Scream and Gunshot Detection and Localization for Audio-Surveillance Systems*

G. Valenzise L. Gerosa M. Tagliasacchi F. Antonacci A. Sarti
Dipartimento di Elettronica e Informazione – Politecnico di Milano
Piazza Leonardo da Vinci 32, 20133 Milano, Italy
Email: valenzis@elet.polimi.it, luigi@gerosa.biz, tagliasa/antonacc/sarti@elet.polimi.it

Abstract

This paper describes an audio-based video surveillance system which automatically detects anomalous audio events in a public square, such as screams or gunshots, and localizes the position of the acoustic source, in such a way that a video-camera is steered consequently. The system employs two parallel GMM classifiers for discriminating screams from noise and gunshots from noise, respectively. Each classifier is trained using different features, chosen from a set of both conventional and innovative audio features. The location of the acoustic source which has produced the sound event is estimated by computing the time difference of arrivals of the signal at a microphone array and using linear-correction least square localization algorithm. Experimental results show that our system can detect events with a precision of 93% at a false rejection rate of 5% when the SNR is 10dB, while the source direction can be estimated with a precision of one degree. A real-time implementation of the system is going to be installed in a public square of Milan.

1. Introduction

Video-surveillance applications are becoming increasingly important both in private and public environments. As the number of sensors grows, the possibility of manually detecting an event is getting impracticable and very expensive. For this reason, research on automatic surveillance systems has recently received particular attention. In particular, the use of audio sensors in surveillance and monitoring applications has proved to be particularly useful for the detection of events like screams or gunshots [1][2]. Such detection systems can be efficiently used to signal to an automated system that an event has occurred and, at the same time, to enable further processing like acoustic source localization for steering a video-camera.

Much of the previous work about audio-based surveillance systems has concentrated on the task of detecting some particular audio events. Early research stems from the field of automatic audio classification and matching [3]. More recently, specific works covering the detection of particular classes of events for multimedia-based surveillance have been developed. The SOLAR system [4] uses a series of boosted decision trees to classify sound events belonging to a set of predefined classes, such as screams, barks, etc.

Successive works have shown that classification performance can be considerably improved if a hierarchical classification scheme, composed by different levels of binary classifiers, is used in place of a single-level multi-class classifier [5]. This hierarchical approach has been employed in [2] to design a specific system able to detect screams/shouts in public transport environments. A slightly different technique is used in [1] to detect gunshots in public environments. Several binary sub-classifiers for different types of firearms are run in parallel. In this way, the false rejection rate of the system is reduced by a 50% on average with respect to a single gunshot/noise classifier.

The final objective of sound localization in most surveillance systems consists in localizing the acoustic source position over a topological grid. The most popular technique for source localization in environments with small reverberation time (such as a typical public square) is based on the Time Difference of Arrivals (TDOA) of the signal at an array of microphones. These time delays are further processed to estimate the source location [6].

In this paper we propose a surveillance system that is able to accurately detect and localize screams and gunshots. The audio stream is recorded by a microphone array. Audio segments are classified as screams, gunshots or noise. Audio classified as noise is discarded. If an anomalous event (scream or gunshot) is detected, the localization module estimates the TDOAs at each sensor pair of the array and computes the position of the sound source, steering the video-camera accordingly.

Our approach is different from the previous works in the following aspects. First, we give more weight to the phase of *feature selection* for event detection. In traditional audio-surveillance works, features have been either selected by the classification algorithm itself [4] or reduced in dimensionality by Principal Component Analysis (PCA) [1]. In most of the cases, features have been manually selected on the basis of some heuristic criteria [7]. We provide an exhaustive analysis of the feature selection process, mixing the classical filter and wrapper feature selection approaches. Second, in addition to video-camera steering based on localization of the sound source, we compare time delay estimation errors with theoretical results, and we give some hints on heuristic methods for zooming the camera based on the confidence of localization.

*The work presented was developed within VISNET II, a network of excellence of the European Commission (http://www.visnet-noe.org)

978-1-4244-1696-7/07/$25.00 ©2007 IEEE.

