



# **Video/Audio Networked surveillance system enhancement through Human-centered adaptive Monitoring**

**Large-scale integrating project  
Grant Agreement n°248907  
01/02/2010 – 31/07/2014**

---

**Contractual delivery date: 31 January 2011  
Actual delivery date: 11 February 2011**

## **Deliverable D4.2 First report on audio features extraction and multimodal activity modelling (v1)**

---

### **D4.2**

**Version: 2.0  
Author: TCF  
Contributors: —  
Reviewers: IDIAP, MULT  
Dissemination level: PU  
Related document(s): —  
Number of pages:42**

**Document information**

Ver.	Date	Changes	Author (partic.)
0.0	17/01/2011	Creation	F. Capman/S. Lecomte/B. Ravera (TCF)
1.0	02/02/2011	Final	F. Capman/S. Lecomte/B. Ravera (TCF)
2.0	04/02/2011	Minor changes	F. Capman/S. Lecomte/B. Ravera (TCF)

Ver.	Date	Approval/Rejection decision/comments	Author (partic.)
1.0	03/02/2011	Approved subject to minor changes	J.M. Odobez (IDIAP)
1.0	03/02/2011	Approved	C. Carincotte (MULT)

**Filename convention is defined as follow:**

**1. Project number:** VANAHEIM-FP7-248907

**2. Leading participant acronym (MULT, GTT, IDIAP ...):** xxx

**3. Type of document:**

Working Document (by default)	WD
Meeting Minutes	MM
Management Report	MR
Activity Report	AR
Deliverable	DR

**4. Distribution:**

Public (PU)	PU
Consortium restricted (CO)	CO

**5. Serial number (one letter + 2 digits corresponding to the task, deliverables or meeting number):**

Deliverables	D
Meeting	M
Report	R

**6. Revision number:**

draft	d
approved	a
version sequence (one digit)	

## Copyright

© Copyright 2010, 2014 the VANAHEIM Consortium

Consisting of:

<b>Coordinator:</b>	Multitel asbl (MULT)	Belgium
<b>Participants:</b>	GruppoTorineseTrasporti (GTT)	Italy
	Institut Dalle Molle d'Intelligence Artificielle Perceptive (IDIAP)	Switzerland
	Institut National de Recherche en Informatique et en Automatique (INRIA)	France
	Régie Autonome des Transports Parisiens (RATP)	France
	Thales Communications (TCF)	France
	Thales Italia (THALIT)	Italy
	University of Vienna (UNIVIE)	Austria

**This document may not be copied, reproduced, or modified in whole or in part for any purpose without written permission from the VANAHEIM Consortium. In addition to such written permission to copy, reproduce, or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.**

All rights reserved.

This document may change without notice.

## 1 Executive Summary

In this document we will outline proposed methods addressing audio analysis and multimodal analysis applied to automatic surveillance. This first report is only focused on audio analysis and describes the different technical options we have followed.

The first technical issue is related to features extraction and selection. Most of regular audio features have been implemented and a software library is available. The second issue was dedicated to the development of an evaluation framework. For algorithmic evaluation purpose, we have studied and implemented a generic framework for performance evaluation using audio signals recorded in test sites (Metro of Torino) and also audio signals extracted from professional databases.

Some algorithmic development has been carried out during this first year of the project. The main technical options we decided to follow are based on unsupervised learning. In order not to dedicate our audio surveillance system to specific abnormal audio events, we preferred to drive our training steps by normal ambience modelling. A GMM-based solution (Gaussian Mixture Model) has been adapted to this aim, and a One-Class SVM-based solution (Support Vector Machine) has been studied and evaluated. Finally, and based on promising video surveillance studies, a PLSA (Probabilistic Latent Semantic Analysis) based content analysis system has been also investigated. The presented document deals with the outcomes of the first year, and the proposed solutions need to be further improved and studied before integration inside the final VANAHEIM multimodal surveillance system.

