ICICS-PCM 2003
15-18 December 2003
Singapore

Audio Events Classification Using Hierarchical Structure

W. Huang*; S. Lau+; T. Tan*, L. Li* and L. Wyse*

*Institute for Infocomm Research, Singapore
+Department of Electrical & Computer Engineering, National University of Singapore
{wmhuang, teletan, lyli, lonce}@i2r.a-star.edu.sg
lsl21@singtel.com.sg

Abstract

This paper presents a novel approach using hierarchical structure with different feature sets to classify the audio signals efficiently into audio events. Most of the past researches focused on the speech (male, female), music (different genre) and environment sound (noise). To further differentiate the environment sound, this work studies the feature selection and classification. Different from that of other methods, the audio signals in our work are segmented into different length perceptually. So human can also recognize a sound based on the short segments. A top-down tree structure with the selected features is designed to classify the audio events while each node is a support vector machine trained as a classifier. Experiments show the robustness and efficiency of the method with a small set of training database.

1. Introduction

Audio signals together with video play important role to tell people what is happening in the scene. Sometimes audio signals can provide more accurate information than video signals and sometimes audio information may be the only clue for an event. For example a person screaming can tell that something unusual happens while from video signal it may be hard to recognize any abnormal activity.

One study related to the audio event classification is on the auditory scene analysis that is to recognize an environment using audio information only [1,2]. However, its focus is to recognize the context environment instead of the audio events, in which we are more interested.

Besides the effort on auditory scene recognition, more research works are focusing on the efficient indexing and retrieval of audio data due to the large amount of music, speech and other sound clips available today for human to browse [3,4,5]. In the same time audio classification has also attracted much interest from researchers for speech and music classification, musical genre classification and some other sounds [6,5,7].

One way to classify sounds into multiple classes is based on the nearest neighbor method, where a distance measure must be obtained for two samples. However when there are no enough training samples, the KNN method can't generate good result compared to the hierarchical classification [8].

To form a hierarchical tree structure classification, two approaches can be used. One is the top-down (TD) approach and the other is bottom-up (BU) approach. To compare, those two methods have different advantages. The TD approach can provide some intermediate information and need less storage for the classifiers and BU approach is generally faster but need to store more classifiers [5,8], which are further discussed in Section 4.

In this work we studied the audio event classification using support vector machine (SVM) [10], which is proved to be a robust classifier based on examples, and the top-down hierarchical structure for classification of 7 audio events, i.e., *Screaming*, *Crying*, speech (*Male*), speech (*Female*), *Laughing*, *Knocking*, *Explosion*. Later the first capital of the name will be used for each class. With SVM, we deployed the efficiency of using different audio features for audio event classification. Further a TD hierarchical structure is generated based on the binary SVM results. Figure 1 shows the flowchart of the system.

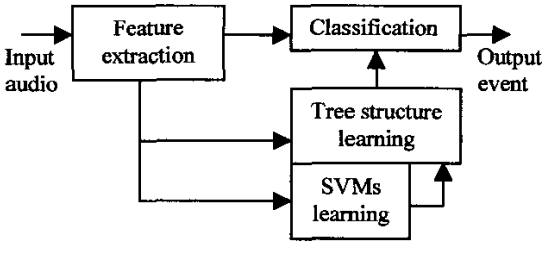

Fig 1 Audio event classification

The paper is arranged as follows. Section 2 reviews the feature extraction for audio signals. Section 3 briefs the SVM for multi-class classification. Section 4 describes the tree structure formation for multiple audio event classification. Section 5 gives the experimental results and conclusion.

2. Audio Features

The objective of feature extraction is to transform the original acoustic wave into more representative and

0-7803-8185-8/03/$17.00 © 2003 IEEE

distinguishable data for comparison. For our purpose, here we only consider the features extracted from short audio segments.

Before the features are extracted the signal is divided into short time frames of 25ms, with a 50% overlapping between the neighboring frames. For each category of audio events, a fixed length of the signal is used to extract the whole feature vector, for example we use 1.5s segment for the feature extraction of explosion. To get the consistent feature, all the audio signals are sampled at 11kHz.

