Audio Feature Extracton And Classification Using Wavelet Transform And Svm Tool

Mr. S.Ilaiyaraja,Professer
Nandhini .P, Infant Sneha .C, Sangamithira .I
Dept. of Electronics &Communication Engineering, Velammal Institute of Technology

Abstract *__Feature extraction and analysis are the foundation of audio classification. Here we propose an improved audio classification and categorizing method which makes use of wavelets and Support Vector Machines (SVM's).An audio signal is preprocessed using hamming window when a audio is given, wavelets are first applied to extract acoustical features such as sub-band power & pitch information. Also by using Fourier transform Bandwidth & Brightness of the audio features are extracted features are extracted. The proposed method uses SVM over these acoustical Features and additional parameters such as mean and median values of acoustical features.*

1,INTRODUCTION

Audio information often plays an essential role in understanding the semantic content of digital media. Recently audio information has been used for content-based multimedia indexing and retrieval. Audio semantic and visual semantic can complement each other, so audio feature extraction and classification are important research contents for content-based video analysis and retrieval. In the age of digital information, audio data has become part in many modern applications

A typical multimedia database often contains millions of audio clips, including environmental sounds and other nonspeech utterances. The need to automatically recognize to which class an audio sound belongs makes audio classification and categorization an emerging and important research area. The efficiency 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.Thus here audio signals are classified by analyzing its features. Over the past years audio classification and analysis is done using rule base method that include classification based on the NFL (nearest feature line), which draws a trajectory linking the corresponding features when the sound changes from one way to another. But while classifying sound clips, from muscle fish database it resulted in 40 classification errors.

Hence opting for a more accurate method for audio classification, several statistical techniques such as HMMs (Hidden Markova's Model) or support vector machine (SVM) can be used. Support vector machine is regarded as an efficient algorithm which can be used for audio categorizing. Here in this paper we improve audio classification by incorporating wavelet transform along with the SVM bottom up and top down approach. Wavelets are widely used technique that has also been applied to speech and audio feature extraction. In contrast with the conventional methods using the fourier transform, the subband power is extracted from the wavelet domain can improve the performance. To increase the discriminability of sounds, the overlap size between the windowed frames of the proposed feature extraction algorithm is redesigned, and a normalization process is carried out after feature extraction.

Finally, this paper proposes a bottom-up SVM categorization strategy that uses an iterative procedure to match a given audio data to progressively larger subsets of classes.

1.1 Audio feature extraction

Feature extraction is an important step for audio classification. Different features should be used in different methods for different applications. So in order to obtain high accuracy for audio classification and segmentation, it is critical to extract good features that can capture the temporal and spectral characteristics of audio signal. Before feature extraction, audio data is converted into a raw format, which is with 16 KHz, 16 bit, mono channel. Then it is pre-emphasized with the parameter 0.97 to equalize the inherent spectral tilt. The audio signal is divided into frames by using a Hamming window. Detailed preprocessing
and feature extraction are described as follows.

Preprocessing

As stated above, the original audio signal is sampled at 8000 Hz with 16 bit resolution. Each sound is divided into frames. The frame length is 256 samples (32 ms) with 192 sample (75%) overlap between adjacent frames. The procedure can be implemented via a pre-emphasizing filter that is defined as

$$s_n' = s_n - 0.96 \times s_{n-1}$$

for n = 1, 2, …., 255 (1)

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

$$s_i^h = s_i' * ⍰_i$$

$$for\ i = 0,1,\dots\dots.,255 \qquad (2)$$

with $⍰_i$ = 0.54 - 0.46 × *cos(2πi /255).* The pre-processed frame will be detected as a nonsilent frame for feature extraction if the total power is large, i.e.,

$$\sum_{i=0}^{255} (s_i^h)^2 > 400^2 \qquad (3)$$

where 400 is an experience value.

