AUDIO ANALYSIS FOR SURVEILLANCE APPLICATIONS

Regunathan Radhakrishnan, Ajay Divakaran and Paris Smaragdis

Mitsubishi Electric Research Labs
201 Broadway
Cambridge, MA, USA
regu@merl.com, ajayd@merl.com, paris@merl.com

ABSTRACT

We proposed a time series analysis based approach for systematic choice of audio classes for detection of crimes in elevators in [1]. Since all the different sounds in a surveillance environment cannot be anticipated, a surveillance system for event detection cannot completely rely on a supervised audio classification framework. In this paper, we propose a hybrid solution that consists two parts; one that performs unsupervised audio analysis and another that performs analysis using an audio classification framework obtained from off-line analysis and training. The proposed system is capable of detecting new kinds of suspicious audio events that occur as outliers against a background of usual activity. It adaptively learns a Gaussian Mixture Model(GMM) to model the background sounds and updates the model incrementally as new audio data arrives. New types of suspicious events can be detected as deviants from this usual background model. The results on elevator audio data are promising.

1. INTRODUCTION

Past work on event detection from surveillance data has mostly concentrated on video analysis. In this paper, we focus on audio analysis for the same. Audio analysis can take us closer to semantics than video analysis could and also is computationally more efficient. This motivates us to detect events in surveillance based on an audio classification framework that classifies every time segment of audio into one of a set of trained audio classes. Since certain sound classes (e.g banging and screaming sounds) are indicative of suspicious events, such an audio classification framework can be used to detect suspicious events. In [1], we proposed a time series analysis framework for the systematic choice of these audio classes for the framework.

A surveillance system based on an audio classification framework (with supervised models learned for sound classes obtained off-line), would only be able to detect known kinds of suspicious activity (e.g the ones that are accompanied by banging sounds and screaming). However, it is also important to detect suspicious events that the system has not seen before. Towards that end, we propose a hybrid solution that consists two parts; one that performs unsupervised audio analysis and another that performs analysis using an audio classification framework obtained from off-line analysis and training. The unsupervised audio analysis is motivated by the observation that suspicious events happen as outliers against a background of usual sounds. A GMM is used to parameterize the distribution of sound features that characterize the usual background sounds. As new audio data arrives, its likelihood under the current model of the background is used to flag a suspicious event. In the absence of a suspicious event, the current GMM is updated incrementally. The audio classification framework that is obtained from off-line analysis and training, is used to flag known suspicious activity and to suppress known false alarms (e.g. outliers that are not interesting from the standpoint of the application).

The rest of the paper is organized as follows. In the next section, we describe the proposed hybrid solution for surveillance based on audio analysis. In section 3, we describe the time series analysis and training procedure for the audio classification framework. In section 4, we describe the adaptive background modelling procedure that uses newly arrived audio data to incrementally update the current GMM. In section 5, we present the experimental results with elevator surveillance data before we conclude in section 6.

2. PROPOSED FRAMEWORK

The proposed framework for suspicious event detection is shown in Fig. 1. The framework has two parts: one that analyzes the training audio data off-line and another that analyzes the real time audio data. Let us first describe the role of each of the off-line analysis blocks.

- **Feature extraction:** In this step, low-level features are extracted from the input content in order to generate a time series from which events are to be discovered. For example, the extracted features from the audio stream, could be Mel Frequency Cepstral Coefficients (MFCC) or any other spectral/cepstral representation of the input audio.
- **Detection of subsequences that are outliers in time series:** In this step, we detect outlier subsequences from the time series of low-level features. It is motivated by the observation that "suspicious" events are unusual events in a background of "usual" events. For instance, there is a burst of screaming sounds in the vicinity of a suspicious event in an otherwise relatively silent background.
- **Clustering of detected outliers** The output of the previous analysis block not only gives an inlier/outlier based temporal segmentation of the content but also distinguishable sound classes for the chosen low-level features in terms of distinct backgrounds and outlier sound classes. Then, by examining individual clusters from the detected outliers one can identify consistent patterns in the data that correspond to the events of interest and build supervised statistical learning models.