## Table of contents

<b>1</b>	<b>EXECUTIVE SUMMARY .....</b>	<b>4</b>
<b>2</b>	<b>INTRODUCTION.....</b>	<b>7</b>
<b>3</b>	<b>AUDIO FEATURES EXTRACTION .....</b>	<b>9</b>
3.1	IMPLEMENTED ACOUSTIC FEATURES .....	9
3.2	AUDIO FEATURE EXTRACTION SOFTWARE TOOL.....	10
3.2.1	<i>Configuration file .....</i>	<i>11</i>
3.2.2	<i>Features declaration file .....</i>	<i>12</i>
<b>4</b>	<b>AUDIO FEATURES SELECTION.....</b>	<b>14</b>
4.1	OVERVIEW OF FEATURE SELECTION.....	14
4.2	STRATEGIES.....	15
4.3	PARADIGMS AND CRITERIA .....	16
4.4	STOPPING CRITERIA.....	17
<b>5</b>	<b>METHODOLOGY FOR ABNORMAL AUDIO SEQUENCE GENERATION .....</b>	<b>17</b>
5.1	THE WEIGHTED MEASURE OF SNR.....	18
5.2	DISCUSSION ON MEASURING EVENTS' SNR IN AUDIO SURVEILLANCE SIGNALS .....	18
5.3	AUDIO FOR SURVEILLANCE SIMULATION FRAMEWORK.....	21
<b>6</b>	<b>UNSUPERVISED ABNORMAL AUDIO DETECTION.....</b>	<b>23</b>
6.1	GMM-BASED SYSTEM.....	23
6.2	EVALUATION OF THE GMM-BASED SYSTEM .....	23
6.3	ONE CLASS SVM-BASED SYSTEM .....	27
<b>7</b>	<b>AUDIO ANALYSIS BASED ON PLSA.....</b>	<b>31</b>
7.1	PROBABILISTIC LATENT SEMANTIC ANALYSIS .....	31
7.2	AUDIO PLSA MODEL FORMULATION .....	32
7.3	AUDIO PLSA ANALYSIS EVALUATION ON REAL AUDIO SURVEILLANCE DATA .....	34
<b>8</b>	<b>CONCLUSION.....</b>	<b>39</b>
<b>9</b>	<b>BIBLIOGRAPHIE.....</b>	<b>40</b>
<b>10</b>	<b>GLOSSARY .....</b>	<b>42</b>

## List of Figures

Figure 1: AFE Configuration file .....	11
Figure 2: AFE file configuration (Features parameters).....	13
Figure 3- Generic Feature Selection scheme .....	15
Figure 4- Taxonomy of strategies for Feature Selection Algorithms .....	15
Figure 5- Standardize weighting curves for noise level measurement .....	19
Figure 6- Typical weighted and unweighted long time spectrum shape of an ambience signal .....	19
Figure 7- Empirical variations of noise measurements depending on weighting function.....	20
Figure 8- Mean weighted SNR variation from flat measurements depending on event type.....	20
Figure 9- Simulation flowchart .....	22
Figure 10: DET curve calculated on the complete set of abnormal events for different SNRs (10 dB, 15 dB, 20 dB, 25 dB, 30 dB).....	25
Figure 11: Evaluation of the proposed GMM-based system for abnormal audio event detection. ....	26
Figure 12: Evaluation of the proposed SVM-based system for abnormal audio event detection.....	29
Figure 13: SVM based DET curves.....	30
Figure 14 : PLSA Model with $N$ the number of documents and $nd$ the number of elements in the document $d$ . .....	31
Figure 15: Evaluation data.....	35
Figure 16: PLSA model training ( $K=3$ , $d=5s$ ).....	36
Figure 17: PLSA test with Speech, music and random co-occurrence matrix ( $k=3$ , $d=5s$ ).....	37
Figure 18: Ambiance changing over different time periods .....	37

## List of Tables

Table 1: Comparisons of solutions for building audio test signals.....	21
Table 2: List and number of abnormal audio events for GMM-based system evaluation.....	24
Table 3: Summary of evaluation data set for GMM-based system evaluation.....	24
Table 4: List and number of abnormal audio events for SVM-based system evaluation .....	28
Table 5: Summary of evaluation data set for SVM-based system evaluation .....	28
Table 6: Audio signals description .....	35

## 2 Introduction

The aim of this document is the presentation of studied methods addressing audio analysis and multimodal analysis applied to automatic surveillance. This first report is only focused on audio analysis and describes the different technical and scientific choices.

In the context of the VANAHEIM project, audio based surveillance thematic has been proposed. The surveillance thematic have been already addressed for many years but mainly focused on video modality. Audio modality as a significant automatic surveillance support tools is more recent. One can mention, among others, one past EC funded project on this thematic (IST CARETAKER (project, 2006-2008)) in which audio analysis tools have been demonstrated as a pertinent tools supporting daily tasks of security operators. The main audio challenges that have to be addressed are the same, in term of functionalities, as those addressed by video based surveillance system:

- Signal Acquisition,
- Features extraction,
- Key Features selection,
- Statistical models building (Machine Learning),
- Performance evaluation
- Surveillance system deployment.

The signal acquisition provides a sampled and quantized waveform. In order to perform audio signal analysis, some set of representative acoustic features has to be extracted. During this first year, a software tool dedicated to features extraction has been developed. It includes more than 30 acoustic features and offers the possibility to add derivatives (Section 3). The multiplication of the acoustic features may lead to redundancy (features that capture almost the same information) or noise (useless features). Thus there is a necessity for taking a look at dimensionality reduction algorithms as we expect in the future to enhance our systems performances through a near-optimal selection of features among a large variety of acoustic descriptors. This thematic called feature selection has been addressed (Section 4).

We have also studied two main kinds of audio surveillance system answering real issues:

- Detection of abnormal audio event,
- Audio ambiance changes tracking.

These two tasks are crucial for surveillance operators. Audio abnormality refers to abnormal audio events that occur in the station. Audio modality in this sense should be considered as a complementary modality that can easily be coupled with video analysis.