2 Audio Features

A considerable number of audio features have been used for the tasks of audio analysis and content-based audio retrieval. Traditionally, these features have been classified in *temporal features*, e.g. Zero Crossing Rate (ZCR); *energy features*, e.g. Short Time Energy (STE); *spectral features*, e.g. spectral moments, spectral flatness; *perceptual features*, e.g. loudness, sharpness or Mel Frequency Cepstral Coefficients (MFCCs). In this work, we have chosen to discard audio features which are too sensitive to the SNR conditions, like STE and loudness. In addition to the traditional features listed above, we employ some other features which have not been used before in similar works, such as *spectral distribution* (spectral slope, spectral decrease, spectral roll-off) and *periodicity* descriptors. In this paper we also introduce a few innovative features based on the *auto-correlation* function: correlation roll-off, correlation decrease, correlation slope, modified correlation centroid and correlation kurtosis.

These features are similar to spectral distribution descriptors (spectral roll-off, spectral decrease and spectral slope [8]), but, in lieu of the spectrogram, they are computed starting from the auto-correlation function of each frame. The goal of these features is to describe the energy distribution over different time lags. For impulsive noises, like gunshots, much of the energy is concentrated in the first time lags, while for harmonic sounds, like screams, the energy is spread over a wider range of time lags. Features based on the auto-correlation function are labeled in two different ways, filtered or not filtered, depending on whether the autocorrelation function is computed, respectively, on a band-pass filtered version of the signal or on the original signal. The rationale behind this filtering approach is that much of the energy of some signals (e.g. screams) is distributed in a relatively narrow range of frequencies; thus the autocorrelation function of the filtered signal is much more robust to noise. In this paper, the limits of the frequency range for filtering the autocorrelation function have been fixed to $1000-2500$ Hz: experimental results have shown that most of the energy of the screams harmonics is concentrated in this frequency range.

Table 1 lists the feature set composition. All the features are extracted from 23 ms analysis frames (at a sampling frequency of 22050 Hz) with $1/3$ overlap.

#	Feature Type	Features	Ref.
1	Temporal	ZCR	[7]
2-6	Spectral	4 spectral moments + SFM	[8]
7-36	Perceptual	30 MFCC	[9]
37-39	Spectral distribution	spectral slope, spectral decrease, spectral roll-off	[8]
40-49	Correlation-based	(filtered) periodicity, (filtered) correlation slope, decrease and roll-off, modified correlation centroid, correlation kurtosis	[7][8]

Table 1: Audio features used for classification.

3 Feature Selection

Starting from the full set of 49 features, we can build a feature vector of any dimension l, $1 \leq l \leq 49$. It is desirable to keep l small in order to reduce the computational complexity of the feature extraction process and to limit the overfitting produced by the increasing number of parameters associated to features in the classification model.

Two main feature selection approaches have been discussed in literature. In the *filter* method, the feature selection algorithm filters out features that have little chance to be useful for classification, according to some performance evaluation metrics calculated directly from the data, without direct feedback from a particular classifier used. In the second approach, known as *wrapper* approach, the performance evaluation metrics is some form of feedback provided by the classifier (e.g. accuracy). Obviously, wrapper approaches outperform filter methods, since they are tightly coupled with the employed classifier, but they require much more computation time.

The feature selection process adopted in this work is a hybrid filter/wrapper method. First, a feature subset of size l is assembled from the full set of features according to some class-separability measure and a heuristic search algorithm, as detailed in Section 3.1. The so-obtained feature vector is evaluated by a GMM classifier, which returns some classification performance indicator related to that subset (this procedure is explained in Section 3.2). Repeating this procedure for different l's, one can choose the feature vector dimension that optimizes the desired target performance.

3.1 Selection of a Feature Vector of size l

This section reviews some heuristic methods used to explore the feature space, searching for a (locally) optimal feature vector. We consider two kinds of search algorithms [10]: *scalar* methods and *vectorial* methods.

3.1.1 Scalar Selection

In this work, we adopt a feature selection procedure described in [10]. The method builds a feature vector iteratively, starting from the most discriminating feature and including at each step k the feature $\hat{r}$ that maximizes the following function:

$$J(r) = \alpha_1 C(r) - \frac{\alpha_2}{k-1} \sum_{i \in \mathscr{F}_{k-1}} |\rho_{ri}|, \text{ for } r \neq i. \qquad (1)$$

In words, Eq. 1 says that the feature to be included in the feature vector of dimension k has to be chosen from the set of features not yet included in the feature subset $\mathscr{F}_{k-1}$. The objective function is composed of two terms: $C(r)$ is a class separability measure of the rth feature, while ρ_{ij} indicates the cross-correlation coefficient between the ith and jth feature. The weights α_1 and α_2 determine the relative importance that we give to the two terms. In this paper, we use either the Kullback-Leibler divergence (KL) or the Fisher Discriminant Ratio (FDR) to compute the class separability $C(r)$ [10].

