. But even in popular music, the voice is not the only instrument creating energy in this region. Drum,for example, scatter energy over an extensive collection of frequencies, a significant amount of which falls in our range of interest. So another measure is needed to separate the voice from these other sources.

Singing voice is highly harmonic [4] and other high energy sounds in this region, particularly Drum are not as harmonic and distribute their energy more widely in frequency. To exploit this difference, an inverse comb filter bank is used to detect high amounts of harmonic energy. By passing the filtered signal (F) through a set of inverse comb filters with varying delays, we can find the fundamental frequency which the signal is most weakened. By taking the ratio of the total signal energy to the maximally harmonic attenuated signal, Harmonicity is measured (1).

$$H_{armonicity} = \frac{F_{original}}{MIN_i(F_{filtered,\,i})} \tag{1}$$

By thresholding the Harmonicity against a fixed value, we have a detector for harmonic sounds. Most of these signals correspond to region of singing.

C. Feature Extraction

Music signal is described using various numerical values extracted from the signal. These are called as features of the signal. A large amount of different feature sets, mainly originating from the area of speech recognition, have been proposed to characterize audio signals [5]. The features used to signify timbre texture are based on typical features proposed for music-speech separation [6]. From the available features the most relevant features are selected for this System. They are Spectral and Ceptral features. Spectral features used are Spectral Centroid, Spectral Rolloff and Spectral Flux. Cepstral feature used is MFCC.

The purpose of feature extraction is to preserve useful information, eliminate noise and other unwanted information. Aforesaid features are extracted by the steps given in the following sections. Finally all these features are combined together for creating the dataset for our System.

1. *Spectral Centroid*: 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 a sound. It is calculated as the weighted mean of the frequencies present in the signal, determined using a Fast Fourier Transform, with their magnitudes as the weights:

$$c = \frac{\sum_{n=0}^{N-1} S(n)\, d(n)}{\sum_{n=0}^{N-1} d(n)} \tag{2}$$

where $d(n)$ represents the weighted frequency value, or magnitude of bin number n, and $S(n)$ represents the center frequency of that bin.

2. *Spectral Rolloff*: The *Spectral Rolloff* is the frequency R_t under which 95% of the power distribution is concentrated.

$$\sum_{n=1}^{R_t} S_t[n] = 0.85 \times \sum_{n=1}^{N} S_t[n] \tag{3}$$

Where n is time index ranging from $0 <= n <= N - 1$, N is duration of file and t is current time frame.

3. *Spectral Flux:* The *Spectral flux* is defined as the squared difference between the normalized magnitudes of successive spectral distributions.

$$SF = \sum (F_t[n] - F_{t-1}[n])^2 \tag{4}$$

where $F_t[n]$ and $F_{t-1}[n]$ are the normalized magnitude of the Fourier transform at the current time frame t, and the previous time frame $t - 1$, respectively.

4. *Mel-Frequency Cepstral Coefficients(MFCC):* The MFCC is a very common and efficient technique for signal processing. It describes the spectral shape of the signal. Its computation involves five main steps, including the conversion of signal frame into a Mel scale representation [7] in order to emphasize the middle frequency bands. The MFCC transformation has been proved useful for computing music similarity [8].

To extract MFCC features, the audio signal is divided into a number of overlapped frames. To minimize the ringing effect [9], multiply each frame by a Hamming window *hwd*(*h*) is given in (5).

$$hwd(h) = 0.54 - 0.46\cos\left(\frac{2\Pi h}{\mathrm{N}-1}\right), 0 \le h \le \mathrm{N}-1 \tag{5}$$

where N is the length of Hamming window. FFT is then applied on each pre-emphasized, Hamming windowed frame to obtain the corresponding spectrum. The audio samples are sampled at 44.1 KHz. To extract features, music samples are segmented into 23 milliseconds (ms) frames to get accurate FFT [10]. When compared with other sample rates and segment size combinations, 44.1 KHz and 23 ms gives the best performance [11]. For windowing Hamming window is used because combination of Mel frequency and Hamming window gives good results [12]. For each window, thirteen MFCC coefficients are calculated. Feature vector of MFCC for each window of 23 ms is obtained. As the size of this feature vector is very large, instead of using these features directly for classification, their mean and standard deviations are obtained. Totally it produces (13 standard deviation values, 13 mean values) 26 features.

D. Classifiers

It is necessary to use more than one classifier to get the average accuracy. In the proposed System, five efficient classifiers are used to train and test the dataset. They are given as follows: Naive Bayes, Sequential Minimal Optimization (SMO), Multiclass Classifier, J48 and Random Tree.