In this work we studied the features both in time domain and frequency domain [11] for the SVM classification. However we found that only the Mel-frequency cepstral coefficients (MFCC) and the derivative are more useful for the event classification.

The MFCC is so called Mel-scaled frequency cepstral coefficient, which represents the human perception of the frequency of sound. The subjective pitch of a sound is measured on a *Mel* scale, emphasizing the mid-frequency bands in proportion to their perceptual importance [11]. First a Hamming window is used to get a *frame* of the audio, followed by a discrete cosine transform (DCT). The log of the power spectrum of the DCT is scaled by weighting with a Mel-scale. Then a further DCT is used to transform (reduce) the Mel-weighted spectrum into the low dimensional feature vector.

The above process can generate a 13-dimension vector for each frame comprising of 12 MFCCs with a normalized log energy component. The log-energy part is calculated by taking the log of the sum of the squared data samples. Two sets of features are implemented using MFCCS. First a 13-dimension feature vector of MFCCs and second, a 36-dimension feature vector, including the 12 MFCCs and its first and second derivatives (Delta-MFCCs). Other audio features such as zero-crossing-rate (ZCR) and fundamental frequency can be found in [11].

Usually there are many frames in an audio segment, which is to be classified into an event. In the above calculation, a 2s segment of audio has 157 frames, 13 coefficients which totals to a feature size of 2041 while for delta MFCCs with 36 coefficients, 157 frames will total to a feature size of 5652 coefficients. So the total features of the audio segment are still too many for later use to train the classifier. Here, *Principal Component Analysis (PCA)* is adopted to reduce dimensionality and noise on these high dimensional features.

3. Multi-Class SVMs

Originally a support vector machine is designed for binary classification [15]. A discriminant function f(x) is learned from the two-class examples, so that $f(x_i)y_i>0$, where $y_i \in \{1,-1\}$ is the label of x_i, $y_i=1$ means a positive example, -1 means a negative example.

In order to cope with multi-class issue, many methods were proposed in the literature to transform the multi-class problem into a series of binary-class problem [12]. Typically two approaches are stated as one-against-all SVM and one-against-one SVM in the training phase. The former constructs a SVM for each class, where all the samples in this class are labeled as positive example and all samples in other classes are labeled as negative example.

Then a critical issue is how to decide the final classification result with all the SVMs. One method is by voting strategy, where multiple votes from SVMs are counted for each class after the outputs f(x) of all of the SVMs are calculated. The highest vote for a class indicates that it is the best. Another way to use the class with the largest value of the decision function f(x) as the final result. However, as pointed in [13], these methods may have the problem of inconsistent output. There exist possible contradictory voting and the value of the function f(x) for different SVMs are also in different scale. It is sometimes hard to classify a sample.

Another approach to the multi-class decision is to form a decision tree, such as the DAGSVM [14] and bottom-up tree [9], which are fast for the multi-class classification. To classify a sample in the case of N-class, only N-1 comparisons are needed. However it has the same inconsistency problem like the one-against-one SVM, the result depends on the order of the pair to be compared for some samples.

In this paper we consider to construct a top-down decision tree for the multi-class classification that will form the structure by learning from the examples.

4. Decision Tree Construction

One way to construct a decision tree is based on the physical meaning behind the classes. In the music genre classification [5], it is natural to classify speech and music first. Then in the music class, it is divided into Classical, Country, Jaz, HipHop etc. In genre Classical, the audio is further divided into different instruments, such as Choir, Piona, Orchestra etc. based on the musical knowledge.

In some cases however it is not clear which classes should be clustered together, for example in the application of audio surveillance, where there is little domain knowledge available due to it is a new research area or lack of the expertise. Randomly clustering may not provide a good interpretation for the tree structure. Considering this, we develop a learning process here for the automatic tree construction.

Given a N-class problem, and N sets of labeled training samples $\{X_1, X_2, \ldots, X_N\}$, a top-down decision tree can be learned by a series of binary SVMs. We start from the one-against-all SVMs. So totally N SVMs can be constructed based on a feature vector, such as the ZCR, MFCC or delta-MFCC. To balance the training samples for each SVM, we limit the number of negative samples, so the two classes have almost the same number of the training samples.