3.1.2 Vectorial Selection

The vectorial feature selection is carried out using the *floating search* algorithm [10]. This procedure builds a feature vector iteratively and, at each iteration, reconsiders features previously discarded or excludes features selected in previous iterations from the current feature vector. Though not optimal, this algorithm provides better results than scalar selection, but with an increased computational cost. The floating search algorithm requires the definition of a vectorial class separability metrics. In the proposed system, we use either one of the following objective metrics [10]:

$$J_1 = \frac{\text{trace}(S_m)}{\text{trace}(S_w)}, \qquad J_2 = \frac{\det(S_m)}{\det(S_w)} \tag{2}$$

where S_w is the *within-class* scatter matrix, which carries information about *intra*-class variance of the features, while $S_m = S_w + S_b$ is the *mixture* scatter matrix; S_b, the *between-class* scatter matrix, gives information about *inter*-class covariances.

3.2 Selection of the Feature Vector Dimension l

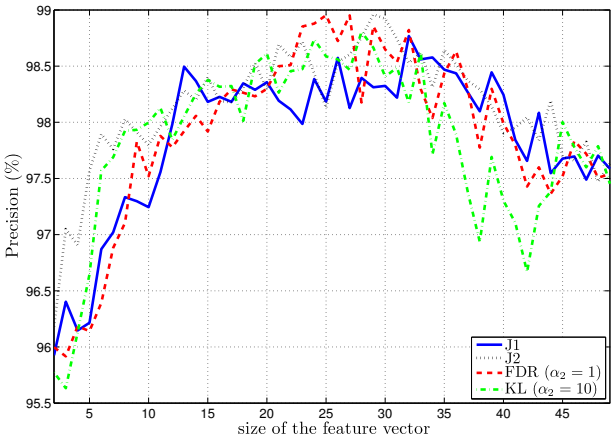
The optimal vector dimension is determined using a wrapper approach. The two classification feedbacks we take into consideration are the precision and the false rejection rate (FR), defined as follows:

$$\text{precision} = \frac{\text{number of events } \textit{correctly} \text{ detected}}{\text{number of events detected}} \tag{3}$$

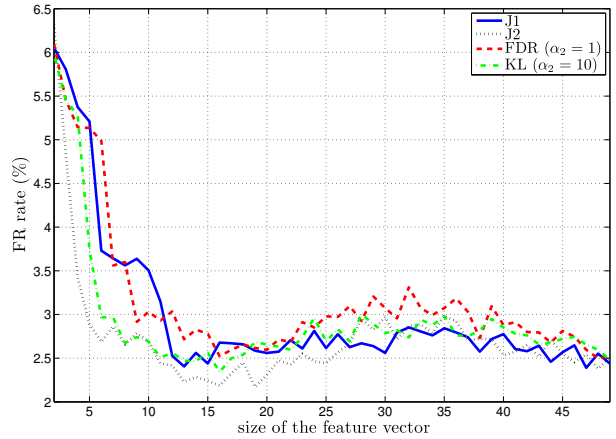
$$FR = \frac{\text{number of events } \textit{not} \text{ detected}}{\text{number of events to detect}}, \tag{4}$$

where the term "event" denotes either a scream or a gunshot. The rationale behind the choice of precision and false rejection rate as performance metrics is that in an audio-surveillance system the focus is on minimizing the number of events "missed" by the control system, while at the same time keeping as small as possible the number of false alarms.

We evaluate the precision and false rejection rate for feature vectors of any dimension l. Figure 1 shows how the performance vary as l increases, for the case of scream events (analogous results are obtained with gunshot samples). From these graphs, it is clear that good performance may be obtained with a small number of features, while increasing l above a certain dimension $\hat{l}$ (e.g. 13-15 in the case of screams) not only the performance does not improve significantly, but the results get worse due to overfitting. The choice of $\hat{l}$ can be formalized as a trade-off optimization problem and will be further investigated in a future work. For now, $\hat{l}$ is selected empirically by inspection of the graphs shown in Figure 1 ($\hat{l} = 13$ for screams and $\hat{l} = 14$ for gunshots).



