Expert Systems with Applications 36 (2009) 6069–6075

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Classification of audio signals using SVM and RBFNN

P. Dhanalakshmi *, S. Palanivel, V. Ramalingam

Department of Computer Science and Engineering, Annamalai University, Annamalainagar, Chidambaram 608 002, Tamil Nadu, India

ARTICLE INFO

Keywords:
Support vector machines
Radial basis function neural network
Linear predictive coefficients
Linear predictive cepstral coefficients
Mel-frequency cepstral coefficients

ABSTRACT

In the age of digital information, audio data has become an important part in many modern computer applications. Audio classification has been becoming a focus in the research of audio processing and pattern recognition. Automatic audio classification is very useful to audio indexing, content-based audio retrieval and on-line audio distribution, but it is a challenge to extract the most common and salient themes from unstructured raw audio data. In this paper, we propose effective algorithms to automatically classify audio clips into one of six classes: music, news, sports, advertisement, cartoon and movie. For these categories a number of acoustic features that include linear predictive coefficients, linear predictive cepstral coefficients and mel-frequency cepstral coefficients are extracted to characterize the audio content. Support vector machines are applied to classify audio into their respective classes by learning from training data. Then the proposed method extends the application of neural network (RBFNN) for the classification of audio. RBFNN enables nonlinear transformation followed by linear transformation to achieve a higher dimension in the hidden space. The experiments on different genres of the various categories illustrate the results of classification are significant and effective.

1. Introduction

Audio data is an integral part of many modern computer and multimedia applications. A typical multimedia database often contains millions of audio clips, including environmental sounds, machine noise, music, animal sounds, speech sounds, and other non-speech utterances. The effectiveness of their deployment is greatly dependent on the ability to classify and retrieve the audio files in terms of their sound properties or content. The need to automatically recognize to which class an audio sound belongs makes audio classification and categorization an emerging and important research area. However, a raw audio signal data is a featureless collection of bytes with most rudimentary fields attached such as name, file format and sampling rate.

Content-based classification and retrieval of audio sound is essentially a pattern recognition problem in which there are two basic issues: feature selection, and classification based on the selected features. In the first step, an audio sound is reduced to a small set of parameters using various feature extraction techniques. The terms linear predictive coefficients (LPC), linear predictive cepstral coefficients (LPCC), mel-frequency cepstral coefficients (MFCC) refer to the features extracted from audio data. In the second step, classification or categorization algorithms ranging from simple Euclidean distance methods to sophisticated statistical techniques are carried out over these coefficients. The efficacy of an audio classification or categorization depends on the ability to capture proper audio features and to accurately classify each feature set corresponding to its own class.

1.1. Related work

During the recent years, there have been many studies on automatic audio classification and segmentation using several features and techniques. The most common problem in audio classification is speech/music classification, in which the highest accuracy has been achieved, especially when the segmentational information is known beforehand. In Lin, Chen, Truong, and Chang (2005), wavelets are first applied to extract acoustical features such as subband power and pitch information. The method uses a bottom-up SVM over these acoustic features and additional parameters, such as frequency cepstral coefficients, to accomplish audio classification and categorization. An audio feature extraction and a multigroup classification scheme that focuses on identifying discriminatory time–frequency subspaces using the local discriminant bases (LDB) technique has been described in Umapathy, Krishnan, and Rao (2007). For pure music and vocal music, a number of features such as LPC and LPCC are extracted in Xu, Maddage, and Shao (2005), to characterize the music content. Based on calculated features, a clustering algorithm is applied to structure the music content.

A new approach towards high performance speech/music discrimination on realistic tasks related to the automatic transcription

* Corresponding author. Tel.: +91 04144 220433.
E-mail addresses: abi_dhana@rediffmail.com (P. Dhanalakshmi), spal_yughu@yahoo.com (S. Palanivel), aucsevr@yahoo.com (V. Ramalingam).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.06.126

of broadcast news is described in Ajmera, McCowan, and Bourlard (2003), in which an artificial neural network (ANN) and hidden Markov model (HMM) are used. In Kiranyaz, Qureshi, and Gabbouj (2006), a generic audio classification and segmentation approach for multimedia indexing and retrieval is described. A method is proposed in Panagiotakis and Tziritas (2005) for speech/music discrimination based on root mean square and zero-crossings. The method proposed in Eronen et al. (2006), investigates the feasibility of an audio-based context recognition system where simplistic low-dimensional feature vectors are evaluated against more standard spectral features. Using discriminative training, competitive recognition accuracies are achieved with very low-order hidden Markov models (Ajmera et al., 2003).