With a testing data set, we can evaluate the SVMs performance and adjust the partition of the data with the rules to partition the samples with minimum errors. Assume for a SVM, a testing set with $M = M_1+M_2+\ldots+M_N$ samples for the N classes, M_i is the testing samples in class i. We can have the probability measure for the classification: $P_k = \{P^+, P^-\}$. $P^+=\{p_k\}$, $P^-=\{p_i,\ i=1,2,\ldots,N, i\neq k\}$, where $p_i = M_i' / M_i$. M_i' is the number of samples in class i classified correctly. Usually for a one-against-all SVM, assuming that the k-th class is the positive class initially, we should have at least $p_k>0.5$. For all $i\neq k$, if $1-p_i > \alpha$, a small positive value, which means that more than α percent of this negative class are classified as positive class, we will put all these classes together to the positive class k to train the SVM using the training set again. The error rate for this SVM(k) is then

$$\pi_k = 1-\Sigma \{p_i \cdot (M_i/\Sigma M_j)\} \quad (1)$$

Starting from N one-against-all SVMs and iterating the above process if $\pi_k(t)<\pi_k(t-1)$ until $\pi_k(t-1)-\pi_k(t)<\beta$ or $t>N_t$, it will end to N new SVMs using the above process. So the first SVM on the root of the tree to partition the N-class is the r-th SVM that

$$r = \text{argmin}_k\{\pi_k\} \quad (2)$$

Now we have a SVM that partition all the data into two classes. Within each class, the same process can be repeated to find the best SVM to partition the data further until each class contains only one audio event. A decision tree is built now for the multi-class SVM classification.

Up to now we did not consider the audio features' impact for the classification. To evaluate the impact of the features, we tested each SVM from the root using different features, which results in SVMs of possible different partitions with that feature vectors. The error rates of all the SVM at the same node in the tree are compared, the feature f(i) and the SVM(k_f) are selected so that

$$SVM(k_f) = \min_{f(i)} SVM(k_{f(i)}) \quad (3)$$

The final hierarchical structure is a SVM tree, where at each node we try to find a SVM that can partition the data with minimum error.

5. Experiments

We have collected the sound clips from different sources, such as web sites, movie and audio. Totally 7 categories, 296 clips are collected for the experiments. To simply the data, we selected those data that human can distinguish. The tree structure, the classification confusion matrix and the accuracy are reported here.

5.1 Data preparation

To classify data using SVM, the data feature should be extracted in the same size. However in the audio event classification, an audio event may last only a very short time, from which we hope the system is also viable. Here we segment the clip into 1 to 2s segments based on that the category of the segment can still be distinctively recognized by human. The list below shows the audio segment length for each class.

class	E	K	L	S	C	M	F
Length(s)	1.5	1	2	2	1.5	2	2

Since the minimum lengths are different, we need to calculate features based on the segments of different lengths for each SVM. For example, to distinguish a Scream from others, all audio segments cropped from the original clips are 2s. The 1s length segment is used for Knock detection. Most of computation is used in the calculation of MFCC, for which the window size is fixed. So the calculation of the whole feature vector is still in real time.

To train SVM, both positive and negative samples are needed. To balance these two types of samples, we keep numbers of them in the same range by randomly add or delete from the training set. The testing samples are extracted from testing clips which are different with those clips in the training set. To validate the generalization of the method, we will use the leave-ten-out approach to cross validate the result.

5.2 Decision tree for classification

For space reason, the result using other audio features are omitted since they are not so effective for the audio classification. Here only the MFCC and Delta-MFCC are discussed.

The initial one-against-all SVM result is shown in Table 1. The left column is the SVMs trained using training data. The figure in the table is the percentage of the testing samples of a class being classified as the positive class. With the example of Explosion SVM, where 70% of Explosion testing samples are classified correctly as Explosion (positive), non Laughing (0%) samples are classified as Explosion correctly and 35% Male speech testing samples are classified wrongly as positive samples, i.e. Explosion.