(a) Precision



(b) False Rejection Rate

Figure 1: Classification precision and false rejection rate of scream with increasing feature vector dimension l.

4 Classification

The event classification system is composed by two Gaussian Mixture Model (GMM) classifiers that run in parallel to discriminate, respectively, between screams and noise, and between gunshots and noise. Each binary classifier is trained separately with the samples of the respective classes (gunshot and noise, or scream and noise), using the Figueiredo and Jain algorithm [11]. This method is conceived to avoid the limitations of the classical Expectation-Maximization (EM) algorithm for estimating the parameters of a mixture model: through an automatic "component annihilation" procedure, the Figueiredo-Jain algorithm automatically selects the number of components and rules out the problem of determining adequate initial conditions; furthermore, singular estimates of the mixture parameters can be automatically avoided by the algorithm.

For the testing step, each frame from the input audio stream is classified *independently* by the two binary classifiers. The decision that an event (scream or gunshot) has

occurred is then taken by computing the logical OR of the two classifiers.

5 Localization

5.1 Time Delay Estimation

The localization system employs a T-shaped microphone array composed of 4 sensors, spaced 30 cm apart from each other. The center microphone is taken as the reference sensor (hereafter referred with the number 0) and the three Time Difference of Arrivals (TDOAs) of the signal between the other microphones and the reference microphone are estimated. We use the Maximum-Likelihood Generalized Cross Correlation (GCC) method for estimating time delays [12], i.e. we search

$$\hat{\tau}_{i0} = \arg\max_{\tau} \hat{\Psi}_{i0}(\tau),\ i = 1,2,3, \tag{5}$$

where

$$\hat{\Psi}_{i0}(\tau) = \sum_{k=0}^{N-1} \frac{S_{x_i x_0}(k)}{|S_{x_i x_0}(k)|} \cdot \frac{|\gamma_{i0}(k)|^2}{|S_{x_i x_0}(k)|\,(1-|\gamma_{i0}(k)|^2)} \cdot e^{\frac{j2\pi\tau k}{N}} \tag{6}$$

is the generalized cross correlation function, $S_{x_i x_0}(k) = E\{X_i(k)X_0^*(k)\}$ is the cross spectrum, $X_i(k)$ is the discrete Fourier transform (DFT) of $x_i(n)$, γ_{i0} is the Magnitude Square Coherence (MSC) function between x_i and x_0, and N denotes the number of observation samples during the observation interval.

To increase the precision, the estimation of $\hat{\tau}_{i0}$ can be refined by a parabolic interpolation [13]. However, a fundamental requirement to increase the performance of (5) is a high-resolution estimation of the cross-spectrum and of the coherence function. We use a non-parametric technique, known as minimum variance distortionless response (MVDR), to estimate the cross spectrum and therefore the MSC function [14]. The MVDR spectrum can be viewed as the output of a bank of filters, with each filter centered at one of the analysis frequencies. Following this approach, the MSC is given by:

$$|\gamma_{i0}(k)|^2 = \frac{\left|\mathbf{f}_k^H \mathbf{R}_{ii}^{-1} \mathbf{R}_{i0} \mathbf{R}_{00}^{-1} \mathbf{f}_k\right|^2}{\left[\mathbf{f}_k^H \mathbf{R}_{ii}^{-1} \mathbf{f}_k\right]^2 \left[\mathbf{f}_k^H \mathbf{R}_{00}^{-1} \mathbf{f}_k\right]^2}, \tag{7}$$

where superscript ^{H} denotes transpose conjugate of a vector or a matrix, $\mathbf{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}(n)^H\}$ indicates the covariance matrix of a signal x, $\mathbf{f}_k = 1/\sqrt{L} \cdot [1\ \exp(j\omega_k)\ \ldots\ \exp(j\omega_k(L-1))]^T$ and $\omega_k = 2\pi k/K$, $k = 0,1,\ldots,K-1$. Assuming that $K = L$ and observing that matrices $\mathbf{R}$ have a Toeplitz structure, we can compute (7) efficiently by means of the Fast Fourier Transform. In our experiments we set $K = L = 200$ and an observation time $N = 4096$ samples.