The classification of continuous general audio data for content-based retrieval was addressed in Li, Sethi, Dimitrova, and McGee (2001), where the audio segments where classified based on MFCC and LPC. They also showed that cepstral-based features gave a better classification accuracy. The method described in content-based audio classification and retrieval using joint time–frequency analysis exploits the non-stationary behavior of music signals and extracts features that characterize their spectral change over time (Esmaili, Krishnan, & Raahemifar, 2004). The audio signals were decomposed in Umapathy, Krishnan, and Jimaa (2005), using an adaptive time frequency decomposition algorithm, and the signal decomposition parameter based on octave (scaling) was used to generate a set of 42 features over three frequency bands within the auditory range. These features were analyzed using linear discriminant functions and classified into six music groups. An approach given in Jiang, Bai, Zhang, and Xu (2005), uses support vector machine (SVM) for audio scene classification, which classifies audio clips into one of five classes: pure speech, non-pure speech, music, environment sound, and silence. Radial basis function neural networks (RBFNN) are used in McConaghy, Leung, Boss, and Varadan (2003) to classify real-life audio radar signals that are collected by a ground surveillance radar mounted on a tank.

For audio retrieval, a new metric has been proposed in Guo and Li (2003), called distance-from-boundary (DFB). When a query audio is given, the system first finds a boundary inside which the query pattern is located. Then, all the audio patterns in the database are sorted by their distances to this boundary. All boundaries are learned by the SVMs and stored together with the audio database. In Mubarak, Ambikairajah, and Epps (2005) a speech/music discrimination system was proposed based on mel-frequency cepstral coefficient (MFCC) and GMM classifier. This system can be used to select the optimum coding scheme for the current frame of an input signal without knowing a priori whether it contains speech-like or music-like characteristics. A hybrid model comprised of Gaussian mixtures models (GMMs) and hidden Markov models (HMMs) is used to model generic sounds with large intra class perceptual variations in Rajapakse and Wyse (2005). The number of mixture components in the GMM was derived using the minimum description length (MDL) criterion.

A new pattern classification method called the nearest feature line (NFL) is proposed in Li (2000), where the NFL explores the information provided by multiple prototypes per class. Audio features like MFCC, ZCR, brightness and bandwidth, spectrum flux were extracted (Lu, Zhang, & Li, 2003), and the performance using SVM, *K*-nearest neighbor (KNN), and Gaussian mixture model (GMM) were compared. Audio classification techniques for speech recognition and audio segmentation, for unsupervised multispeaker change detection are proposed in Huang and Hansen (2006). Two new extended-time features: variance of the spectrum flux (VSF) and variance of the zero-crossing rate (VZCR) are used to pre-classify the audio and supply weights to the output probabilities of the GMM networks. The classification is then implemented using weighted GMM networks.

1.2. Outline of the work

In this paper, automatic audio feature extraction and classification approaches are presented. In order to discriminate the six categories of audio namely music, news, sports, advertisement, cartoon and movie, a number of features such as LPC, LPCC, MFCC are extracted to characterize the audio content. Support vector machine (SVM) is applied to obtain the optimal class boundary between the classes by learning from training data. The performance of SVM is compared to RBF network. Each RBF center approximates a cluster of training data vectors that are close to each other in Euclidean space. When a vector is input to the RBFNN, the centers near to that vector become strongly activated, in turn activating certain output nodes. Experimental results show that the classification accuracy of RBF with mel cepstral features can provide a better result. Fig. 1 illustrates the block diagram of audio classification.

The paper is organized as follows. The acoustic feature extraction is presented in Section 2, modeling techniques for audio classification is described in Section 3. Experimental results using SVM and RBF are reported in Section 4. Finally, conclusions and future work are given in Section 5.

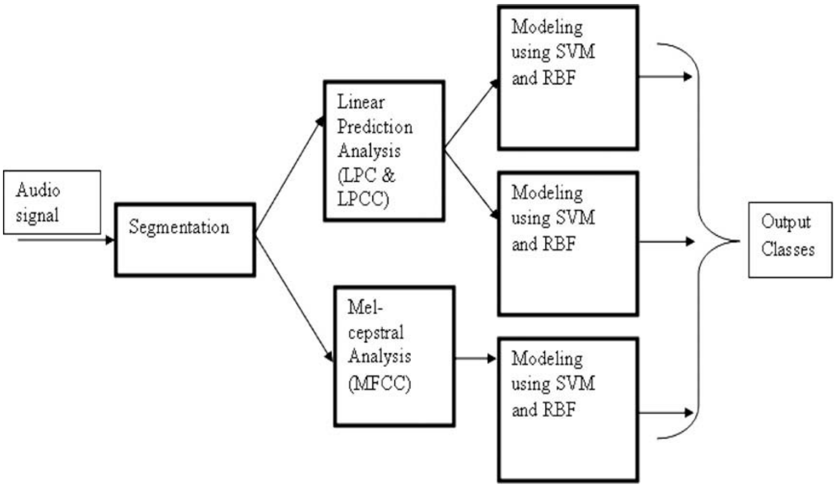
Audio signal
Segmentation
Linear Prediction Analysis (LPC & LPCC)
Mel-cepstral Analysis (MFCC)
Modeling using SVM and RBF
Modeling using SVM and RBF
Modeling using SVM and RBF
Output Classes