In the context of noisy environments, such as public transportation environments (railway stations, metro stations, ...), sport stadiums or urban centres, the acoustic environment is complex, and can be viewed as a superposition of many single audio events that are considered as normal (people talking, cars honking, trains arrivals and departures, silences, etc.). It might also present temporal structures at different scales: regular train arrivals, rush hours, weekdays / weekends rotations, seasons, etc. Though it is not possible to reliably simulate these signals, it is still possible to get records of representative sequences that rarely include abnormal events. That is the reason why we decided to pay attention to normal audio ambiance modelling (unsupervised statistical modelling of normal audio ambiance) rather than building detectors dedicated to few specific abnormal events (supervised learning). We have studied two different systems based on GMM (Gaussian Mixture Model) and One-Class SVM (Support Vector Machine). These two systems are described in Section 6.

Performance evaluations are one of the most important tasks to properly characterize the developed solutions. For the project purposes, a specific evaluation process has been implemented using both real audio ambiances and simulated abnormal audio events (Section 5).

We also have studied the application of an innovative approach based on PLSA (Probabilistic Latent Semantic Analysis) to the tracking of normal audio ambiance. PLSA has been initially applied to text

analysis. This method suggests a new view of the concept of topics in written text collection analysis (document analysis). The objective is to decompose a document into a mixture of latent aspects (hidden aspects, hidden random variables or unobserved random variables) defined by a multinomial distribution over the words of the vocabulary. This model has been initially proposed for document analysis and then fruitfully adapted to image and video analysis. We have extended this existing concept to audio analysis (Section 7).

One should note that for development and evaluation, we have used audio signals recorded in Torino “XVIII Dicembre” metro station and also signals recorded during the IST FP6 CAREATKER project (Roma metro stations). This is due to technical problems (audio acquisition chain not perfectly working in Torino) only fixed on last January 2011. Based in this database and for each of the three systems, performances are presented.

### 3 Audio Features extraction

The signal acquisition provides a sampled and quantized waveform, this is our raw material. In order to be further processed, this waveform has to be analysed and the acoustical information extracted. This procedure, the so-called parameterization of the signal, consists in transforming the waveform into a series of vectors of parameters. The parameters are also called acoustic features or descriptors; we will use these terms without distinction.

We developed an Audio Feature Extraction software tool that transforms audio files into feature files containing the parametric representation. In this section, we first present the acoustic descriptors that were implemented. Then, we provide information on the software usage and configuration.

#### 3.1 Implemented acoustic features

We grouped the implemented features (see (Kim, Moreau, & Sikora, 2005)(Peeters, 2004)) into six categories. We give a short description of each feature with its declaration identifier that is used for declaring extraction (see next section). These six categories are:

- Loudness features (relatives to energy considerations),
- Time-Domain features,
- Frequency-Domain features,
- Statistical features,
- Regression features,
- Parametric features.

##### Loudness features:

- LoudnessTime: extract frame total loudness (mean instant energy) from time representation.
- LoudnessBand: extract frame loudness (mean instant energy) in a given frequency band.
- LoudnessSpec: specific loudness, extract frame loudness in more than one given band.
- LoudnessRel: relative loudness, extract the portion of energy in given bands relatively to the energy of a base band.
- LoudnessFbk: filterbank loudness, extract mean instant energy outputs from a specified filterbank (frequency scale and number of filters has to be defined).

##### Time-Domain features:

- ZeroXingRate: measures the number of time that the potion of signal crosses the zero, relatively to its length.
- EnergyEntropy: returns an information measure.
- Periodicity: find the most likely signal period into a given lag interval.
- Autocorrelation: measures signal autocorrelation into a given lag.
- XCorSeq: cross correlation sequence, returns the sequence of cross correlation coefficients.
- xCorPkness: returns a measure of significance of a detected autocorrelation peak.
- TimeDomainBurst
- AudioWaveForm: returns min and max values of the signal waveform. In the MPEG7 norms, it is recommended to use this descriptor without overlapping.

##### Frequency-Domain features:

- SpecFlatness: quantifies the flatness of spectrum to distinguish noise-like from tone-like signals.
- TonalityCoef: provides a measure of the tonal behavior of the signal.
- Brightness and BrightnessMag: returns the frequency centroid of PSD or magnitude spectrum.

- Bandwidth and BandwidthMag: quantifies the distribution of spectrum around the brightness.
- SpecCrestFactor: this is another measure of tonality of the signal.
- SpecRollOff et SpecRollOffMag: returns the frequency below which a given percentile of the PSD of magnitude spectrum distribution is concentrated.
- SpecSparsity: ratio between the  $L_\infty$  and  $L_1$  norms of the magnitude spectrum.
- SpecEntropy: this is another measure of the noise-like or tone-like behavior of the signal.
- SpecContrast: estimates the strength of spectral peaks, valleys and their differences in subbands.
- SpecFlux: the averages variation value of spectrum between two frames.