Three level wavelet transform

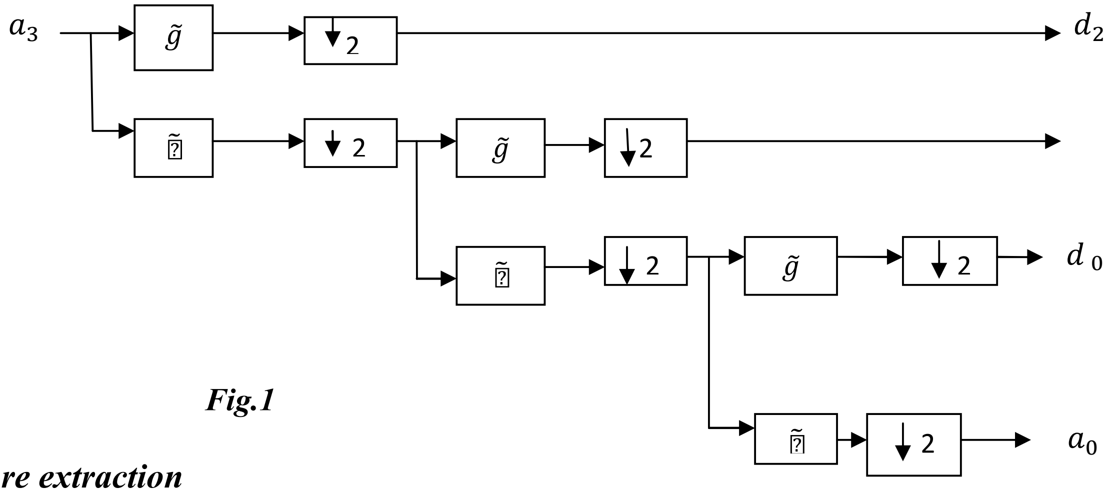


Fig.1

Feature extraction

The fourier transform is the most popular method that's maps audio signals from time domain to frequency domain. The wavelet transform is another choice for this feature extraction, especially a three level wavelet transform gives better performance for an audio signal with a sampling rate of 8000 Hz. Hence this paper applies both fourier and wavelet transforms to increase the ability to capture proper audio features.

1.2 Wavelet Transform – Brief introduction:

The wavelet transform discussed here is implemented via a filter bank structure. A fast discrete algorithm is shown in the below fig.1,

where ⍰̃(n) and $\tilde{g}$(n) are the analysis lowpass and highpass filters. The symbol 2
denotes down sampling by 2. ↓

d_0

d_1

d_2

a_0

Let $\{a_3(n)\}_{n \in Z}$ be the input to the analysis filter bank. Then the outputs of the analysis filter bank are given by

$$a_i(k) = \sum_n \tilde{h}(n - 2k)a_{i+1}(n) \qquad (4)$$

and $$d_i(k) = \sum_n \tilde{g}(n - 2k)a_{i+1}(n) \qquad (5)$$

where $a_i(k)$ and $d_i(k)$ are called the approximation and detail coefficients of the wavelet decomposition of a_{i+1}(n), respectively. The list of extracted features which uses both fourier and wavelet transforms are given in the following Table I

Table I: List of extracted features

Feature	Type of Transform	No. of features
Sub-band power P_j	Wavelet	3
Brightness ω_c	Fourier	1
Bandwidth B	Fourier	1

The detailed extraction process of each feature is given in the following

1. **Sub-band power P_j:** Three sections of sub-band power calculated in the wavelet domain are used in this paper. Let *ω* be the half sampling frequency. Then the sub-band intervals are [0, *ω/8], [ω/8, ω/4], and [ω/4, ω/2],* corresponding to the approximation and detail coefficients $a_0(k)$, $d_0(k)$, $d_1(k)$ of a given audio signal $a_3(k)$ respectively. The sub-band power is calculated by

$$P_j = \sum_k z_j^2$$

where z_j(k) is the corresponding approximation or detail coefficients of sub-band *j.*

2. **Brightness ω_c:** The brightness is the frequency centroid of fourier transform and is computed as