Fig. 1. Block diagram of audio classification.

2. Acoustic feature extraction

Acoustic features representing the audio information can be extracted from the speech signal at the segmental level. The segmental features are the features extracted from short (10–30 ms) segments of the speech signal. These features represent the short-time spectrum of the speech signal. The short-time spectrum envelope of the speech signal is attributed primarily to the shape of the vocal tract. The spectral information of the same sound uttered by two persons may differ due to change in the shape of the individual's vocal tract system, and the manner of speech production. The selected features include linear prediction coefficients (LPC), Linear prediction derived cepstrum coefficients(LPCC) and mel-frequency cepstral coefficients (MFCC).

2.1. Linear prediction analysis

For acoustic feature extraction, the differenced speech signal is divided into frames of 20 ms, with a shift of 5 ms. A pth order LP analysis is used to capture the properties of the signal spectrum.

In the LP analysis of speech each sample is predicted as linear weighted sum of the past p samples, where p represents the order of prediction (Rabiner & Schafer, 2005; Palanivel, 2004). If $s(n)$ is the present sample, then it is predicted by the past p samples as

$$\hat{s}(n) = -\sum_{k=1}^{p} a_k s(n-k). \tag{1}$$

The difference between the actual and the predicted sample value is termed as the prediction error or residual, and is given by

$$e(n) = s(n) - \hat{s}(n) = s(n) + \sum\nolimits_{k=1}^{p} a_k s(n-k) \tag{2}$$

$$= \sum_{k=0}^{p} a_k s(n-k), \quad a_0 = 1 \tag{3}$$

The LP coefficients $\{a_k\}$ are determined by minimizing the mean squared error over an analysis frame and it is described in Rabiner and Juang (2003).

The recursive relation (4) between the predictor coefficients and cepstral coefficients is used to convert the LP coefficients into LP cepstral coefficients $\{c_k\}$

$$\begin{aligned} c_0 &= \ln \sigma^2, \\ c_m &= a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k a_{m-k}, \quad 1 \leqslant m \leqslant p, \\ c_m &= \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k a_{m-k}, \quad p < m \leqslant D, \end{aligned} \tag{4}$$

where σ^2 is the gain term in the LP analysis and D is the number of LP cepstral coefficients.. The cepstral coefficients are linearly weighted to get the weighted linear prediction cepstral coefficients (WLPCC). In this work, a 19 dimensional WLPCC is obtained from the 14th order LP analysis for each frame. Linear channel effects are compensated to some extent by removing the mean of the trajectory of each cepstral coefficient. The 19 dimensional WLPCC (mean subtracted) for each frame is used as an acoustic feature vector.

2.2. Mel-frequency cepstral coefficients

The mel-frequency cepstrum has proven to be highly effective in recognizing structure of music signals and in modeling the subjective pitch and frequency content of audio signals. Psychophysical studies have found the phenomena of the mel pitch scale and the critical band, and the frequency scale-warping to the mel scale has led to the cepstrum domain representation. The mel scale is defined as

$$F_{\text{mel}} = \frac{c \log\left(1 + \frac{f}{c}\right)}{\log(2)}, \tag{5}$$

where F_{mel} is the logarithmic scale of f normal frequency scale. The mel-cepstral features (Xu et al., 2005), can be illustrated by the MFCCs, which are computed from the fast Fourier transform (FFT) power coefficients. The power coefficients are filtered by a triangular bandpass filter bank. When c in (5) is in the range of 250–350, the number of triangular filters that fall in the frequency range 200–1200 Hz (i.e., the frequency range of dominant audio information) is higher than the other values of c. Therefore, it is efficient to set the value of c in that range for calculating MFCCs. Denoting the output of the filter bank by S_k $(k = 1, 2, \ldots, K)$, the MFCCs are calculated as

$$c_n = \sqrt{\frac{2}{K}} \sum_{k=1}^{K} (\log S_k) \cos\left[n(k-0.5)\frac{\pi}{K}\right], \quad n = 1, 2 \ldots, L. \tag{6}$$