#### Statistical features:

A total of 24 statistic descriptors are implemented. Their name indicates to what feature they refer and is constituted as follows: StatDomainStatistic. “Stat” always start the descriptor identifier. Then “Domain” is one of the following:

- Time: compute statistics over the waveform.
- SpecMag: compute statistics over the magnitude spectrum.
- SpecPsd: compute statistics over the power spectrum density.
- SpecLog: compute statistics over the logarithmically compressed PSD.
- Finally is indicated the statistic with the following codes: Ave (average), Var (variance), Sdev (standard deviation), Adev (averaged deviation), Skew (skewness) and Kurt (kurtosis).

Example: StatSpecPsdVar computes the variance of the PSD spectrum.

#### Regression features:

A total of 6 regression descriptors are implemented. The regression parameters are slope and offset of an obtained linear regression over spectrum. These descriptors are declared as the statistic ones. “Stat” is replaced by “Reg” and the regression feature extracted is: RegA (slope) or RegB (offset).

Example: RegSpecLogRegA corresponds to the extraction of the slope coefficient of a linear regression over the logarithmic PSD.

#### Parametric features:

- FCC: cepstral coefficient extraction.
- LPCC: linear predictive coding coefficients extraction.
- LSF: line spectral frequencies are an alternative representation of the linear predicting coefficients.
- PLPC: perceptual linear prediction coefficients extraction.
- SPC: spectrum centroids extraction.
- RSF: ratio spectrum feature extraction.

### **3.2 Audio Feature Extraction software tool**

The incoming signal is processed on-line and frame-by-frame. A frame is defined as a set of consecutive samples, corresponding to a window length of a few tenths of milliseconds. In general, consecutive frames are overlapped for further analysis. Depending on the acoustic features to be extracted, the signal frame can be weighted by a Hamming or Hanning window. All the extracted features are concatenated into a single global vector.

The matrix composed with the extracted descriptors (cols) for each frame (rows) is the parametric representation of the signal.

One execution of the software deals with the feature extraction for one given file. Online parameters are the input file name, the features file name, and the full path to the configuration file.

AFE_main.exe audio_file_name feature_file_name datfiles\AFE_configuration.dat
---

### 3.2.1 Configuration file

The Audio Feature Extraction (AFE) tool takes as inputs uncompressed audio signals of specified format. It allows files with headers such as wav, and is able to deal with 16 or 32 bits fixed-point or 32-bits floating-point samples. The sampling frequency is also given into the configuration file.

Then, the analysis parameters can be fully determined. Size of FFT analysis, frame size, frame shift (overlap) are given in samples. The weighting window is also selected (Hamming or Hanning).

The AFE tool also implements a pre-processing step to filter the audio signal. In one hand, it is possible to perform high-pass or low-pass filtering. Filters coefficients are implemented for different cutting frequencies. In another hand, the tool offers the possibility to remove the DC component.

Output files format is either binary or ASCII. In the first case, we output a flow of 32-bits floating-point values. In the second case, the acoustic vectors are stored with 6 decimals, each line corresponds to one feature vector (one frame) and values are separated by a specified separator. The tool can also generate an output audio file reflecting the effects of filtering.

Finally, working directories, files extensions and additional configuration files are declared. This approach allows distinguishing the analysis parameters from the declaration of acoustic features to be extracted.

```

/: ===== :/
/: = AFE: generic configuration parameters ===== :/
/: ===== :/
/: 1 - INPUT FILE PARAMETERS :/
0          : header [0:none/1:wav].
1024       : header size.
1          : format, [1:short|2:long|3:float]
16000.0    : sf, sampling frequency in Hz.
/: 2 - ANALYSIS PARAMETERS :/
1024       : fft_size, fft size in samples.
320        : frame_size, frame size in samples.
16         : frame_shift, frame shift in samples.
0          : win_type, wola analysis/synthesis window [0:Hamming|1:Hanning]
/: 3 - PREFILLTERING :/
0          : hp_flag, high-pass filter flag [0:off | 1:on].
300        : hp_type, high-pass filter type [50,100,150,200,250,300].
0          : lp_flag, low-pass filter flag [0:off | 1:on].
7000       : lp_type, low-pass filter type [7000].
/: 4 - OUTPUT FILES PARAMETERS :/
0          : bin or txt output for feature extraction [0:txt|1:bin]
2          : separator for txt output [0: |1:, |2: \t|3:,\t]
0          : vectorial features have their own file [0:off|1:on]
/: 5 - DIRECTORIES :/
.\example\ : source directory
.\example\ : features directory
/: 6 - FILES EXTENSIONS :/
.S1        : input extension
.bin       : features extension
/: 7 - LIST OF FEATURES :/
.\cnfg\list.dat
/: ===== :/

```

Figure 1: AFE Configuration file

### 3.2.2 Features declaration file

This file (see example in Figure 2) lists the features to be extracted and the parameters of extraction. It is possible to add comments using “/:” symbol. No line should be left blank; this would duplicate the previous descriptor. It is possible to directly access to analysis parameters when declaring descriptors: -1 = frame size, -2 = frame size / 2, -3 = frame shift. We now present the three ways to declare a feature.