$$\omega_c = \int_0^{\omega} u|F(u)|^2 du \,/\, \int_0^{\omega} |F(u)|^2 du$$

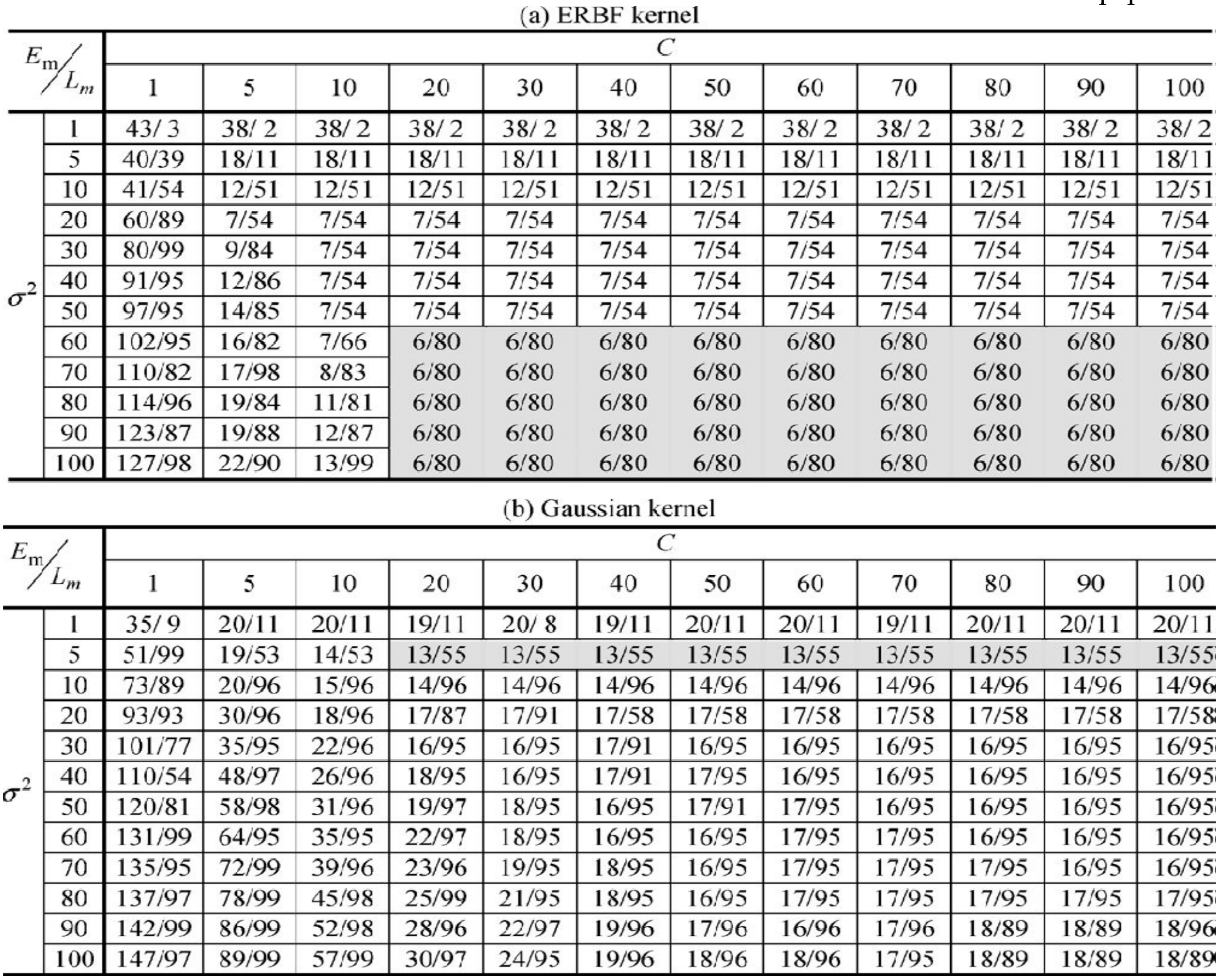
3. **Bandwidth *B:*** It is the square root of the power- weighted average of the square difference between the spectral components and the frequency centroid.

$$B\ is: \sqrt{\int_0^{\omega} (u - \omega_c)^2 |F(u)|^2 du \,/\, \int_0^{\omega} |F(u)|^2 du}$$

1.3 Audio classification:

SVM –Classifier : Support Vector Machine(SVM) is used as a binary classifier. SVM learns an optimal separating hyper plane from a given set of position and negative examples. It is based on the principle of structural risk minimization. SVM can be either linear or non-linear (kernel based). The former is used in linearly separable case, and the latter is used in linearly non-separable but nonlinearly separable case. The reason is that SVM is much more effective than other conventional classifiers in terms of classification accuracy, computational time, and stability

to parameter setting. They also prove to be more effective than the traditional pattern recognition approaches based on the combination of a feature selection procedure and a conventional classifier. SVMs use a known kernel function to define a hyperplane in order to separate given points into two predefined classes. An improved SVM called the soft-margin SVM can tolerate minor misclassifications. It is considered to be more suitable for classification and, therefore, is sed in this paper.



