EUSIPCO 2013 1569744475

EVENT DETECTION IN SHORT DURATION AUDIO USING GAUSSIAN MIXTURE MODEL AND RANDOM FOREST CLASSIFIER

Anurag Kumar[1], Rajesh M Hegde[1], Rita Singh[2], Bhiksha Raj[2]

1. Indian Institute of Technology Kanpur, India
2. Carnegie Mellon University, Pittsburgh, USA

ABSTRACT

The amount of online multimedia files is increasing day by day with the ever increasing popularity of video sharing websites. This has led to a huge interest in content analysis of multimedia files. Audio being a major component of multimedia has the potential to help analyze different events occurring in a multimedia recording. In this paper we present an audio event detection mechanism based on Gaussian Mixture Model (GMM) and Random Forest Classifier. Experiments show that our proposed mechanism shows significant improvement in detection of specifically finer audio events in short duration recordings.

Index Terms— Multimedia Events, Gaussian Mixture, Clustering, Random Forest Classifier

1. INTRODUCTION

The popularity of videos sharing websites has led to a huge collection of online multimedia data. This requires some intelligent mechanism to analyze the content of these multimedia recordings, to aid in cataloging, indexing and retrieval of these data. Multimedia recordings include both audio and video, but the audio part can in itself provide sufficient evidence to detect many events in them. For example, events like gunshots, crowd noise, children's voices, etc. can be characterized using only the audio component of the recordings. Techniques for automatic detection of such *audio events* will enable improved analysis of the recordings. Detection of audio events also finds applications in audio-based surveillance systems. As a result, there has been a significant amount of research devoted lately to audio event detection.

The conditions under which videos available online are recorded are unconstrained, and it is difficult to make any assumptions regarding the state of the surroundings or the recording conditions themselves. This makes the automatic analysis of multimedia recordings tricky. Also, sounds from multiple sources or phenomena often occur concurrently, and this makes the event detection using audios more complex. As a result, robust representations of the audio are required that will permit classification or detection of the events even when recording situations are unconstrained.

In this paper we propose a robust feature representation for detection of *fine* audio events based on characterization of data distributions through Gaussian Mixture Models (GMMs). By *fine* audio events we mean events which have unique, identifiable characteristics, such as clanking sounds, clapping, children's voices etc. whereas *broad* event categories are like Birthday Party, Wedding Ceremony, Football Stadium etc. which are characterized by patterns of occurrence of finer events.

Detection of events in generic multimedia has been interest of several authors. In [1] markov-model based clustering has been used for concept detection. An SVM based method has been proposed in [2]. The authors of [3] use clustering and vector quantization to generate a bag-of-audio-words representation to characterize audio and detect events. Bag of audio words representations have also been used as a part of multimodal approaches to event detection in multimedia in [4], [5] and [6]. An alternative method is proposed in [7] which models classes as Gaussians and employs probabilistic latent semantic analysis of Gaussian component histograms on soundtracks of videos to identify types of videos. In [8] a speech recognition framework using HMMs is used for detection of events. Detection of specific events such as Gunshots using GMM-based classifiers [9] and using Bayesian networks [10] have been employed in surveillance systems.

Possibly the most successful approach in all of this is the bag-of-audio-words representation, due to its simplicity and considerable success in detecting audio events. In a slightly different context it has even been used for copy detection in audio [11]. The bag-of-audio-words (BoAW) method involves generating "words" with a clustering algorithm, quantizing the original features to generate the "bag-of-words" in the form of a histogram, which is used as the feature to represent the audio recording for classification [3]. This characterization is particularly effective when capturing relatively long-term characteristics of sounds. Capturing fine audio events, which only last for short intervals, and that too in short duration clips which have been recorded in natural surroundings is a considerably tougher task, and such representations can often be too ``noisy'', in that the short duration of the events can result in large cross-instance variations in the features.

In the following sections we first try to explain different problems associated with detection of short duration fine audio event categories using the bag-of-audio-words approach and then suggest ways to address them. Section 2 describes the problem under consideration. Section 3 describes our approach to solve the problems. Section 4 describes our experiments and the results. In Sections 5 we discuss our conclusions and future work.