The aforementioned off-line analysis of the training data for surveillance helps in the systematic choice of audio classes. The chosen audio classes can be further grouped into one of the follow-

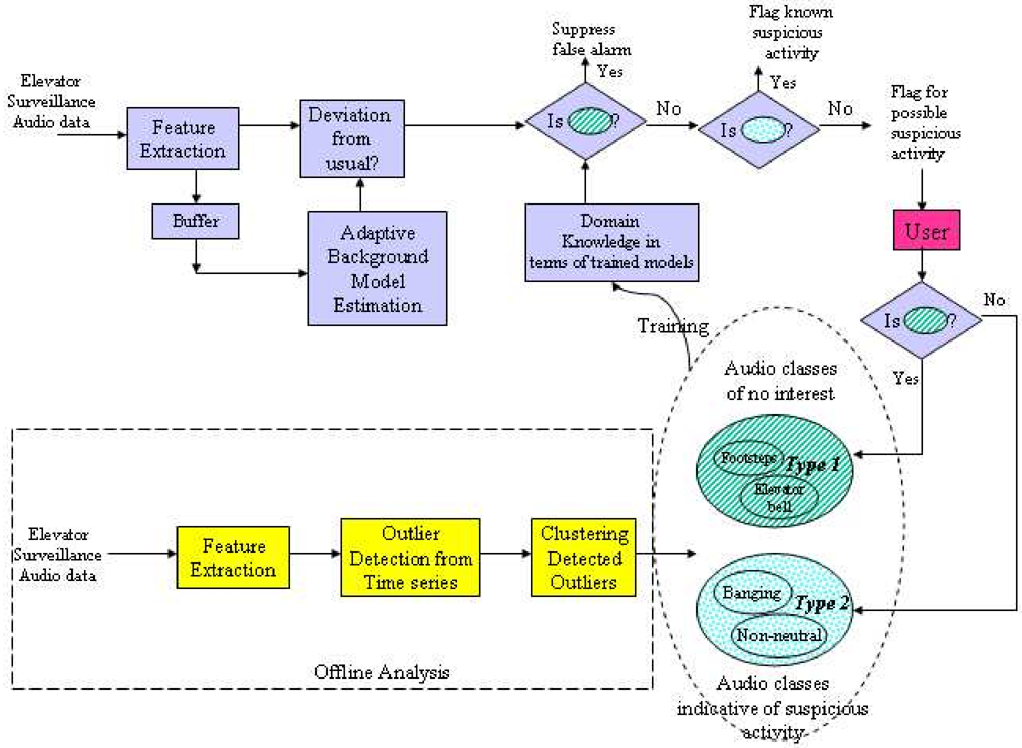
Elevator Surveillance Audio data
Feature Extraction
Deviation from usual?
Buffer
Adaptive Background Model Estimation
Suppress false alarm
Yes
Is ?
No
Flag known suspicious activity
Yes
Is ?
No
Flag for possible suspicious activity
Domain Knowledge in terms of trained models
Training
User
Is ?
No
Yes
Audio classes of no interest
Footsteps
Type 1
Elevator bell
Elevator Surveillance Audio data
Feature Extraction
Outlier Detection from Time series
Clustering Detected Outliers
Banging
Type 2
Non-neutral
Offline Analysis
Audio classes indicative of suspicious activity

Figure 1: *Proposed Framework for suspicious event detection using audio analysis*

ing two types: ***Type 1***:"classes that are **NOT** indicative of suspicious events" and ***Type 2***:"classes that are indicative of suspicious events".

Now, let us look at the analysis blocks that work with real-time audio data.

- **Feature extraction:** The features extracted for online analysis are same as those extracted for off-line analysis.
- **Buffer:** This is the buffer from which a background model is estimated initially and determines the reliability of the estimated background model.
- **Adaptive background model estimation:** As in the case of time series outlier detection block in off-line analysis, the goal of this analysis block also is to detect a burst of time series observations that are not from the usual process. However, now the outlier subsequence detection has to work with only causal data. Towards that end, a model of background sounds is adaptively learned from a window of data stored in the buffer. The likelihood of new audio data under the current background model is compared against a threshold to flag an outlier. In the case of an inlier, the background model is updated. In the case of an outlier, it is checked to see if it belongs to ***Type 1****:* or ***Type 2****:*. ***Type 1****:* implies it is a known false alarm to be suppressed and ***Type 2****:* implies it is a known suspicious event. When encountered with a new type of outlier that has a small likelihood under the models for ***Type 1****:* and ***Type 2****:* classes, end user's feedback is sought to create a model for the new type of outlier.
- **Domain knowledge:** This analysis refers to the audio classification framework which tries to determine whether a detected outlier belongs to ***Type 1****:* or ***Type 2****:*.