3. Modeling techniques for audio classification

3.1. Support vector machine (SVM) for classification

The SVM (Vapnik, 1998), is a useful statistic machine learning technique that has been successfully applied in the pattern recognition area (Jiang et al., 2005; Guo & Li, 2003). If the data are linearly nonseparable but nonlinearly separable, the nonlinear support vector classifier will be applied. The basic idea is to transform input vectors into a high-dimensional feature space using a nonlinear transformation ϕ, and then to do a linear separation in feature space as shown in Fig. 2.

To construct a nonlinear support vector classifier, the inner product (x,y) is replaced by a kernel function $K(x,y)$:

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{l} \alpha_i y_i K(x_i, x) + b\right). \tag{7}$$

The SVM has two layers. During the learning process, the first layer selects the basis $K(x_i, x)$, $i = 1, 2, \ldots, N$, from the given set of bases defined by the kernel; the second layer constructs a linear function in this space. This is completely equivalent to constructing the optimal hyperplane in the corresponding feature space.

The SVM algorithm can construct a variety of learning machines by use of different kernel functions. Three kinds of kernel functions are usually used. They are as follows:

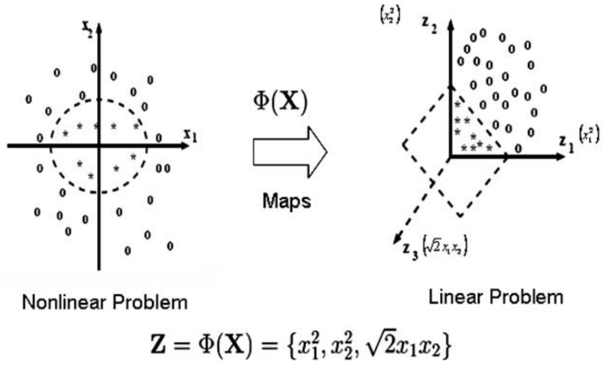

Fig. 2. Principle of support vector machines.

1. Polynomial kernel of degree d:

$$K(X,Y) = (\langle X,Y\rangle + 1)^d. \tag{8}$$

2. Radial basis function with Gaussian kernel of width $C > 0$:

$$K(X,Y) = \exp\left(\frac{-|X-Y|^2}{c}\right). \tag{9}$$

3. Neural networks with tanh activation function:

$$K(X,Y) = \tan h(K\langle X,Y\rangle + \mu), \tag{10}$$

where the parameters K and μ are the gain and shift.

3.2. Radial basis function neural network (RBFNN) model for classification

The radial basis function neural network (Haykin, 2001) has a feed forward architecture with an input layer, a hidden layer, and an output layer as shown in Fig. 3. Radial basis functions are embedded into a two-layer feed forward neural network. Such a network is characterized by a set of inputs and a set of outputs. In between the inputs and outputs there is a layer of processing units called hidden units. Each of them implements a radial basis function. The input layer of this network has n_i units for a n_i dimensional input vector. The input units are fully connected to the n_h hidden layer units, which are in turn fully connected to the n_c output layer units, where n_c is the number of output classes. The activation functions of the hidden layer were chosen to be Gaussians, and are characterized by their mean vectors (centers) $\boldsymbol{\mu}_i$, and covariance matrices C_i, $i = 1, 2, \ldots, n_h$. For simplicity, it is assumed that the covariance matrices are of the form $C_i = \sigma_i^2 I$, $i = 1, 2, \ldots, n_h$. Then the activation function of the ith hidden unit for an input vector $\boldsymbol{x}_j$ is given by

$$g_i(\boldsymbol{x}_j) = \exp\left(\frac{-\|\boldsymbol{x}_j - \boldsymbol{\mu}_i\|^2}{2\sigma_i^2}\right). \tag{11}$$

The $\boldsymbol{\mu}_i$ and σ_i^2 are calculated by using suitable clustering algorithm. Here the k-means clustering algorithm is employed to determine the centers. The algorithm is composed of the following steps:

1. Randomly initialize the samples to k means (clusters) $\boldsymbol{\mu}_1, \ldots, \mu_k$.

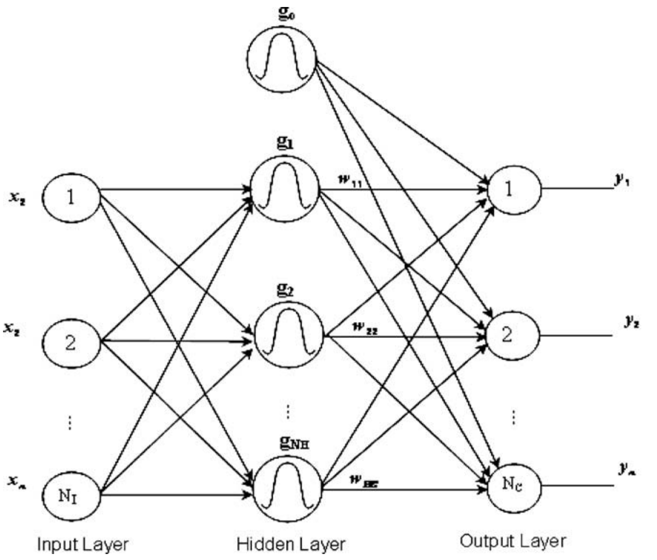

Fig. 3. Radial basis function neural network.

2. Classify n samples according to nearest $\boldsymbol{\mu}_k$.
3. Recompute $\boldsymbol{\mu}_k$.
4. Repeat steps 2 and 3 until no change in μ_k.

The number of activation functions in the network and their spread influence the smoothness of the mapping. The assumption $\sigma_i^2 = \sigma^2$ is made and σ^2 is given in (12) to ensure that the activation functions are not too peaked or too flat:

$$\sigma^2 = \frac{\eta d^2}{2}. \tag{12}$$

In the above equation d is the maximum distance between the chosen centers, and η is an empirical scale factor which serves to control the smoothness of the mapping function. Therefore, the above equation is written as

$$g_i(\boldsymbol{x}_j) = \exp\left(\frac{-\|\boldsymbol{x}_j - \boldsymbol{\mu}_i\|^2}{\eta d^2}\right). \tag{13}$$

The hidden layer units are fully connected to the n_c output layer units through weights w_{ik}. The output units are linear, and the response of the kth output unit for an input $\boldsymbol{x}_j$ is given by

$$y_k(\boldsymbol{x}_j) = \sum_{n_h}^{i=0} w_{ik} g_i(\boldsymbol{x}_j), \quad k = 1, 2, \ldots, n_c, \tag{14}$$

where $g_0(\boldsymbol{x}_j) = 1$. Given n_t feature vectors from n_c classes, training the RBFNN involves estimating $\boldsymbol{\mu}_i$, $i = 1, 2, \ldots, n_h$, η, d^2, and w_{ik}, $i = 0, 1, 2, \ldots, n_h$, $k = 1, 2, \ldots, n_c$. The training procedure is given below:

Determination of $\boldsymbol{\mu}_i$ and d^2: Conventionally, the unsupervised k-means clustering algorithm (Duda, Hart, & Stork, 2001), can be applied to find n_h clusters from n_t training vectors. However, the training vectors of a class may not fall into a single cluster. In order to obtain clusters only according to class, the k-means clustering may be used in a supervised manner. Training feature vectors belonging to the same class are clustered to n_h/n_c clusters using the k-means clustering algorithm. This is repeated for each class yielding n_h cluster for n_c classes. These cluster means are used as the centers $\boldsymbol{\mu}_i$ of the Gaussian activation functions in the RBFNN. The parameter d was then computed by finding the maximum distance between n_h cluster means.

Determining the weights $\mathbf{w}_{ik}$ between the hidden and output layer: Given that the Gaussian function centers and widths are computed from n_t training vectors (14) may be written in matrix form as

$$Y = GW, \tag{15}$$

where Y is a $n_t \times n_c$ matrix with elements $Y_{ij} = y_j(\boldsymbol{x}_i)$, G is a $n_t \times (n_h + 1)$ matrix with elements $G_{ij} = y_j(\boldsymbol{x}_i)$, and W is a $(n_h + 1) \times n_c$ matrix of unknown weights. W is obtained from the standard least squares solution as given by

$$W = (G^{\mathrm{T}} G)^{-1} G^{\mathrm{T}} Y. \tag{16}$$

To solve W from (16), G is completely specified by the clustering results, and the elements of Y are specified as

$$Y_{ij} = \begin{cases} 1 & \text{if } \boldsymbol{x}_i \in \mathit{class}\ j, \\ 0 & \text{otherwise}. \end{cases} \tag{17}$$