(a) ERBF kernel

E_{m}/L_m		C											
		1	5	10	20	30	40	50	60	70	80	90	100
σ^2	1	43/ 3	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2	38/ 2
	5	40/39	18/11	18/11	18/11	18/11	18/11	18/11	18/11	18/11	18/11	18/11	18/11
	10	41/54	12/51	12/51	12/51	12/51	12/51	12/51	12/51	12/51	12/51	12/51	12/51
	20	60/89	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54
	30	80/99	9/84	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54
	40	91/95	12/86	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54
	50	97/95	14/85	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54	7/54
	60	102/95	16/82	7/66	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80
	70	110/82	17/98	8/83	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80
	80	114/96	19/84	11/81	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80
	90	123/87	19/88	12/87	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80
	100	127/98	22/90	13/99	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80	6/80

(b) Gaussian kernel

E_{m}/L_m		C											
		1	5	10	20	30	40	50	60	70	80	90	100
σ^2	1	35/ 9	20/11	20/11	19/11	20/ 8	19/11	20/11	20/11	19/11	20/11	20/11	20/11
	5	51/99	19/53	14/53	13/55	13/55	13/55	13/55	13/55	13/55	13/55	13/55	13/55
	10	73/89	20/96	15/96	14/96	14/96	14/96	14/96	14/96	14/96	14/96	14/96	14/96
	20	93/93	30/96	18/96	17/87	17/91	17/58	17/58	17/58	17/58	17/58	17/58	17/58
	30	101/77	35/95	22/96	16/95	16/95	17/91	16/95	16/95	16/95	16/95	16/95	16/95
	40	110/54	48/97	26/96	18/95	16/95	17/91	17/95	16/95	16/95	16/95	16/95	16/95
	50	120/81	58/98	31/96	19/97	18/95	16/95	17/91	17/95	16/95	16/95	16/95	16/95
	60	131/99	64/95	35/95	22/97	18/95	16/95	16/95	17/95	17/95	16/95	16/95	16/95
	70	135/95	72/99	39/96	23/96	19/95	18/95	16/95	17/95	17/95	17/95	16/95	16/95
	80	137/97	78/99	45/98	25/99	21/95	18/95	16/95	17/95	17/95	17/95	17/95	17/95
	90	142/99	86/99	52/98	28/96	22/97	19/96	17/96	16/96	17/96	18/89	18/89	18/96
	100	147/97	89/99	57/99	30/97	24/95	19/96	18/96	18/96	17/95	18/89	18/89	18/89

The mean C and variance σ^2 values for the above extracted features like sub-band power, brightness and bandwidth. The above figure shows the experimental results of (a) ERBF (b) Gaussian kernel functions for the preselected values C and σ^2 , where E_m is computed as the least value of errors and L_m indicates the level L at which the first E_m happens.

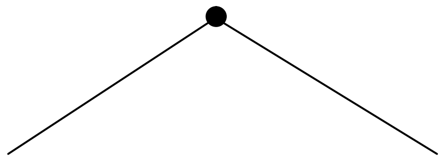
There are three common kernel functions for the nonlinear feature mapping (1)ERBF , 2) Gaussian function , where parameter is the variance of the Gaussian function, and 3) polynomial function, where parameter is the degree of the polynomial. Many lassification problems are always separable in feature space and are able to obtain better accuracy by using the Gaussian kernel function than the linear and polynomial kernel functions.

Multicase classification in SVM

A typical SVM is a two-class classifier that organizes all training sets into two classes, namely, plus-class (+ 1) and minus-class (-1). It has to be augmented with other strategies to achieve multicase classification. Four commonly used schemes are given below.