In the following section, we describe the off-line analysis and training procedure for the audio classification framework.

3. OFF-LINE ANALYSIS FOR THE CHOICE OF AUDIO CLASSES

Given a large collection of surveillance audio data for training, the volume of data and lack of domain knowledge rule out the option of selecting these audio classes based on intuition. There is a definite need for a framework to systematically select these audio classes to characterize the domain to be able to detect events successfully.

We proposed one such framework to systematically acquire domain knowledge to arrive at the audio classes that would characterize the different sounds in a given domain. We treat the low-level audio features as a multivariate time series and use the time series analysis framework proposed in [1] to perform an inlier/outlier based temporal segmentation of the audio content. The analysis framework in [1] models suspicious events as "unusual" events in a background of "usual" events. Then, by performing an automatic clustering on the detected outliers, we identify consistent patterns for which we can train supervised detectors.

Since these sound classes were obtained as distinguishable sounds from the data, we already know what features and supervised learning method are to be used for modelling them. Hence, we use a GMM to model the distribution of low-level features for each of the chosen sound class. The low-level features that we extracted were 12 MFCC coefficients for every $8ms$ frame of audio. Let us represent this vector of 12 dimensions by a random vector Y whose distribution is to be modelled by a GMM. Let M denote the dimensionality of Y and let K denote the number of Gaussian mixtures, and we use the notation π, μ, and R to denote the parameter sets $\{\pi_k\}_{k=1}^{K}$, $\{\mu_k\}_{k=1}^{K}$, and $\{R_k\}_{k=1}^{K}$, respectively, for mixture coefficients, means, and variances. The complete set of parameters are then given by K and $\theta = (\pi, \mu, R)$. The log of the probability of the entire sequence $Y = \{Y_n\}_{n=1}^{N}$ is then given by

$$\log p_y(y|K,\theta) = \sum_{n=1}^{N} \log \left(\sum_{k=1}^{K} p_{y_n|x_n}(y_n|k,\theta)\pi_k \right) . \qquad (1)$$

The objective is then to estimate the parameters K and $\theta \in \Omega^{(K)}$. The maximum likelihood (ML) estimate is given by

$$\hat{\theta}_{ML} = \arg \max_{\theta\in\Omega^{(K)}} \log p_y(y|K,\theta)$$

the estimate of K is based on the minimization of the expression

$$MDL(K,\theta) = -\log p_y(y|K,\theta) + \frac{1}{2}L\log(NM) , \qquad (2)$$

where L is the number of continuously valued real numbers required to specify the parameter θ. In this application,

$$L = K\left(1 + M + \frac{(M+1)M}{2}\right) - 1 .$$

Notice that this criterion has a penalty term on the total number of data values NM, suggested by Rissanen called the MDL estimator. Let us denote the parameter learning of GMMs using the MDL criterion MDL-GMM.

While the expectation maximization (EM) algorithm can be used to update the parameter θ, it does not provide a solution to the problem of how to change the model order K. Our approach will be to start with a large number of clusters, and then sequentially decrement the value of K. For each value of K, we will apply the EM update until we converge to a local minimum of the MDL functional. After we have done this for each value of K, we may simply select the value of K and corresponding parameters that resulted in the smallest value of the MDL criterion.

The question remains of how to decrement the number of clusters from K to $K-1$. We will do this by merging two closest clusters to form a single cluster. More specifically, the two clusters l and m are specified as a single cluster (l,m) with prior probability, mean and covariance given by

$$\pi^*_{(l,m)} = \bar{\pi}_l + \bar{\pi}_m \qquad (3)$$

$$\mu^*_{(l,m)} = \frac{\bar{\pi}_l\bar{\mu}_l + \bar{\pi}_m\bar{\mu}_m}{\bar{\pi}_l + \bar{\pi}_m} \qquad (4)$$

$$R^*_{(l,m)} = \frac{\bar{\pi}_l\left(\bar{R}_l + (\bar{\mu}_l - \mu_{(l,m)})(\bar{\mu}_l - \mu_{(l,m)})^t\right)}{\bar{\pi}_l + \bar{\pi}_m} + \frac{\bar{\pi}_m\left(\bar{R}_m + (\bar{\mu}_m - \mu_{(l,m)})(\bar{\mu}_m - \mu_{(l,m)})^t\right)}{\bar{\pi}_l + \bar{\pi}_m} . \qquad (5)$$