1. *Naive Bayes Classifier:* The Naive Bayes algorithm is based on Bayes rule that assumes the attributes $X_1 \ldots X_n$ are all conditionally independent of one another, given Y. In this case

$$\mathrm{P}(\mathrm{X}_1 \ldots \mathrm{X}_n \mid \mathrm{Y}) = \prod_{i=1}^{n} \mathrm{P}(\mathrm{X}_i \mid \mathrm{Y}) \tag{6}$$

   Where conditionally independent defined as "Given random variables X;Y and Z, we say X is *conditionally independent* of Y given Z, if and only if the probability distribution governing X is independent of the value of Y given Z; that is $(\forall i, j, k)\ \mathrm{P}(\mathrm{X} = x_i \mid \mathrm{Y} = y_j \mid \mathrm{Z} = z_k) = \mathrm{P}(\mathrm{X} = x_i \mid \mathrm{Z} = z_k)$"

2. *Sequential Minimal Optimization (SMO):* SMO is an algorithm that quickly solves the Support Vector Machine Quadratic Programming (SVM QP) problem without any extra matrix storage and without calling an iterative mathematical routine for each sub-problem [13]. SMO divided the complete QP problem into QP sub-problems. Contrasting to the previous methods, SMO chooses to solve the smallest possible optimization problem at each step. For the SVM QP problem, the lowest possible optimization problem comprises of two Lagrange multipliers because the Lagrange multipliers essentially follow a linear equality constraint. At every step, SMO chooses two Lagrange multipliers to jointly optimize, calculates the optimal values for these multipliers, and updates the SVM to reflect the new optimal values. It was implemented by John Platt.

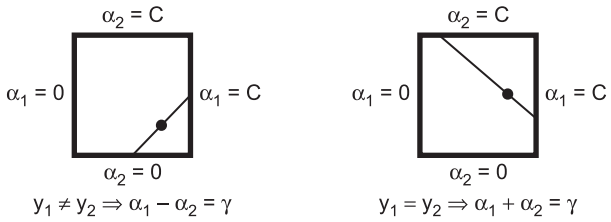

Figure 2: SMO classifier

The two Lagrange multipliers must satisfy constraints of the full problem. In Figure 2 it shows the inequality constraints that cause the Lagrange multipliers to fall in the box. The linear equality constraint causes them to fall on a diagonal line. Therefore, one step of SMO must find an optimum of the objective function on a diagonal line segment. In this figure, $\gamma = \alpha_1^{old} + s\alpha_2^{old}$ is a constant that depends on the previous values of α_1 and α_2, $s = y_1 y_2$.

3. *Multiclass classifier:* Multiclass classification can be achieved by any one of the following ways: (a) One versus All based Multi Class Classification (b) All versus All based Multi Class Classification (c) Error Correcting Code based Multi Class Classification. The frequently used approach to multiclass classification is the *All versus All* (AvA) approach, makes direct use of standard binary classifiers to encode and train the output labels [14].

   AvA is used for classifying our dataset. It is faster and more memory efficient. It requires O (N^2) classifiers instead of O (N), but each classifier is much smaller.

   In AvA, build N(N-1) classifiers, one classifier to distinguish each pair of classes *l* and *m*. Let f_{lm} be the classifier where class *l* were positive examples and class *m* were negative . Note $f_{ml} = -f_{lm}$. Classify using (7).

$$f(x) = \arg^{\max_l (\Sigma_m f_{lm}(x))} \tag{7}$$

4. *Random Forest classifier:* "Random Forests produces several classification trees. To classify a new object from an input vector, put the input vector below each of the trees in the forest. Every tree provides a classification and the tree votes for that class. The forest selects the classification having the maximum votes".

   Each tree is grown as follows:

   1. If the number of cases in the training set is X, sample X cases at random - but *with replacement*, from the original data. This sample will be the training set for developing the tree.
   2. If there are Y input variables, a number $y << Y$ is specified such that at each node, m variables are selected at random out of the Y and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
   3. Each tree is developed to the largest degree possible. There is no pruning.