2. AUDIO BASED MULTIMEDIA EVENT DETECTION

A considerable number of research efforts on multimedia event detection are centered on the TRECVID Multimedia Event Detection Track. The TRECVID 2011 corpus [12] contains a total of 15 broad event categories such as attempting a board trick, landing a fish, birthday party, wedding ceremony, flash mob gathering etc. Since audio information in a multimedia is critical audio based event detection mechanisms too are important. In [3] the first five events of TRECVID 2011 corpus have been used for the performance evaluation of the developed method. As stated earlier they have used Bag of Audio Words mechanism to describe the audio in order to perform event detection. The Bag of Audio Words approach first learns a codebook of audio "words" using a clustering mechanism such as K-means or random forests to cluster feature vectors such as mel-frequency cepstra. Feature vectors from newer recordings are clustered into these codewords, to result in a bag of (code) words for the recording. A histogram representing counts of occurrence of each "word" is generated which is then used for the purpose of event detection using a classifier. Although the bag of audio words approach presents a simple and successful approach and has been widely used for audio event detection, there are constraints related to it in the context of detection of short duration finer events.

2.1. Detection of finer audio events in short duration recordings

In our present work our objective is to detect finer events such as clapping, clanking clicking sounds, scraping (complete list in section 4) in short duration audio recordings. This has been studied mainly due to three reasons. *Firstly* detection of finer events in a recording will permit description of recordings, as opposed to categorization into broad event categories. *Secondly*, broader event categories detection can be modeled on detection of finer events, for instance through a decision fusion model which can integrate finer events detection decision to detect broader events. For example, the detection of a birthday party can be modeled on finer events such as clapping, children voices, singing etc. *Thirdly*, in an ideal multimedia event detection task we would like to know all the events present in a recording at different times. A time-stamped analysis of events is highly desirable. A very simple approach for this would be to analyze small segments (say 1 sec) of the given recording and detect events present in that small segment. Although at a slightly coarse level this can give us a well-timed analysis of events in audios.

2.2. Issues with bag-of-audio-words in detecting short duration finer audio events

When we aim to detect finer events and that too in short duration clips three major problems with the Bag of Audio Words approach comes into picture. They are as follows:

1. The BoAW approach uses a clustering scheme such as vector quantization for generating bags of audio words from the raw feature vectors (e.g. MFCC) of an audio recording. The generated bags of words are represented as word-count histograms, which simply count the number of times each audio "word" is chosen by some feature vector from the recording. Quantization schemes such as vector quantization allocate a cluster to a vector based on the distance of the vector from the clusters; the vector is assigned to the cluster it is closest to. Often, however, the selection of the "closest" cluster is unclear – the difference in the distance of the top several closest clusters may be small enough that the assignment of closest cluster effectively becomes arbitrary. The assignment of vectors to clusters decides the histograms used to represent a recording. Histograms generated from such assignment are thus highly susceptible to distortion due to any aberration captured in the feature vectors. In user generated data like those in Youtube where sounds are often mixed, and noise is frequently present, this is expected to occur to a larger extent and the event may not be properly characterized by histograms. This can have serious impact on the performance of the overall system. Also the distortion of histogram will be more visible in short duration clips where the number of raw feature vectors is lesser and a few miss-assignments will be much more visible and hence detectable by classifier.
2. The second problem is related to codebook size. In [3] experiments were performed for codebook sizes varying from 500 to 2000, and a codebook size of 1000 was reported to be the best one for events under consideration. In [4], [6] codebook of size 4000 was used for generating the audio "words". The variation in the reported optimal number indicates that the best codebook size will depend on the events under consideration and rigorous experimentation will be required to decide the optimal codebook size. A smaller sized codebook produces a more general vocabulary of audio but it is less discriminative. A larger codebook size is more discriminatory because it will put similar sounding sounds to different audio "words". For finer events such as those we address in this paper, it is expected that a more discriminatory vocabulary will be required and hence a larger codebook size in the range of 2000 to 4000 will be required. This in turn not only impacts computation time; it

also makes the representation more susceptible to random variations in data such as those mentioned earlier.

3. The third one is related to detection of events in short duration audio recordings. Histograms generated for short duration recordings with large codebook sizes will result in sparse histograms. This in turn places constraints on the classification scheme employed which must now be able to perform on sparse feature vectors.

3. PROPOSED APPROACH

The discussion of point number 1 in section 2.2 gives us the idea that instead of hard assignment of a vector to a cluster, we require *soft* assignments, where due consideration is given to all the clusters based on how far they are from the vector. This will address multiple issues – the resulting representation will naturally be more robust to variations in the data; moreover it is less likely to be sparse.