Table 1. Testing result of the one-against-all SVMs

Delta-MFCC SVM	Testing SVM classification (%)						
	E	*L*	*K*	*C*	*S*	*M*	*F*
E	70	0	95	10	0	35	0
L	0	90	5	95	85	15	0
K	0	95	20	30	100	15	45
C	10	70	10	85	15	25	10
S	0	5	0	5	100	0	0
M	100	0	100	0	0	75	80
F	100	0	100	0	0	35	100

The new result growing from the initial one-against-all SVMs using the proposed decision tree is shown in Table 2, but with different features.

Table 2. Testing result of the SVMs after tuning by the Decision tree. The "(d)" following the audio event indicates the audio feature is Delta-MFCC. Otherwise the feature is MFCC.

SVM	Testing SVM classification (%)						
	E	*L*	*K*	*C*	*S*	*M*	*F*
S(d)	0	5	0	5	100	0	0
K	100	0	100	95	-	0	0
E	100	-	5	100	-	-	-
C (d)	10	-	-	90	-	-	-
L(d)	-	100	-	-	-	15	0
M(d)	-	-	-	-	-	100	0

The decision tree associated with the result from Table 2 is shown in Fig 2. At the root a SVM trained by using only the Scream clips as positive samples and all the others are the negative sample, with Delta-MFCC as the feature vector. At the second level, Knock, Explosion and Crying are distinguished from Laughing and Male and Female speech with MFCC feature vector. The figure explains the other branches in the same way.

5.3 Results

To test the tree classification using SVM, a testing set different from those used for training the SVM and tuning the tree structure are used. We randomly take 20 (in fact 3x20 segments considering the varying length) audio segments from each class for the testing. Totally we have 140 segments for the test. Table 3 shows the confusion matrix using the decision tree of Fig 2. From the table the accuracy of the tree is 132/140=94.29%. Further a leave-ten-out testing shows an accuracy of 92.14% (387/420) using the proposed method.

6. Analysis and Conclusion

In this paper a new approach is proposed to learn a decision tree for audio event recognition and the preliminary result is reported here. The decision tree is constructed by refining the SVMs using different group of audio, feature set, and with different audio length.

One related work uses the confusion matrix to obtain a two stage hierarchical multi-class classification [16], which is built from a naïve Bayesian classifier. Different from that, we did not try to use the confusion matrix directly but proposed a multi-stage classifier by taking care the merge of classes at each node in a decision tree. So there is no voting or maximum-win in later decision stage.

Table 3. Testing result of the hierarchical tree classifier

Audio Class	Audio segments for testing						
	E	*L*	*K*	*C*	*S*	*M*	*F*
E	20	0	1	0	0	0	0
L	0	19	0	0	0	3	0
K	0	0	19	1	0	0	0
C	0	0	0	17	0	0	0
S	0	1	0	2	20	0	0
M	0	0	0	0	0	17	0
F	0	0	0	0	0	0	20

With the tree structure we can see that Screaming is the most distinguishable event from others. It can be classified at the root node. After that, the events of Explosion, Knocking and Crying are grouped together in the tree. We

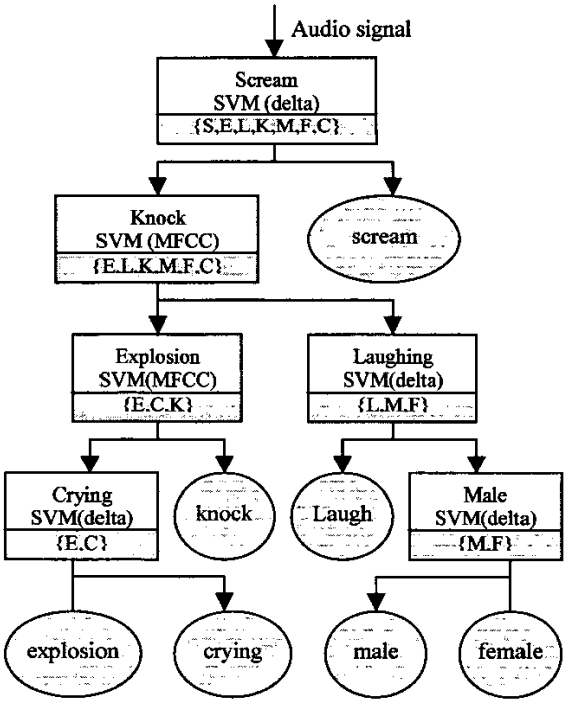

Figure 2. Top-down decision tree structure for audio event classification

also observed that the speech of Male and Female are grouped together in the tree, as well as Laughing. By hearing those clips, we find that there are some similarities among them, which is revealed by these intermediate tree node.