#### Using an external file:

On a single line, specify:

- The descriptor identifier (as mentioned in previous section),
- The activation flag,
- A name for the feature
- The path to the declaration file (starting with “file:”).

```
LoudnessFbk 0 4LinFbkEnergies file:descriptors/4linfbk.txt
```

The declaration file always starts with a comment line. Then follow the configuration elements, one per line with possible comments after each parameter. Other comments can be added at the end of the file such as references.

```
/: Extraction of energy outputs from a 4 linear filterbankin dB :/
4      : num_of_filters
1      : dB_flag
0      : type of filter [0:lin | 1:me1 | 2:bark | 3:warp]
0.6    : alpha coefficient (only used for warp scale)
```

Even if all the parameters are not used, they must appear in the declaration (such as the alpha coefficient in the above example).

#### Without using an external file:

The first way for declaring a feature without external file is to directly include this file content after the declaration and replace the path to file by “likeinfile:” keyword followed by the number of parameters. In this case the first comment line is unnecessary.

```
LoudnessFbk 0 4LinFbkEnergies likeinfile:4
4          : num of filters in bank
0          : dB_flag
0          : type of filter bank
0.6        : alpha coefficient (warp-freq)
```

Finally, it is also possible to declare an acoustic feature in a compact form writing all the parameters directly of the declaration line.

```
LoudnessFbk 0 4LinFbkEnergies 4,0,0,0.6
```

```

/: ===== :/
/: Type      Active  Name           ConfigFile    : comments    :/
/: ===== :/
LoudnessTime 0      TmpLoudness    1
LoudnessBand 1      TotalEnergy    1,0,8000
LoudnessSpec 0      VoiceBandLdness 1,1,300,4000
LoudnessRel 1      Voice2WhiteRatio 1,0,0,8000,300,4000
LoudnessFbk 0      4LinFbkEnergies likeinfile:4
    4      : num of filters in bank
    0      : dB_flag (for energies)
    0      : type of filter bank
    0.6    : alpha param (warp-freq)
ZeroXingRate 1      ZCR            0,-1
RegSpecLogRegA 1      RegD1          0,-1
TonalityCoef 1      TC             1,-35,0,8000
Brightness 1      Brightness     0,8000
BrightnessMag 1      BrightnessMag 0,8000
Bandwidth 1      BDW            0,8000
SPC 1      SPC            likeinfile:7
    1.0e-03 : split parameter for SPC (SPC epsilon)
    1.0e-05 : distortion threshold for SPC
    20      : maximum number of iteration for SPC computation
    5      : SPC : number of centroids
    1      : SPC mode (0:power|1:mag|2:log|3:root3)
    0      : output (0:c | 1:l | 2:r | 3:m)
    1      : dB_flag for SPCm
/: ===== :/

```

Figure 2: AFE file configuration (Features parameters)

## 4 Audio Features selection

Induction problems, such as building a detector or a classifier are always affected by the dimensionality of the data (Theodoridis & Koutroumbas, 2009)(Dangauthier, 2007). Indeed, the more the representation space has degrees of liberty, the more learning a concept will be demanding in time and resources. Moreover, the multiplication of the descriptors may lead to redundancy (using descriptors that capture almost the same information) or noise (using useless descriptors). Thus there is a necessity for taking a look at dimensionality reduction algorithms ((Blum, 1997)(Dash & Liu, 1997)(Guyon & Elisseeff, 2003)(Hall, 1999)(Liu & Motoda, 2008)(Theodoridis & Koutroumbas, 2009)) as we expect in the future to enhance our systems performances using a large variety of acoustic descriptors (see section 3).

Dimensionality reduction can be divided into two families of algorithms. In one hand, feature<sup>1</sup> extraction aims to capture the useful information from every descriptor by building new parameters, which are combinations (most of the time linear but not only) of the features from the original set of descriptors. Geometrically, this is to find projection axes in the descriptors' space. This approach is inconvenient for several reasons, such as the necessity to extract the whole set of features and lack of interpretability when combining heterogeneous descriptors. In another hand, feature selection aims to select a subset of descriptors from the original set. This is equivalent to divide the descriptors' space into three subspaces: signal space, redundant signal space, and noise space; these spaces define a partition of the original space.

Feature selection approaches can be powerful for visualization as it needs a drastic reduction to 2 or 3 dimensions, but it is not the purpose of this section. Here we consider reduction from hundreds to dozens of dimensions. We also expect a reduction in term of number of extracted descriptors. Thus, we will introduce the taxonomy of feature selection algorithms that will be used in future works for audio surveillance systems enhancement.

### 4.1 Overview of feature selection

Feature selection is a pre-processing of the representation (or space of descriptors). We insure then to access the features (selected descriptors) and eventually their control in order to understand the selection and gain expertise. We now present a generic framework for feature selection.

The first step in feature selection process is to generate a subset of descriptors. It consists in the process of a search heuristic, where iteration proposes a candidate for evaluation. Two keys are to consider: the starting point and the search strategy. The starting point might be a full or empty subset, but it also can be user defined from expertise or previous results.