We do this by modeling the clusters using a Gaussian Mixture Model (GMM). Audio data are represented as sequences of Mel-frequency cepstral vectors (MFCCs). The MFCCs from a large number of training recordings are used to learn a universal background GMM model of *M* Gaussian components. Once we have a background GMM model we propose that the probabilistic distribution of MFCC vectors over the components of background GMM can characterize different events under consideration. Moreover, using Gaussian modeling we expect to keep *M* low, to around 100 for reasonable success in event detection, as empirically supported by our experiments.

3.1. GMM based probabilistic feature vector generation

We use MFCCs features of the audio recordings as primary features for the observed acoustic data. For each training audio recording we have a sequence of *d* dimensional MFCCs vectors denoted by $\vec{x}_t$ where *t* goes from 1 to *T*. *T* is the total number of cepstral vectors for the given recording. For each component of the background GMM we compute

$$P(i) = \sum_{t=1}^{T} p(\vec{x}_t / \lambda_i) \tag{1}$$

where λ_i collectively represents the mean and covariance parameters for the i^{th} Gaussian component of the background Gaussian Mixture Model. To account for varying lengths for different audio recording normalization is done as

$$P(i) = \frac{1}{T} \sum_{t=1}^{T} p(\vec{x}_t / \lambda_i) \tag{2}$$

Thus $P(i)$ is a normalized soft-word-count histogram representing the recording. This gives an *M* dimensional feature vector $\vec{F}$ where each element of $\vec{F}$ is equal to $P(i)$

This feature vector $\vec{F}$ thus captures the distribution of all the mel-frequency cepstral vectors of the recording over the Gaussian components of the background GMM.

3.2. GMM-MAP features

A more effective characterization is obtained by actually representing the distribution of the feature vectors in the recording. To do so, we train GMMs for each audio recording by adapting from the background GMM. The means of the background GMM are adapted for each training recordings using the maximum-a-posteriori (MAP) criterion as described in [13]. This is done as follows for i^{th} component of the mixture

$$\Pr(i / \vec{x}_t) = \frac{w_i p(\vec{x}_t / \lambda_i)}{\sum_{j=1}^{M} w_j p(\vec{x}_t / \lambda_j)} \tag{3}$$

$$n_i = \sum_{t=1}^{T} \Pr(i / \vec{x}_t) \tag{4}$$

$$E_i(\vec{x}) = \frac{1}{n_i} \sum_{t=1}^{T} \Pr(i / \vec{x}_t) \vec{x}_t \tag{5}$$

w_i is the weight of i$^{\text{th}}$ Gaussian component.

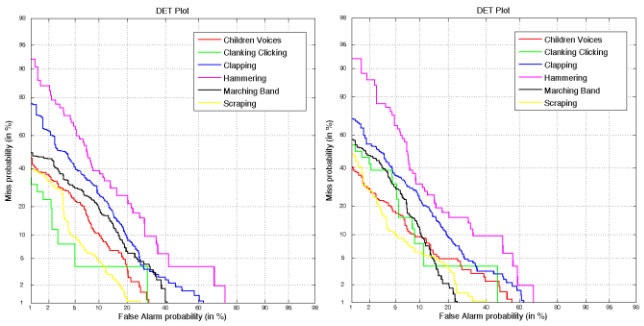
Finally the updated means are computed as

$$\hat{\vec{\mu}}_i = \frac{n_i}{n_i + r} E_i(\vec{x}) + \frac{r}{n_i + r} \vec{\mu}_i \tag{6}$$

where $\vec{\mu}_i$ is the mean vector of i^{th} component of the background GMM and *r* is a relevance factor. The means of all components are then appended to form a new vector of *M* x *d* dimensions. This feature is similar to that used in [14] where it has been used specifically for acoustic fall detection. In our experiments we first use F-vector as a standalone feature and then along with adapted means to train the classifier. Feature dimensionality is *M* or *M* x *(d+1)* depending on whether F-vector alone is used or in combination with GMM-MAP features.