5.2 Source Localization

Differently from popular localization algorithms, the approach we use needs no *far field* hypothesis about source location, and is based on the spherical error function [6]

$$\mathbf{e}_{\mathrm{sp}}(\mathbf{r}_{\mathrm{s}}) = \mathbf{A}\boldsymbol{\theta} - \mathbf{b}, \tag{8}$$

where

$$\mathbf{A} \triangleq \begin{bmatrix} x_1 & y_1 & d_{10} \\ x_2 & y_2 & d_{20} \\ x_3 & y_3 & d_{30} \end{bmatrix}, \boldsymbol{\theta} \triangleq \begin{bmatrix} x_s \\ y_s \\ R_s \end{bmatrix}, \mathbf{b} \triangleq \frac{1}{2} \begin{bmatrix} R_1^2 - d_{10}^2 \\ R_2^2 - d_{20}^2 \\ R_3^2 - d_{30}^2 \end{bmatrix} \tag{9}$$

for a two dimensional problem. Pairs (x_i, y_i) are the coordinates of the ith microphone, (x_s, y_s) are the unknown coordinates of the sound source, R_i and R_s denote, respectively, the distance of microphone i and of the sound source from the reference microphone, and $d_{i0} = c \cdot \hat{\tau}_{i0}$, with c being the speed of sound.

To find an estimate of the source location we solve the linear minimization problem

$$\min_{\boldsymbol{\theta}} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}) \tag{10}$$

subject to the constraint $x_s^2 + y_s^2 = R_s^2$. The solution of (10) can be found in [6].

6 Experimental Results

In our simulations we have used audio recordings taken from movies soundtracks and internet repositories. Some screams have been recorded live from people asked to shout into a microphone. Finally, noise samples have been recorded live in a public square of Milan.

6.1 Classification performance with varying SNR conditions

This experiment aims at verifying the effects of the noise level on the training and test sets. We have added noise both to the audio events of the training set and to the audio events of the test set, changing the SNR from 0 to 20dB, with a 5dB step. The performance indicators we have used in this test are the false rejection rate, defined in (4), and the false detection rate (FD), defined as follows:

$$FD = \frac{\text{number of detected events that were actually noise}}{\text{number of noise samples in the test set}}, \tag{11}$$

where, as usual, an event could be both a scream or a gunshot. The results for scream/noise classification are reported in Figure 2. As expected, performance degrades noticeably as the SNR of both training and test sequences decreases. In particular, as the training SNR decreases, the false detection rate tends systematically to increase. At the same time, once the training SNR has been fixed, a reduction of SNR on the

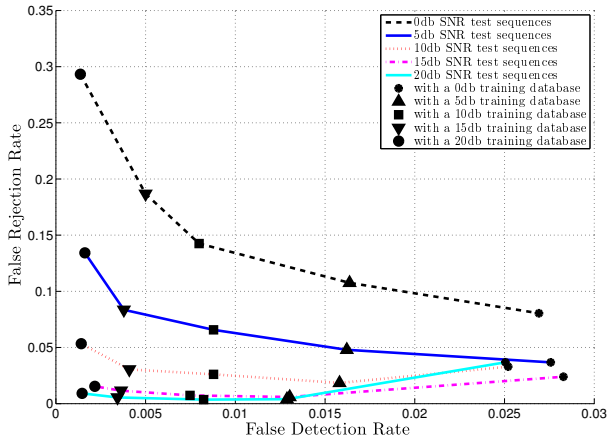


Figure 2: False rejection rate as a function of false detection rate for various SNR training database and test sequences. The graph refers to the scream/noise classifier using $\hat{l} = 20$ features.

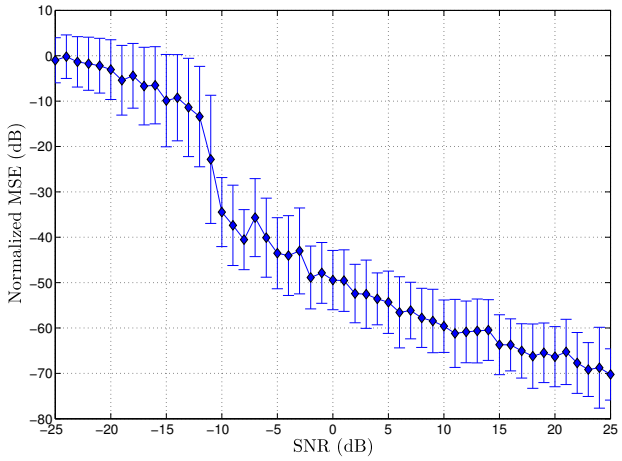


Figure 3: Mean Square Error of delay estimation for gunshot and scream samples at 95% confidence level. Data is normalized to the variance of a uniform random guess.