The second step is the evaluation of the selected subset. This is done by the mean of a criterion, which gives a measure of utility of the descriptors (or group of descriptors) to a given concept (ex. information measure) or to a given task (ex.: detection results). For ease, we will consider that a criterion value always increase with the utility of a descriptor or subset of descriptors. The result from evaluation might drive the generation process by constructing subsets taking into account the bests subsets already known. This generation-evaluation procedure is iteratively repeated until either we complete the search (all combinations are evaluated) or a stopping criterion has been raised (ex.: convergence or threshold). Figure 3 summarizes the overall feature selection concept.

---

<sup>1</sup> In the scope of dimensionality reduction, *feature* or *parameter* is equivalent to *descriptor*. We prefer using the term descriptor (acoustic descriptors) but to be faithful to the state of the art terminology, we may sometimes use feature or parameter in this section. Do not mistake feature with feature space in the context of kernel machines (space of projection through a kernel function).

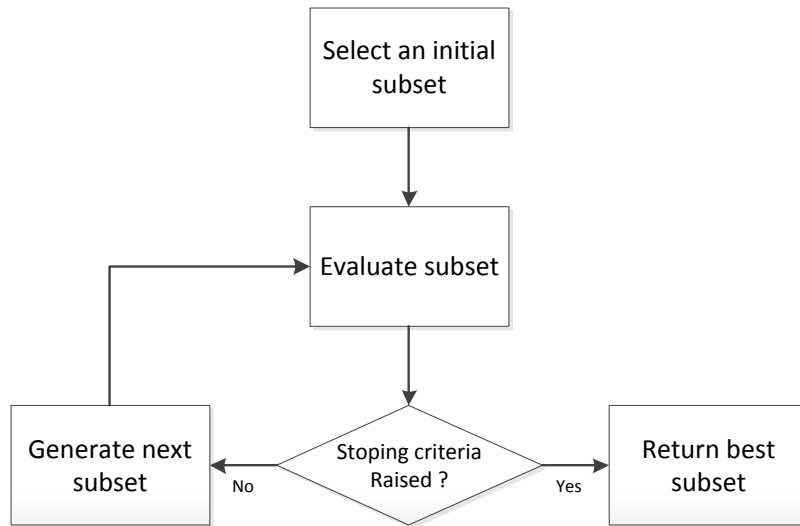


Figure 3- Generic Feature Selection scheme

## 4.2 Strategies

Basically, we distinguish strategies that are optimal, i.e. that are systematically leading to the best possible result from suboptimal ones, that tend to get one of the best results but not surely the best one. Figure 4 gives a simple taxonomy for feature selection search strategies; a short description and examples of each strategy follows.

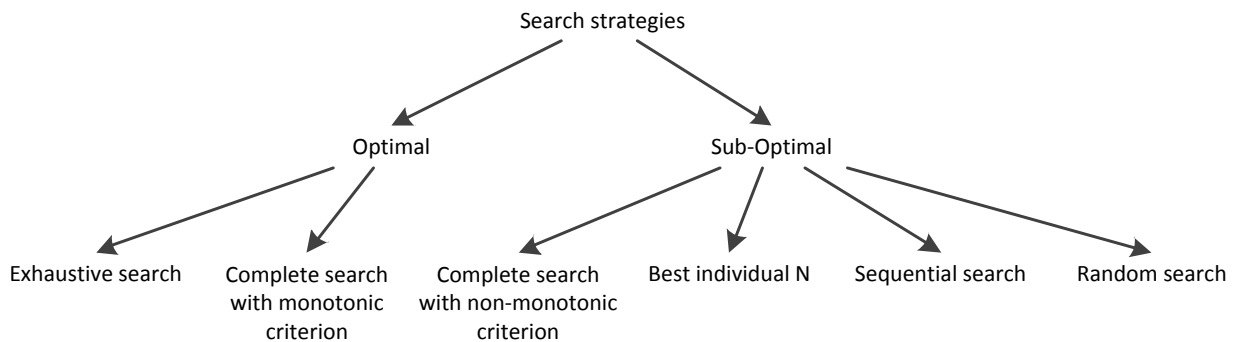


Figure 4- Taxonomy of strategies for Feature Selection Algorithms

### Exhaustive search:

This approach considers all the possible combinations. Unfortunately it is time consuming. Let  $N \in$  be the number of descriptors in the original representation space. There are a total of  $2^N$  possible combinations of such descriptors,  $C_N^K = N!/((N - K)!K!)$  if only considering the subsets of  $K \in$  descriptors. For instance, the choice of 10 features from 50 represents 10 billions of combinations to be evaluated. From this consideration, it is clear that performing an exhaustive search is prohibited once we raise a reasonable number of descriptors.