Different with other methods, here we use different features at each node in order to obtain the optimal classification at each level. One interesting observation is that the Explosion and Crying are both perceptually distinguishable at 1.5s and they are grouped together to the end of the tree. Similar finding is on the speech signals (Male and Female), shown in Fig 2.

Due to the lack of enough training data, the initial multi-class SVM method with the *maximum-win* could only achieve an accuracy of 81.43%. With the hierarchical SVMs, the performance is improved greatly, i.e. 94.29%. The current testing is based on the audio clips obtained from the internet or extracted from movie. The quality is quite good. For real application the signal-noise-ratio is usually much lower than the data we have. How to improve the performance under such a condition is our future research work.

In this paper we report a new adaptive SVM-based decision tree approach to the multi-class audio event classification. It is applied to the audio event classification for surveillance. The future work is to test it against other data sets to test its performance and compare it with other methods used for multi-class classification. In this paper we did not go through the automatic segmentation process of audio signals. The excessive silence clips were manually edited. A more challenging work is to develop an audio event detection system from continuos audio input with other unknown audio classes.

References

[1] V. T. K. Peltonen, A. J. Eronen, M. P. Parviainen and A. P. Klapuri, "Recognition of Everyday Auditory Scenes: Potentials, Latencies and Cues," The AES 110th Convention, Amsterdam, The Netherlands, May 2001.

[2] A. S. Bregman, Auditory Scene Analysis: The Perceptual Organization of Sound, Cambridge, MIT Press, 1990.

[3] J. Foote. "Content-based retrieval of music and audio," In C. C. J. Kuo et al., editors, Multimedia Storage and Archiving Systems II, Proc. SPIE, volume 3229, pp.138-147, 1997.

[4] C. Yang, "MACS: Music Audio Characteristic Sequence Indexing for Similarity Retrieval," IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New York, 2001.

[5] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signals," IEEE Trans on Speech and Audio Processing, Vol. 10, No. 5, pp.293-302, July 2002.

[6] A Bugatti and A Flammini, "Audio Classification in Speech and Music: A comparison Between a Statistical and a Neural Approach," EURASIP Journal on Applied Signal Processing, No 4, pp.372-378, April, 2002.

[7] L. Lu, S. Z. Li and H.-J. Zhang, "Content-Based Audio Segmentation Using Support Vector Machine," IEEE ICME 2001, Tokyo, Japan, Aug, 2001.

[8] P. Herrera, A. Yeterian, and F. Gouyon, "Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques," Proceedings of Second International Conference on Music and Artificial Intelligence, Edinburgh, Scotland, 2002.

[9] G. Guo and S. Z. Li, Content-Based Audio Classification and Retrieval by Support Vector Machine," IEEE Trans on Neural Networks, Vol 14, No 1, pp.209-215, Jan 2003.

[10] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization", in Advances in Kernel Methods - Support Vector Learning, (Eds) B. Scholkopf, C. Burges, and A. J. Smola, MIT Press, Cambridge, Massachusetts, chapter 12, pp 185-208, 1999.

[11] L. R. Rabiner and B. H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[12] C.-W. Hsu and C.-J. Lin, "A Cpmparison of Methods fro Multi-class Support Vector Machines," IEEE Trans on Neural Networks, Vol 13, No. 2, pp.415-425, March 2001.

[13] D. M. J. Tax and R. P. W. Duin, "Using two-class classifiers for multiclass classification," Proc of Int Conf on Pattern Recognition, 2002.

[14] J. C. Platt, N. Cristianini, and J. Shawe-ayor," Large margin DAGs for multiclass classification," in Advances in Neural Information Processing Systems, Vol. 12, pp.547-553, MIT Press, 2000.

[15] V. N. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.

[16] S. Godbole, S. Sarawagi and S. Chakrabarti, "Scaling multi-class Support Vector Machines using inter-class confusion," Proc of 8th ACM SIGKDD, pp.513-518, 2002.