test set leads to worse performance in terms of false rejection rate. To account for this behavior, we must consider that using a high SNR training set implies that the classifier is trained with almost clean scream/gunshot events. On the contrary, a noisy training set implies that the classifier is trained to detect events *plus noise*. Obviously, in this way the probability of labeling noise as a scream or gunshot is greater. On the other hand, if the training set SNR is high but the system is tested in a noisy environment, the classifier is able to correctly detect only a small fraction of the actual events, since it was not trained to be robust to noise.

This experiment illustrates the trade-off existing between false rejection and false detection rate. According to the average noise conditions of the environment in which the system will be deployed, one should choose the appropriate SNR for the training database. Similar results have been obtained with the gunshot/noise classifier.

6.2 Combined system

Putting together the scream/noise classifier and the gunshot/noise classifier we can yield a precision of 93% with a false rejection rate of 5%, using samples at 10dB SNR. We have used a feature vector of 13 features for scream/noise classification, and a feature vector of 14 features for gunshot/noise classification. In both cases the J2 criterion has been employed. The two feature vectors are reported in Table 2.

#	Scream/Noise classifier	Gunshot/Noise classifier
1	ZCR	SFM
2	SFM	spectral centroid
3	MFCC 2	spectral kurtosis
4	MFCC 3	MFCC 2
5	MFCC 4	MFCC 4
6	MFCC 9	MFCC 6
7	MFCC 11	MFCC 7
8	periodicity	MFCC 19
9	(filtered) periodicity	MFCC 20
10	correlation decrease	MFCC 28
11	filtered correlation decrease	MFCC 29
12	correlation slope	MFCC 30
13	correlation centroid	periodicity
14		spectral slope

Table 2: Feature vectors used in the combined system

6.3 TDE error with different SNR conditions

Localization has been evaluated against different values of SNR by properly mixing audio events with a colored noise with a pre-specified power. To generate the noise samples, we use a white noise to feed an AR process, whose coefficients have been obtained by LPC analysis on ambient noise records. This is necessary to simulate isotropic noise conditions. TDOAs are estimated as explained in Section 5.1; we narrow the search space of Eq. (5) to time lags $\tau \in [-T_{max}, T_{max}]$, where $T_{max} = \lceil d/c \cdot f_s \rceil$, d is the distance between the microphones of a pair (here $d = 30$ cm) and f_s is the sampling frequency ($f_s = 44100$ Hz). The GCC peak estimation is refined using parabolic interpolation.

Figure 3 shows the mean square error (MSE) of the TDOA between a pair of microphones for a scream sample, normalized by $(2T_{max}+1)^2/12$, which corresponds to the variance of a uniform distribution over the search interval. Values in figure are expressed in decibel, while the true time delay for the simulation has been set to 0 without any loss of generality. Analogous results are obtained for gunshots records. From the figure, it's clearly observable the so-called "threshold effect" in the performance of GCC: under some threshold SNR^*, in this example about -10dB, the error of time delay estimation suddenly degrades as far

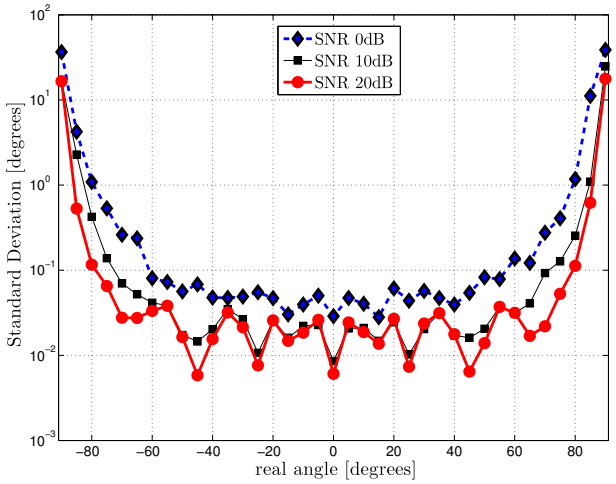


Figure 4: Standard deviation of the estimated angle $\hat{\vartheta}$ between the sound source and the axis of the array, as a function of the true angle. The distance of the source has been fixed to 50 m.

as the estimated TDOA becomes just a random guess. This phenomenon agrees with theoretical results [13]. An immediate consequence of this behavior is that no steering is applied to the video-camera if the estimated SNR is below the threshold. This is feasible in our system since the audio stream is classified as either an audio event or ambient noise. Under the assumption that the two classes of sounds are uncorrelated, the SNR can be easily computed from the difference in power between events and noise, and tracked in real time.