5. *J48 classifier:* J48 implements Quinla's C4.5 algorithm for generating a pruned or unpruned C4.5 decision tree. C4.5 is an extension of Quinlan's former ID3 algorithm [15]. The decision trees generated by J48 can be used for classification. J48 creates decision trees from a set of labeled training data using the concept of information entropy. It uses the fact that each attribute of the data can be used to make a decision by splitting the data into reduced subsets. J48 inspects the normalized information gain that results from choosing an attribute for splitting the data. To make the decision, the attribute with the highest normalized information gain is used. Then the algorithm recurs on the reduced subsets. The splitting process stops if all instances in a subset belong to the same class. Then a leaf node is created in the decision tree telling to choose that class. But there is a chance that none of the features give any information gain. In this case J48 builds a decision node higher up in the tree using the expected value of the class. J48 can handle both continuous and discrete attributes, training data with missing attribute values and attributes with differing costs. Further it provides an option for trimming trees after creation.

3. EXPERIMENTAL RESULTS AND DISCUSSION

A. Dataset

From the internet and VCDs 300 video songs are collected. By using video cutter tool the video songs are trimmed to 10 seconds duration. In Matlab, all the video files are read one by one to extract the audio track and to isolate the vocal. From the isolated vocal track Spectral and Cepstral features are extracted.

B. Classification

For classification , we used WEKA tool [16] in that 10 cross validation technique is used for classification. Cross-validation is a model validation technique for assessing how the results of statistical analysis will generalize to an independent dataset [17]. The confuson matrix for each classifiers are given in Table 1.

Table 1
Confusion matrices for different classifiers.

Table 1(a)
Naïve Bayes Classifier

Singers	*SPB*	*Sushila*	*Swarna*
SPB	95	1	4
Sushila	1	94	5
Swarna	8	5	87

Table 1(b)
Sequential Minimal Optimization

Singers	*SPB*	*Sushila*	*Swarna*
SPB	97	0	3
Sushila	1	96	3
Swarna	2	5	93

Table 1(c)
Multiclass classifier

Singers	*SPB*	*Sushila*	*Swarna*
SPB	96	0	4
Sushila	5	92	3
Swarna	4	7	89

Table 1(d)
Random Forest classifier

Singers	*SPB*	*Sushila*	*Swarna*
SPB	96	1	3
Sushila	1	94	5
Swarna	4	8	88

Table 1(e)
J48 classifier

Singers	*SPB*	*Sushila*	*Swarna*
SPB	81	2	17
Sushila	2	89	9
Swarna	8	9	83

Table 2
Analysis Report

Classifier	*Accuracy %*			*Overall %*
	SPB	*Sushila*	*Swarnalatha*	
Naïve Bayes	95	94	87	92
SMO	97	96	93	95
Multiclass	96	92	89	92
Random Forest	96	94	88	93
J48classifier	81	89	83	84

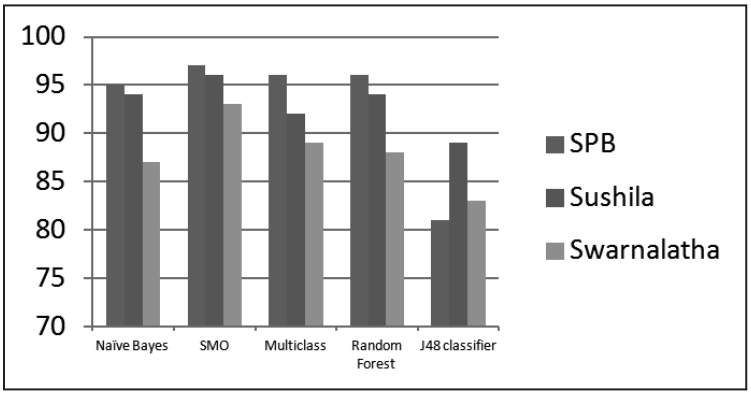

Figure 3: Comparison of different classifiers performance

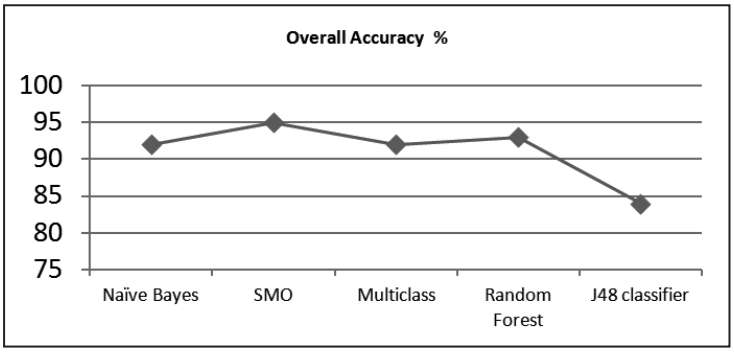

Figure 4: Overall Classification Accuracy for different classifiers

Table 1 depicts confusion matrices for all the five classifiers. In Table 2, the overall accuracy of all these classifiers is listed. These tables show that, SMO classifier has given the highest accuracy of 95 % and J48 classifier given the least accuracy of 84%. To conclude, Artist identification module gives a maximum of 95% accuracy in identifying a Singer in a video song using SMO classifier. Figure 3 and 4 give the pictorial representations of the accuracy percentages.