Here the $\bar{\pi}$, $\bar{\mu}$, and $\bar{R}$ are given by the EM update of the two individual mixtures before they are merged. Finally, once we have trained MDL-GMMs for each sound class, to classify a given a test clip we use maximum likelihood criterion using the learned models. For more details on the audio classification framework, please see [2].

4. ADAPTIVE BACKGROUND MODELLING USING INCREMENTAL LEARNING OF GMMS

In this section, we present a brief description of the online background modelling for event detection. The proposed framework is motivated by the observation that “suspicious” events in surveillance audio happen sparsely in a background of usual or “uninteresting” events. For instance, there is a burst of screaming noise following a suspicious event in a relatively silent background. This motivates us to formulate the problem of detecting “suspicious” events online as that of detecting the onset outlier subsequences that are not from the usual background process.

Let p_1 represent a realization of the “usual” process ($\mathbf{P_1}$) which can be thought of as the background process. Let p_2 represent a realization of the “unusual” process $\mathbf{P_2}$ which can be thought of as the foreground process. Given any causal time sequence of observations or low-level audio features from the two the classes of events ($\mathbf{P_1}$ and $\mathbf{P_2}$), such as

$$...p_1p_1p_1p_1p_1p_1p_2p_2p_2p_2...$$

then the problem is to find the onset and times of occurrences of realizations of $\mathbf{P_2}$ without any a priori knowledge about $\mathbf{P_1}$ or $\mathbf{P_2}$.

Given a causal time series of audio features, we estimate the background process P_1 by training a GMM from a window of W_L observations,$\{O_1, O_2, ...O_{W_L}\}$. The number of mixtures for this model is obtained by using minimum description length principle. Let us represent this GMM as G_b. Then, for every subsequent W_S observations, $\{O_1, O_2, ...O_{W_S}\}$, we compute its deviation from the background process as:

$$d = log(P(\{O_1, O_2, ...O_{W_L}\}/G_b)) - log(P(\{O_1, O_2, ...O_{W_S}\}/G_b))$$

When there is a burst of observations of from P_2, the value of d would be large indicating the onset of the burst or the “foreground” event. On the other hand, when the burst of observations is from the same background process, P_1, one would like to update the current background model (G_b) with the new information. In order to perform this incremental update we perform the following steps:

- **Step 1:**Estimate a Minimum Description Length GMM from observations ($\{O_1, O_2, ...O_{W_S}\}$). Let us represent this as GMM_f.
- **Step 2:**Assign each O_i from ($\{O_1, O_2, ...O_{W_S}\}$) to a mixture component in G_f by finding the component that has maximum posterior probability.
- **Step 3:**For each component k in G_f, search for a matching mixture component in G_b by finding the most likely component in G_b that could have generated the observations assigned to component k in the previous step. If a match is found (say component j in G_b) merge component k from G_f and j from G_b as given below:

$$\mu = \frac{W_L\pi_j\mu_j + W_S\mu_k}{W_L\pi_j + W_S}$$

$$\Sigma = \frac{W_L\pi_j\Sigma_j + W_S\Sigma_k}{W_L\pi_j + W_S} + \frac{W_L\pi_j\mu_j\mu_j^T + W_S\mu_k\mu_k^T}{W_L\pi_j + W_S} - \mu\mu^T$$

	[1]	[2]	[3]	[4]
[1]	0.80	0.06	0.14	0.00
[2]	0.03	0.80	0.10	0.07
[3]	0.03	0.0	0.90	0.07
[4]	0.00	0.16	0.07	0.77

Table 1: Recognition Matrix (Confusion Matrix) on a 90% training/10% testing split of a data set composed of 6 audio classes [1]: Banging; [2]: Footsteps; [3]: non-neutral; [4]: normal speech;

$$\pi = \frac{W_L \pi_j + W_S}{W_L + W_S}$$

where μ, Σ and π are updated parameters of the background model; μ_j, Σ_j and π_j are parameters of component j in G_b; μ_k, Σ_k and π_k are parameters of component k in G_f.