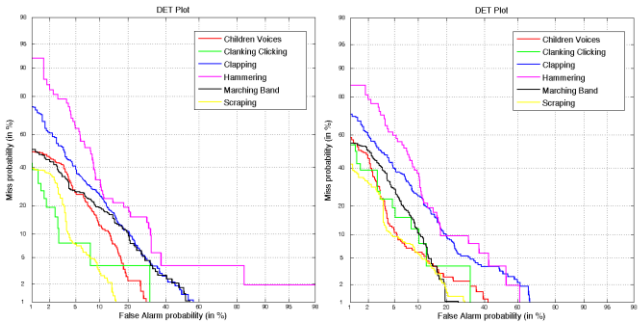
3.3. Random Forest Classifier

Since we need to detect the presence or absence of each event in the given audio recording, the classifier is trained in one versus rest fashion. Classification in all experiments is performed using a random forest classifier [15]. The random forest classifier is a method of ensemble learning in which a given number of decision trees are grown. Each tree in random forest classifier is grown in a slightly different manner than conventional decision trees. In conventional



(a) (b)

Figure 1. DET curves for events with (a) $\vec{F}$ feature only and (b) $\vec{F}$ and GMM-MAP feature combined (M=64)



(a) (b)

Figure 2. DET curves for events with (a) $\vec{F}$ feature only and (b) $\vec{F}$ and GMM-MAP feature combined (M=128)

Table1. AUC and EER values for Miss Detection vs False Alarm

Events	$\vec{F}$ and GMM-MAP features								Bag of Audio Words	
	M=64				M=128				M=1024	
	F-vector alone		F and GMM-MAP combined		F-vector alone		F and GMM-MAP combined			
	AUC	EER	AUC	EER	AUC	EER	AUC	EER	AUC	EER
Children Voices	0.0346	0.103	0.0376	0.093	0.0408	0.110	0.0319	0.069	0.0499	0.143
Clanking	0.0173	0.052	0.0436	0.084	0.0200	0.071	0.0348	0.092	0.0470	0.089
Clapping	0.0766	0.154	0.0743	0.150	0.0747	0.152	0.0805	0.155	0.1170	0.202
Hammering	0.1286	0.206	0.1262	0.178	0.1244	0.187	0.1020	0.159	0.1536	0.227
Marching Band	0.0534	0.138	0.0386	0.102	0.0587	0.142	0.0373	0.101	0.0892	0.190
Scraping	0.0231	0.067	0.0273	0.074	0.0206	0.064	0.0251	0.077	0.0560	0.137

decision trees the best split at each node is computed using all the variables of the input whereas in a random forest the best split at each node is computed using only a subset of input variables randomly chosen at that node. No pruning is done while growing the trees of the forest. The random forest classifier is naturally robust to overfitting; the out-of bag error gives an estimate of the performance of the classifier and hence cross validation as a separate step is not required. In classification using random forest each tree of the forest votes for a class and the total vote obtained by each class is used for the final prediction.

4. EXPERIMENTS AND RESULTS

We performed our experiments on the TRECVID, 2011 corpus. As described previously the dataset contains clips belonging to one of the 15 broad event categories such as attempting a board trick, feeding an animal, landing a fish etc. Finer level audio event labels were provided to us by SRI Sarnoff Labs. These finer events are like clapping, children voices, footsteps, machine sounds, marching bands sound etc. In our experiments we chose 6 audio events namely *children voices, clanking-clicking sound, clapping, hammering, marching band and scraping*. These events are chosen mainly because they represent finer audio events. Although marching band is slightly complex sound we consider it in our experiments because they usually have distinct characteristics which can be identified without relying on other sounds. The total amount of data in seconds available for these events are children voices-1068, clanking clicking sound-110, clapping-1496, hammering- 209, marching band-1156 and scraping-1916. Training is done on 75% of the total data and testing is done on the rest. 13 dimensional MFCCs are computed over every 20ms window with a 10 ms (50%) overlap. Testing is done on 1 sec. clip of each event based on the argument that larger clips can be broken down into segments of one sec. and each segment can be tested separately. Experiments with background GMM of size 64 and 128 have been performed. Although detection results for similar events on short duration clips has not been reported by other authors as far as we know; to compare our results with bag of audio words approach we ran an experiment based on it with codebook size of 1024. Value of '*r*' is fixed to be equal to 0.5.

Figure 1 and Figure 2 show the DET curves for events using (a) F-vector features alone (b) using both F-vector and GMM-MAP features with component size (M) equal to 64 and 128 respectively. *Figures are best viewed in color*. The performance metrics are area under the miss detection and false alarm curve (AUC) values and the Equal Error Rate (EER). These values are reported in Table1. Ideally AUC values should be 0. Smaller the value better is the result.

EER represent the value at which miss detection rate is equal to the false alarm rate.