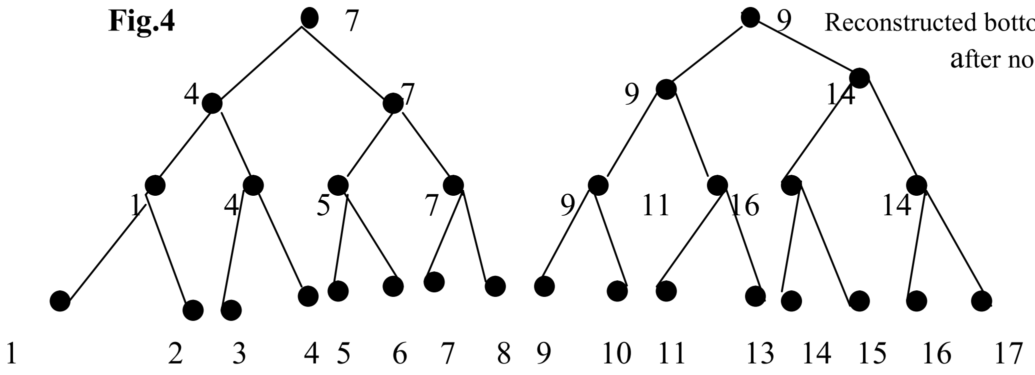
1) One-against-one: Classify between each pair of classes.
2) One-against-all: Classify between each class and all other remaining classes.
3) Top-down binary tree: An initial group contains all classes. A recursive process is done to separate and reduce a larger candidate group of classes into a smaller one until the test pattern is assigned to a final class.
4) Bottom-up binary tree: A recursive comparison process is performed between pairs of classes. The class with a shorter distance from the test pattern is retained for further comparison until the test pattern is assigned to a final class.

1.4 Audio Categorization of an SVM



A typical SVM with a bottom-up binary tree scheme only achieves multicase classification, in which a single class is selected as the class to which a given audio data belongs. Depending on applications, it is sometimes more suitable to select more than one class as the possible candidates. A method based on this tree scheme can be carried out to rank classes with respect to a given audio by proceeding iteratively. In each round, it removes the winning class number from the root of the tree and then reconstructs a new tree structure. The class removed first is the class to which the given audio is most similar. Conversely, the class number removed last belongs to the class to which the given audio is least similar. Fig. 4 illustrates the reconstructed bottom-up binary tree scheme after node 12 was first removed.

Fig.4



Reconstructed bottom up binary tree after node 12 was removed

The collected numbers determine an ordering of similarities between classes and the tested audio data. The first N classes in the ordering as a group are referred to as the Top N group.

CONCLUSION:

Thus from the above we can see that our proposed method reduces the size of the feature set using the wavelet transform instead of the Fourier transform. Even though the conventional Fourier transform is suitable for audio samples with concentrated energy, the wavelet transform is more natural and effective for describing audio characteristics. A great improvement in classification accuracy is achieved from 91.9% to 97.0% in the Top 1. Furthermore, each misclassified sound can be classified 100% correctly into its own class in the Top 2 at most , upper bound , and variance σ^2 settings. Although the Gaussian kernel is widely used to carry out classification, the ERBF kernel turns out to be more suitable for the Muscle Fish audio database in our experiments.

REFERENCE:

[1] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search and retrieval of audio," *IEEE Multimedia*, vol. 3, no. 3, pp. 27–36, Jul. 1996.

[2] S. Z. Li, "Content-based audio classification and retrieval using the nearest feature line method," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 619–625, Sep. 2000.

[3] T. Zhang and C.-C. J. Kuo, "Audio content analysis for online audiovisual data segmentation and classification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 4, pp. 441–457, May 2001.

[4] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 7, pp. 504–516, Oct. 2002.

[5] G. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," *IEEE Trans. Neural Networks*, vol. 14, no. 1, pp. 209–215, Jan. 2003.

[6] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[7] P. Clarkson and P. J. Moreno, "On the use of support vector machines for phonetic classification," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, vol. 2, Mar. 1999, pp. 585–588.

[8] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[9] V. Kecman, *Learning and Soft Computing*. Cambridge, MA: MIT Press, 2001.

[10] P. Ding, Z. Chen, Y. Liu, and B. Xu, “Asymmetrical support vector machines and applications in speech processing,” in *Proc. IEEE Int. Conf Acoustics, Speech, Signal Process.*, vol. 1, May 2002, pp. I-73–I-76.