For each component k in G_f that doesn't have a match in G_b, create a new component with mean equal to μ_k and covariance equal to Σ_k with component weight $\frac{W_S}{W_L+W_S}$.

Similarly, for each component j in G_b that does not have a match in G_f, create a new component with mean equal to μ_j and covariance equal to Σ_j with component weight $\frac{W_L \pi_j}{W_L+W_S}$.

These update rules follow those derived in [3]. For details, please see [3].

In the following section, we present some results of the proposed framework.

5. EXPERIMENTAL RESULTS

The results presented in this section are from audio surveillance data in elevators. It consists of 126 clips (2 hours of content) with suspicious events and 4 clips (40 minutes of content) that are without events. We extract low-level features from 61 clips (1 hour of content) of all the suspicious event clips and 4 clips of normal activity in the elevators (40 minutes of content). The low-level features were 12 MFCC features extracted for 8ms frame of audio. Then, for each of the clips we perform inlier/outlier based segmentation with the proposed framework to detect outlier subsequences. A subsequent clustering of the detected outliers gave us the following four audio classes, namely, Banging, Footsteps, non-neutral speech, normal speech. Banging and non-neutral speech belong to ***Type 2:*** outlier category whereas the other two belong to ***Type 1:***.

Table 1 summarizes the results of audio classification framework for which the audio classes were systematically obtained through the offline time series analysis proposed in [1].

Now, let us look at the results of online adaptive background modelling on the elevator surveillance data. Figure 2 shows the deviation measure for every $W_S = 20$s of audio for a 8 min clip with a suspicious event in the vicinity of 4 mins. The initial background model was learnt from $W_L = 120$s of audio. There are two thresholds involved in the online adaptive background modelling described earlier: one on the value of d to determine foreground audio objects (consequently to decide whether or not to update the background model) and the second threshold on the likelihood value to determine whether a component from GMM_f is statistically equivalent to a component in GMM_b. With the same threshold settings, we were able to detect suspicious events in most of the clips.

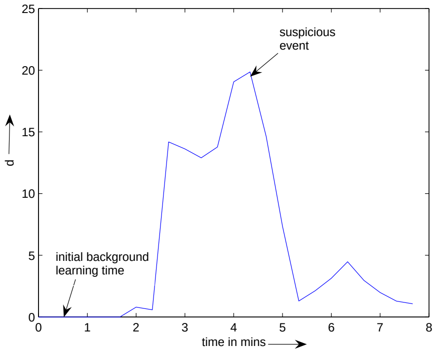

Figure 2: *Deviation (d) from adaptive background model for a clip with suspicious event*

6. CONCLUSIONS

We proposed a hybrid audio analysis framework for surveillance that consists two parts; one that performs unsupervised audio analysis and another that performs analysis using an audio classification framework. The audio classes for the classification framework are obtained from off-line time series analysis of cepstral features and training. It also adaptively learns a Gaussian Mixture Model(GMM) to model the background sounds and updates the model incrementally as new audio data arrives and has been shown to detect suspicious events effectively.

The adaptive background modelling algorithm used in the proposed framework first estimates a GMM from W_S observations and then updates the parameters of statistically equivalent components in the background GMM. An alternative approach proposed in [4], which is analogous to background modelling in computer vision, updates the background model for every new incoming data vector. Our future work would compare these two approaches for audio background modelling.

7. ACKNOWLEDGEMENTS

The authors would like to thank Dr. Ziyou Xiong for his experiments on MDL criterion for learning GMMs.

8. REFERENCES

[1] R.Radhakrishnan and A.Divakaran, "Systematic acquisition of audio classes for elevator surveillance," *Proc. of SPIE*, 2005.

[2] Z. Xiong, R. Radhakrishnan, and A. Divakaran, "Effective and efficient sports highlights extraction using the minimum description length criterion in selecting gmm structures," *Proc. of ICME*, June 2004.

[3] Mingzhou Song and Hongbin Wang, "Highly efficient incremental estimation of gmms for online data stream clustering," *Proc. of SPIE Conference on Intelligent Computing*, 2005.

[4] M.Cristani, M.Bicego and V.Murino, "Online adaptive background modelling for audio surveillance," *Proc. of ICPR*, 2004.