4. Experimental results

The evaluation of the proposed audio classification and segmentation algorithms have been performed by using a generic audio database which consists of the following contents: 100 clips of advertisement in different languages, 100 clips of songs (sung by male and female) in different languages, 100 cartoon clips, 100 clips of movie from different languages, 100 clips of sports and 100 clips of news (both Tamil and English). Audio samples are of different length, ranging from 1 s to about 10 s, with a sampling rate of 8 kHz and 16-bits per sample. The signal duration was slightly increased using the following rationale that the longer the audio signal analyzed, the better the extracted feature which exhibits more accurate audio characteristics. The training data should be sufficient to be statistically significant. The training data is segmented into fixed-length and overlapping frames (in our experiments we used 20 ms frames with 10 ms overlapping). When neighboring frames are overlapped, the temporal characteristics of the audio content can be taken into consideration in the training process. Due to radiation effects of the sound from lips, high-frequency components have relatively low amplitude, which will influence the capture of the features at the high end of the spectrum. One simple solution is to augment the energy of the high-frequency spectrum.This procedure can be implemented via a pre-emphasizing filter that is defined as

$$s'(n) = s(n) - 0.96 \times s(n-1), \quad n = 1, \ldots, N-1, \tag{18}$$

where $s(n)$ is the nth sample of the frame s and $s'(0) = s(0)$. Then the pre-emphasized frame is Hamming-windowed by

$$h(n) = 0.54 - 0.46 * \cos(2\pi n/N - 1), \quad 0 \leqslant n \leqslant N-1. \tag{19}$$

4.1. Preprocessing

The aim of preprocessing is to remove silence from a music sequence. Silence is defined as a segment of imperceptible audio, including unnoticeable noise and very short clicks. We use short-time energy to detect silence. The short-time energy function of a music signal is defined as

$$E_n = \frac{1}{N} \sum_m [x(m)w(n-m)]^2, \tag{20}$$

where $x(m)$ is the discrete time music signal, n is the time index of the short-time energy, and $w(m)$ is a rectangular window, i.e.,

$$w(n) = \begin{cases} 1 & \text{if } 0 \leqslant n \leqslant N-1, \\ 0 & \text{otherwise}. \end{cases} \tag{21}$$

If the short-time energy function is continuously lower than a certain set of thresholds (there may be durations in which the energy is higher than the threshold, but the durations should be short enough and far apart from each other), the segment is indexed as silence. Silence segments will be removed from the audio sequence. The processed audio sequence will be segmented into fixed-length and 10 ms overlapping frames.

4.2. Feature selection from non-silent frames

Feature selection is important for audio content analysis. The selected features should reflect the significant characteristics of different kinds of audio signals. The selected features include linear prediction coefficients (LPC)–linear prediction derived cepstrum coefficients (LPCC) and mel-frequency cepstrum coefficients(MFCC). LPC and LPCC are two linear prediction methods and they are highly correlated to each other. LPC-based algorithms (Abu-El-Quran, Goubran, & Chan, 2006), measure three values from the audio segment to be classified. These values are the change of the energy of the signal, speech duration, and the change of the pitch value. The audio signals are recorded for 60 s at 8000 samples per second and divided into frames of 20 ms, with a shift of 10 ms. A 14th order LP analysis is used to capture the properties of the signal spectrum as described in Section 2.1. The recursive relation (4) between the predictor coefficients and cepstral coefficients is used to convert the 14 LP coefficients into 19 cepstral coefficients. The LP coefficients for each frame is linearly weighted to form the WLPCC.

In order to evaluate the relative performance of the proposed work, we compared it with the well-known MFCC features. MFCCs are short-term spectral features as described in Section 2.2 and are widely used in the area of audio and speech processing. To obtain MFCCs (Umapathy et al., 2007), the audio signals were segmented and windowed into short frames of 256 samples. Magnitude spectrum was computed for each of these frames using fast Fourier transform (FFT) and converted into a set of mel scale filter bank outputs. Logarithm was applied to the filter bank outputs followed by discrete cosine transformation to obtain the MFCCs. For each audio signal we arrived at 39 features. This number, 39, is computed from the length of the parameterized static vector 13, plus the delta coefficients (+13) plus the acceleration coefficients (+13).

4.3. Modeling using SVM

We use a nonlinear support vector classifier to discriminate the various categories. Classification parameters are calculated using support vector machine learning. The training process analyzes audio training data to find an optimal way to classify audio frames into their respective classes. The training data should be sufficient to be statistically significant. The training data is segmented into fixed-length and overlapping frames. When neighboring frames are overlapped, the temporal characteristics of audio content can be taken into consideration in the training process. Features such as LPC, LPCC and MFCC are calculated for each frame. The support vector machine learning algorithm is applied to produce the classification parameters according to calculated features. The derived classification parameters are used to classify audio data. The audio content can be discriminated into the various categories in terms of the designed support vector classifier. The classification results for the different features are shown in Fig. 4. From the results, we observe that the overall classification accuracies is 92% using MFCC as feature.