5. CONCLUSIONS AND FUTURE WORK

Experimental results show that the proposed method is able to detect short duration finer audio events with reasonable success. The AUC and EER values suggest that the F-vector based methods outperform Bag of Audio Words approach for all the events considered. The current set of experiments suggest that using GMM-MAP features along with F-vector can lead to a significant improvement in missed detection vs. false alarm curve for events such as hammering and marching band. In a longer experiment with larger number of events it is expected that a combination of F-vector and GMM-MAP features will be a more robust feature set and hence will lead to better results for most of the events. We are able contain the component size (M) to around 100. Also, repeated experiments to determine optimal component size may not be necessary as reasonable success has been achieved with M as small as 64. The results show that F-vectors combined with GMM-MAP features is more promising than clustering and codebook based methods for detection of short duration finer events. The F-vector distribution is more resistant to distortion by any aberration captured in cepstral vectors.

The detection of finer audio events with reasonable accuracy on short duration clips is of significance for events boundary detection in a large audio recording. Higher level semantic associations may also be made by building on the detection of lower-level events. We continue to investigate in these directions.

6. ACKNOWLEDGMENTS

This work was supported in part by BSNL-IIT Kanpur Telecom Center of Excellence. We would also like to thank SRI Sarnoff Labs for providing us annotations of the data.

7. REFERENCES

[1] K. Lee, D. Ellis and A. Loui "Detecting local semantic concept in environmental sounds using markov model based clsutering" in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010.

[2] L. Lu, F. Ge, Q. Zhao, and Y. Yan, "A svm-based audio event detection system," in *International Conference on Electrical and Control Engineering*, 2010.

[3] S.Pancoast and M. Akbacak, "Bag-of-Audio-Words approach for multimedia event classification" in *Interspeech Conference,* 2012.

[4] Y.G. Jiang, X. Zeng, G. Ye, S. Bhattacharya , D.Ellis, M. Shah and Shih-Fu Chang. "Columbia-UCF TRECVID 2010 multimedia event detection;Combining multiple modalities, contextual concepts, and temporal matching." in *NIST TRECVID Workshop*, 2010.

[5] M. Ayari, J. Delhumeau, M. Douze, H. Jégou, D. Potapov, J. Revaud, C. Schmid, and J. Yuan. "INRIA@ TRECVID'2011: Copy Detection & Multimedia Event Detection." in *TRECVID Workshop*, December 2011.

[6] G. Ye, I. Jhuo, D. Liu, Y.G Jiang, D. T. Lee, and Shih-Fu Chang. "Joint audio-visual bi-modal codewords for video event detection" in *Proceedings of the 2nd ACM Intl. Conf. on Multimedia Retrieval*, p. 39. ACM, 2012.

[7] K..Lee and D. Ellis. "Audio-based semantic concept classification for consumer video" *IEEE Transaction on Audio, Speech and Language Processing*, vol. 18:6, pp. 1406–1416, 2010.

[8] X. Zhuang, X. Zhou, T. S. Huang, and M. Hasegawa-Johnson, "Feature analysis and selection for acoustic event detection," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, 2008.

[9] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," in *IEEE Conference on advanced Video and Signal Based Surveillance*, pp. 21–26, 2010.

[10] A. Pikrakis, T. Giannakopoulos, and S. Theodoridis, "Gunshot detection in audio streams from movies by means of dynamic programming and bayesian networks," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, 2008.

[11] Y. Liu, W.L. Zhao, C.W Ngo, C.S. Xu, and H.Q. Lu. "Coherent bag-of audio words model for efficient large-scale video copy detection." in *Proc. of the ACM Intl. Conf. on Image and Video Retrieval* pp. 89-96. ACM, 2010.

[12] Trecvid 2011,"www.nist.gov/itl/iad/mig/med11.cfm."

[13] F. Bimbot, J.F. Bonastre, C. Fredouille, G. Gravier, I. M. Chagnolleau, S. Meignier, T. Merlin, J. O. García, D. P. Delacrétaz, and D. A. Reynolds. "A tutorial on text-independent speaker verification" *EURASIP Journal on Advances in Signal Processing* 2004, no. 4 (2004): 101962.

[14] X. Zhuang, J. Juang, G. Potamianos, and M. Hasegawa-Johnson, "Acoustic fall detection using gaussian mixture models and gmm supervectors," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, 2009.

[15] L. Breiman, "Random forests," Machine Learning, vol. 45, pp. 5–32, 2001.