6.4 Localization error

The audio localization system has been tested by varying the actual position of the sound source, spanning a range of $\pm 90^\circ$ with respect to the axis of the array. A source positioned at -90° is on the left of the array, one positioned at 0° is in front of the array, while a source located at +90° is on the right. Figure 4 shows the standard deviation of the estimated source angle $\hat{\vartheta}$ for some SNRs above the threshold. For a T-shaped array, the expected angular error is symmetric around 0°. As can be argued from the graph, if the actual sound source is in the range $[-80^\circ, 80^\circ]$, the standard deviation of $\hat{\vartheta}$ is below one degree, even at 0dB SNR. As the sound source moves completely towards the left or the right of the array, the standard deviation of $\hat{\vartheta}$ increases, specially when the ambient noise level is higher. This behavior can be used for deciding whether the video-camera should be zoomed or not. If $\hat{\vartheta}$ is known with sufficient precision, the camera can be zoomed to capture more details. If the estimation is uncertain, a wider angle should be used. A conservative policy could be to zoom the camera only if $|\hat{\vartheta}|$ falls outside the interval $[90^\circ \pm \sigma_{90}]$, where σ_{90} is the standard deviation of $\hat{\vartheta}$ for a given SNR when the true angle is either 90° or -90°. For example, at 10dB SNR σ_{90} is approximately 20° (see Figure 4).

7 Conclusions

In this paper we analyzed a system able to detect and localize audio events such as gunshots and screams in noisy environments. A real time implementation of the system is going to be installed in the public square outside the Central Train Station of Milan, Italy. Future work will be dedicated to the formalization of feature dimension selection algorithm and to the integration of multiple microphone arrays into a sensor network for increasing the range and the precision of audio localization.

References

[1] C. Clavel, T. Ehrette, and G. Richard, "Events Detection for an Audio-Based Surveillance System," *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pp. 1306–1309, 2005.

[2] J. Rouas, J. Louradour, and S. Ambellouis, "Audio Events Detection in Public Transport Vehicle," *Proc. of the 9th International IEEE Conference on Intelligent Transportation Systems*, 2006.

[3] T. Zhang and C. Kuo, "Hierarchical system for content-based audio classification and retrieval," *Conference on Multimedia Storage and Archiving Systems III, SPIE*, vol. 3527, pp. 398–409, 1998.

[4] D. Hoiem, Y. Ke, and R. Sukthankar, "SOLAR: Sound Object Localization and Retrieval in Complex Audio Environments," *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, vol. 5, 2005.

[5] P. Atrey, N. Maddage, and M. Kankanhalli, "Audio Based Event Detection for Multimedia Surveillance," *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2006*, 2006.

[6] J. Chen, Y. Huang, and J. Benesty, *Audio Signal Processing for Next-Generation Multimedia Communication Systems*. Kluwer, 2004, ch. 4-5.

[7] L. Lu, H. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *Speech and Audio Processing, IEEE Transactions on*, vol. 10, no. 7, pp. 504–516, 2002.

[8] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," *CUIDADO Project Report*, 2004.

[9] S. Sigurdsson, K. B. Petersen, and T. Lehn-Schiøler, "Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music," in *Proceedings of the Seventh International Conference on Music Information Retrieval (ISMIR)*, 2006.

[10] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 2006.

[11] M. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, 2002.

[12] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[13] J. Ianniello, "Time delay estimation via cross-correlation in the presence of large estimation errors," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 30, no. 6, pp. 998–1003, 1982.

[14] J. Benesty, J. Chen, and Y. Huang, "A generalized MVDR spectrum," *Signal Processing Letters, IEEE*, vol. 12, no. 12, pp. 827–830, 2005.