4. CONCLUSION

A novel and efficient approach for Identifying a Singer in Video Song is presented. This will enable the music lovers to locate their favorite Singer's Video Song. Current search engines will search the video by their tags not by content. Our proposed system will identify the song based on the Voice track in the video. In the proposed System three Singers namely SPB, P. Susheela and Swarnalatha are selected for analysis. For each Singer 100 video songs of length 10 seconds duration are taken. From these video songs the vocal track alone are extracted by using IIR digital filter and inverse comb filter. Mathematical functions are applied to calculate the Spectral and Cepstral features from the extracted signal. These features are applied to five classifiers for classification. This System gives a maximum of 95% accuracy in identifying a Singer in a video song using SMO classifier. We achieved maximum of 95 % accuracy. The proposed system is capable of identifying the singer for 10 seconds duration. This can be extended for the entire song. In future, this work can be extended to cover more singers and also we can increase the size of dataset by including more contributing features from the audio track. This may increase the accuracy percentage.

References

1. S. Metilda Florence, Dr. S. Mohan, "*Automatic Video Annotation for Music Genre Based on Spectral and Cepstral Features*", in ELSEVIER Proc. Int. Conf. on Applied Information and Communications Technology (ICAICT 2014),Oman, pp. 27-32.

2. S. Metilda Florence, Dr. S. Mohan, "*Automatic Video Annotation for Music Mood using PCA with Rhythm and Cepstral Features*", in ELSEVIER Proc. Int. Conf. on Emerging Research in Computing, Information, Communication and Applications, ERCICA 2014, BANGALORE, pp. 355-360.
3. S. Metilda Florence, Dr. S. Mohan, "A Novel Search Engine for Identifying Musical Instruments in a Video File",in International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015) pp. 34144-34148
4. P. R. Cook, *Identification of Control Parameters in an Articulatory Vocal Tract Model, with Applications to the Synthesis of Singing*. Ph.D. Thesis. Stanford University,Stanford, CA, 1990.
5. George Tzanetakis, Perry Cook ,"Musical Genre Classification of Audio Signals", IEEE Transactions On Speech And Audio Processing, Vol. 10, No. 5, July 2002.
6. E. Scheirer and M. Slaney, "*Construction and evaluation of a robust multifeature speech/music discriminator*," in Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP), 1997, pp. 1331–1334.
7. Logan .B, "*Mel frequency cepstral coefficients for music modeling*" in International symposium on music information retrieval, 2000.
8. Foote, J. (1999). Visualizing music and audio using self-similarity. In Proceedings of 7th ACM international conference on multimedia (part 1) (pp. 77–80).
9. D. M. Chandwadkar, Dr. M. S. Sutaone "*Role of Features and Classifiers on Accuracy of Identification of Musical Instruments*" in Proceedings of CISP 2012.
10. R. Duda, P. Hart, and D. Stork, *Pattern Classification*, New York: Wiley, 2000.
11. Jouni Paulus, "*Acoustic Modelling Of Drum Sounds With Hidden Markov Models For Music Transcription*" in IEEE Transactions, 2006.
12. Md. Rashidul Hasan, Mustafa Jamil, Md. Golam Rabbani Md. Saifur Rahman, "*Speaker Identification Using Mel Frequency Cepstral Coefficients*" in 3rd International Conference on Electrical & Computer Engineering ICECE 2004, 28-30 December 2004, Dhaka, Bangladesh.
13. John C. Platt, "*Fast Training of Support Vector Machines using Sequential Minimal Optimization*" in 2000(Book).
14. Sariel Har-Peled, Dan Roth, Dav Zimak ,"Constraint Classification for Multiclass Classification and Ranking" in Advances in neural information, 2003.
15. http://www.opentox.org/dev/documentation/components/j48
16. Ian H. Witten, Eibe Frank; "*Data Mining- Practical Machine Learning Tools and Techniques*", Second Edition; Morgan Kaufmann Publishers: An Imprint Of Elsevier, 2005.
17. Davide Anguita, Luca Ghelardoni, Alessandro Ghio, Luca Oneto and Sandro Ridella, "*The 'K' in K-fold Cross Validation*" in ESANN 2012 proceedings.