4.4. Modeling using RBFNN

For RBFNN training, 14th order LPC, 19 dimensional LPCC and 26 dimensional MFCC features are extracted from the audio frames as described in Section 4.2. These features are given as input to the

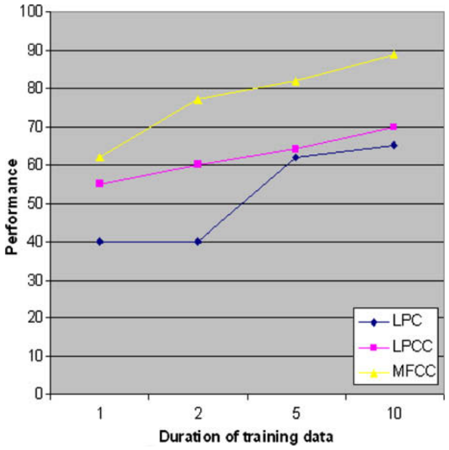

Fig. 4. Performance of SVM for audio classification.

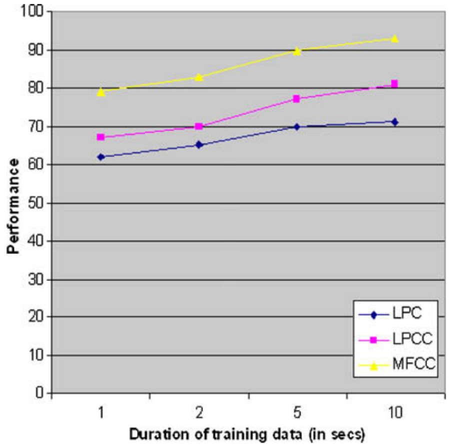

Fig. 5. Performance of RBFNN for audio classification.

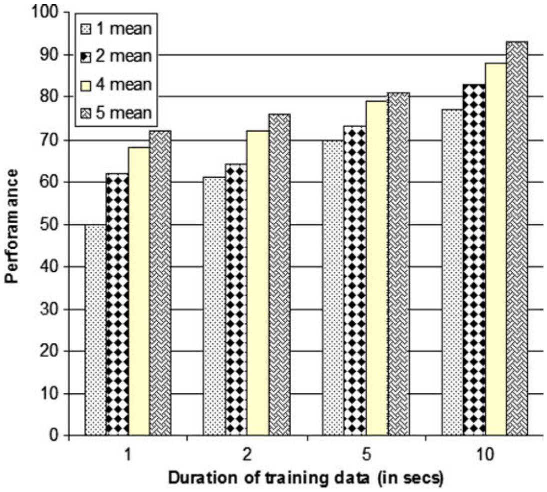

Fig. 6. Performance of RBFNN for different means.

Table 1
Audio classification performance using SVM and RBFNN

SVM (%)	RBFNN (%)
92.1	93.7

RBFNN model. The RBF centers are located using *k*-means algorithm. The weights are determined using least squares algorithm. The value of $k = 1$, 2, and 5 has been used in our studies for each category. The system gives optimal performance for $k = 5$. For training, the weight matrix is calculated using the least squares algorithm discussed in Section 3.2 for each of the features.

For classification the feature vectors are extracted and each of the feature vector is given as input to the RBFNN model. The average output is calculated for each of the output neuron. The class to which the audio sample belongs is decided based on the highest output. Fig. 5 shows the performance of RBFNN for audio classification. Fig. 6 shows the performance of RBFNN for different means.

The performance of the system for MFCC using SVM and RBFNN for audio classification is given in Table 1.

5. Conclusion

In this paper, we have proposed an automatic audio classification system using SVM and RBFNN. Linear prediction cepstrum coefficients (LPC, LPCC) and mel-frequency cepstral coefficients are calculated as features to characterize audio content. A nonlinear support vector machine learning algorithm is applied to obtain the optimal class boundary between the various classes namely music, news, sports, advertisement, cartoon and movie, by learning from training data. Experimental results show that the proposed audio classification scheme is very effective and the accuracy rate is 92%. The performance was compared to Radial basis function neural network which showed an accuracy of 93%. The classification rate using LPC and LPCC were slightly lower than the 39 dimensional MFCC feature vectors. This work indicates that support vector machines and radial basis function neural networks can be effectively used for audio classification. Eventhough by now some progress has been achieved, there are still remaining challenges and directions for further research, such as, extracting different features and developing better classification algorithms and integration of classifiers to reduce the classification errors.