### Complete search:

Complete search means that we build a combination generation procedure that eventually might go through all possibilities. Such approaches broadly consist in trees where each node corresponds to adding (or removing) a descriptor that was not already selected (or not selected) in the path from the top node (initialization node). Then if the criterion used for evaluation is monotonic, it is possible to prune branches of the trees without loss of optimality. The same procedure can be used with non-monotonic criterion. Thus, when pruning a branch, it is possible to eliminate the one containing the best subset. This approach gives

good results when using heuristics that allow draw-back or look-after before definitely giving out a branch. An example of such procedures is Branch and Bound.(Nakariyakul & Casasent, Adaptive Branch and Bound Algorithm for Selecting Optimal Features, 2007)(Nakariyakul, On The Suboptimal Solutions Using The Adaptive Branch and Bound Algorithm for Feature Selection, 2008)(Somol & Pudil, 2004)

#### Best Individual N (BIN):

This approach, maybe the simplest, consists in evaluating the criterion for all given descriptors. Then we keep the N bests. If this approach might give good results for eliminating descriptors belonging to noise space, this does not help for eliminating redundancy.

#### Sequential search:

Sequential approaches imply that the subsets to evaluate are iteratively constructed by either adding or removing descriptors from the selected subset. In Sequential Forward Selection (SFS), iterations consist in adding to the current subset the descriptors that increase the most the criterion. Sequential Backward Selection (SBS) is similar to SFS but rejecting descriptors from an initial subset. These approaches can be generalized by adding or rejecting more than one descriptor at each iteration (GSFS and GSBS), or combined in Plus-1-Take-away-r procedure. Using specific threshold or criteria, Sequential Forward Floating Selection and Sequential Backward Floating Selection do not fix in advance the number of descriptors to add or remove. Additional heuristics can also be added to floating selection, this leads to Improved Forward/Backward Floating Selection (IFFS and IBFS).(Kudo & Sklansky, 2000)(Pudil, Ferri, Novovicova, & J., 1994)

#### Random search:

Finally, this search procedures aim to converge to an as good as possible solution by randomly generating subsets. The probability distributions might be fixed or iteratively improved depending on the results. Approaches based on genetic or particle algorithms belong to this category.

### **4.3 Paradigms and criteria**

Paradigms define ways to integrate the evaluation of subsets with the induction algorithm.(Das, 2001)(Kohavi & John, 1997)(Kudo & Sklansky, 2000)(Dash & Liu, 1997)

#### Filters:

The filter approach consists in evaluating descriptors or subsets of descriptors taking into account the intrinsic properties of the data, without any consideration to the mining algorithm that will be used. The corresponding popular criteria, independent from the task, are:

- Distance measures: separability, divergence, discrimination, etc.
- Information measures: difference between prior and posterior uncertainty.
- Dependency measures: correlation, similarity. That is the capacity to induce a descriptor values (including class information) from another descriptor values.
- Consistency measures: use class information and the so-called min-feature bias for selecting the minimum number of parameters that separate classes with as minimum inconsistency as the whole set. Inconsistency is defined as two instances (data points) having the same descriptors values and a different class label.

#### Wrappers:

Wrappers use criterion that are dependent to a mining algorithm. This approach uses as criterion the performances of the subset to a given classification task. Doing so, we obtain the most valuable selection, but it is to the detriment of complexity as an induction algorithm has to be trained and tested for each subset evaluation.

### Embedded:

Embedded feature selection is a property of data mining algorithms. This means that they have the ability to eliminate descriptors during their learning process. We do not take interest in such approaches as we want to reduce the number of extracted features but we can cite neural networks that might give out a descriptor.

### Hybrids:

Use the advantages of filter and wrapper paradigms. This approach uses an independent criterion for finding the best subset at a given cardinality and uses a task-dependent criterion to decide which subset to keep over all explored cardinalities.

## **4.4 Stopping criteria**

As presented in Figure 3, feature selection procedures include a stopping criterion. This determines the moment at which we consider that enough combinations were evaluated to propose a solution subset. Basically, we can list the following criteria:

- The generation procedure is done (all combinations were evaluated),
- A bound such as the number of iterations or the number of selected parameters has been raised,
- Convergence: adding (or removing) descriptors to (from) the subset does not improve the criterion more than a given threshold,
- A subset that is good enough has been found, i.e. above a given threshold of a criterion (that might be different than the evaluation criterion).

Once the procedure is stopped, the feature selection algorithm returns the best subset among those evaluated.

## **5 Methodology for abnormal audio sequence generation**

In the context of noisy environments, such as public transport stations, sport stadiums or urban centres, the resulting ambiances are complex, compositions of hundreds single events that are considered as normal: people talking, cars honking, trains arriving and departing, silences, etc. It might also present temporal structures at different scales: regular train arrivals, rush hours, weekdays / weekends rotations, seasons, etc. Though it is not possible to reliably simulate these signals, it is still possible to get records of representative sequences, but rarely including abnormal events. In order to qualify the robustness and the generalization capacity of an audio surveillance system, we also may need to control the events in term of SNR.

The evaluation of detection algorithms in the context of audio surveillance and more precisely for the detection of abnormal audio events is problematic. This is due to the nature of the problem itself: we do not have a set of recordings containing representative abnormal audio events both in terms of variety and quantity. This limitation in terms of evaluation data is the main motivation to develop a generation tool with an associated methodology. This tool will provide a full control for mixing real recordings of audio ambiances with recordings of abnormal audio events alone.