References

Abu-El-Quran, A. R., Goubran, R. A., & Chan, A. D. C. (2006). Security monitoring using microphone arrays and audio classification. *IEEE Transactions on Instrumentation and Measurement, 55*(4), 1025–1032.

Ajmera, J., McCowan, I., & Bourlard, H. (2003). Speech/music segmentation using entropy and dynamism features in a HMM classification framework. *Speech Communication, 40*(3), 351–363.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. New York: John Wiley-Interscience.

Eronen, A. J., Peltonen, V. T., Tuomi, J. T., Klapuri, A. P., Fagerlund, S., Sorsa, T., et al. (2006). Audio-based context recognition. *IEEE Transactions on Audio, Speech and Language Processing, 14*(1), 321–329.

Esmaili, S., Krishnan, S., & Raahemifar, K. (2004). Content based audio classification and retrieval using joint time–frequency analysis. In *IEEE international conference on acoustics, speech and signal processing* (pp. 665–668).

Guo, G., & Li, S. Z. (2003). Content-based audio classification and retrieval by support vector machines. *IEEE Transactions on Neural Networks, 14*(1), 308–315.

Haykin, S. (2001). *Neural networks a comprehensive foundation*. Asia: Pearson Education.

Huang, R., & Hansen, J. H. L. (2006). Advances in unsupervised audio classification and segmentation for the broadcast news and NGSW corpora. *IEEE Transactions on Audio, Speech and Language Processing, 14*(3), 907–919.

Jiang, H., Bai, J., Zhang, S., & Xu, B. (2005). SVM-based audio scene classification. *Proceeding of the IEEE*, 131–136.

Kiranyaz, S., Qureshi, A. F., & Gabbouj, M. (2006). A generic audio classification and segmentation approach for multimedia indexing and retrieval. *IEEE Transactions on Audio, Speech and Language Processing, 14*(3), 1062–1081.

Li, S. Z. (2000). Content-based audio classification and retrieval using the nearest feature line method. *IEEE Transactions on Speech and Audio Processing, 8*(5), 619–625.

Lin, C.-C., Chen, S.-H., Truong, T.-K., & Chang, Y. (2005). Audio classification and categorization based on wavelets and support vector machine. *IEEE Transactions on Speech and Audio Processing, 13*(5), 644–651.

Li, D., Sethi, I. K., Dimitrova, N., & McGee, T. (2001). Classification of general audio data for content-based retrieval. *Pattern Recognition Letters, 22*, 533–544.

Lu, L., Zhang, H.-J., & Li, S. Z. (2003). Content-based audio classification and segmentation by using support vector machines. *Multimedia Systems, 8*, 482–492.

McConaghy, T., Leung, H., Boss, E., & Varadan, V. (2003). Classification of audio radar signals using radial basis function neural networks. *IEEE Transactions on Instrumentation and Measurement, 52*(6), 1771–1779.

Mubarak, O. M., Ambikairajah, E., & Epps, J. (2005). Analysis of an MFCC-based audio indexing system for efficient coding of multimedia sources. In *IEEE international conference on acoustics, speech and signal processing* (pp. 619–622).

Palanivel, S. (2004). Person authentication using speech, face and visual speech. Ph.D thesis, Madras: IIT.

Panagiotakis, C., & Tziritas, G. (2005). A speech/music discriminator based on rms and zero-crossings. *IEEE Transactions on Multimedia, 7*(1), 155–156.

Rabiner, L., & Juang, B. (2003). *Fundamentals of speech recognition*. Singapore: Pearson Education.

Rabiner, L., & Schafer, R. (2005). *Digital processing of speech signals*. Pearson Education.

Rajapakse, M., & Wyse, L. (2005). Generic audio classification using a hybrid model based on GMMS and HMMS. *Proceedings of the IEEE*, 1550–1555.

Umapathy, K., Krishnan, S., & Jimaa, S. (2005). Multigroup classification of audio signals using time–frequency parameters. *IEEE Transactions on Multimedia, 7*(2), 308–315.

Umapathy, K., Krishnan, S., & Rao, R. K. (2007). Audio signal feature extraction and classification using local discriminant bases. *IEEE Transactions on Audio, Speech and Language Processing, 15*(4), 1236–1246.

Vapnik, V. (1998). *Statistical learning theory*. New York: John Wiley and Sons.

Xu, C., Maddage, N. C., & Shao, X. (2005). Automatic music classification and summarization. *IEEE Transactions on Speech and Audio Processing, 13*(3), 441–450.