Most of the state-of-the-art approaches do not take into account surveillance signals specificities. In this section, we describe a complete framework for mixing abnormal events with recorded ambiances. In order to qualify the robustness and the generalization capacity of a surveillance system, we implement a control of events' SNR. Where classical methods generally measure a raw SNR over the whole bandwidth, we take a closer look to signal characteristics and propose an enhanced measurement with minimum bias. Our approach has several advantages: precise control of SNR and events position (for labelling signals), and fast generation of a large amount of audio signals, and finally, it does not need to operate in real environments to build significant test signals once representative ambience signals has been recorded.

First, we give some theoretical keys for noise level measurement. Then considering the audio signals from surveillance applications, we discuss from empirical results the use of an appropriate weighting function

when measuring SNR. Finally, based on this optimized control of mixing levels, we present the developed framework for generating databases of audio surveillance signals.

## 5.1 The weighted measure of SNR

Adapt the sound level measure to a given task is a challenge that has been addressed for a long time now. In 1933, Fletcher and Munson (Fletcher & Munson, 1933) introduced the phon as a human ear like perceptual measure (this work is still supported as ISO226:2003 (ISO, 2003)). They defined equal-level contours curves that indicate for a given physical level and a given frequency the perceived level. From these curves were derived weighting functions to measure noise level:

- A-type, based on 40 phons curve, is designed to measure noise level in quiet environments.
- B-type, based on 80 phons curve, is designed to measure noise level in normal environments.
- C-type, is based on the B-type with a weaker attenuation of lower frequencies.

Other types were also defined but deprecated. The actual IEC-61672:2003 (Commission, 2003) norm defines A, C and Z type weighting. Z only gives the cutting frequencies for uniform weighting (flat measure). These weighting functions are currently used in sonometers for measuring sound levels.

In another hand, the BBC started in 1968 (Corporation, 1968) to work on measuring noise level in audio electronic devices in order to enhance broadcasting quality. The motivation of this new approach is that previous weightings were not adapted to random noises. These researches led in 1986 to a recommendation from the CCIR as R468-4, later referenced as ITU-R 468 (Union, 1986) and were widely used, in particular in Dolby A and Dolby B standards.

## 5.2 Discussion on measuring events' SNR in audio surveillance signals

When measuring the Signal-to-Noise Ratio (SNR) of an event, we take the log-difference between ambience and event's mean energies. In real-life ambience signals, one might notice that an important part of ambience's mean energy is located in lower frequencies (see Figure 6) whereas abnormal events are energetic in full band or high-frequency. This leads to a biased evaluation of the SNR when we want to qualify an acoustic event in an audio surveillance signal. Indeed, when computing that ratio, we compare energies from different spectrum supports. To minimize this effect, we suggest using weighted spectrums in order to reinforce the so-called "utile part of signal", which is where ambience and event spectrums overlap. This approach also gives a more perceptive evaluation of SNR.

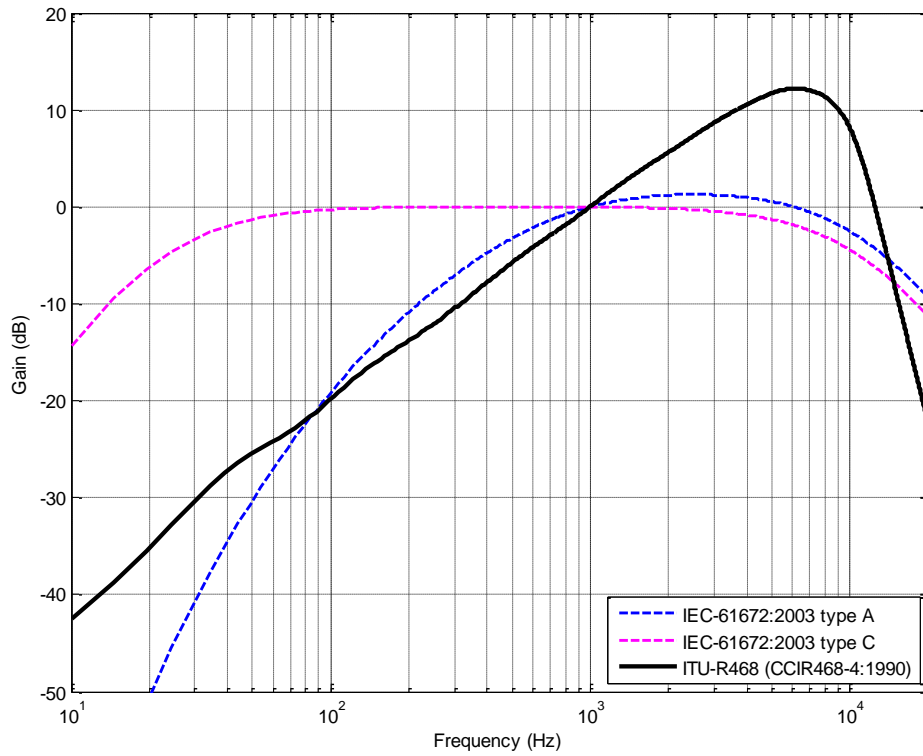


Figure 5- Standardize weighting curves for noise level measurement

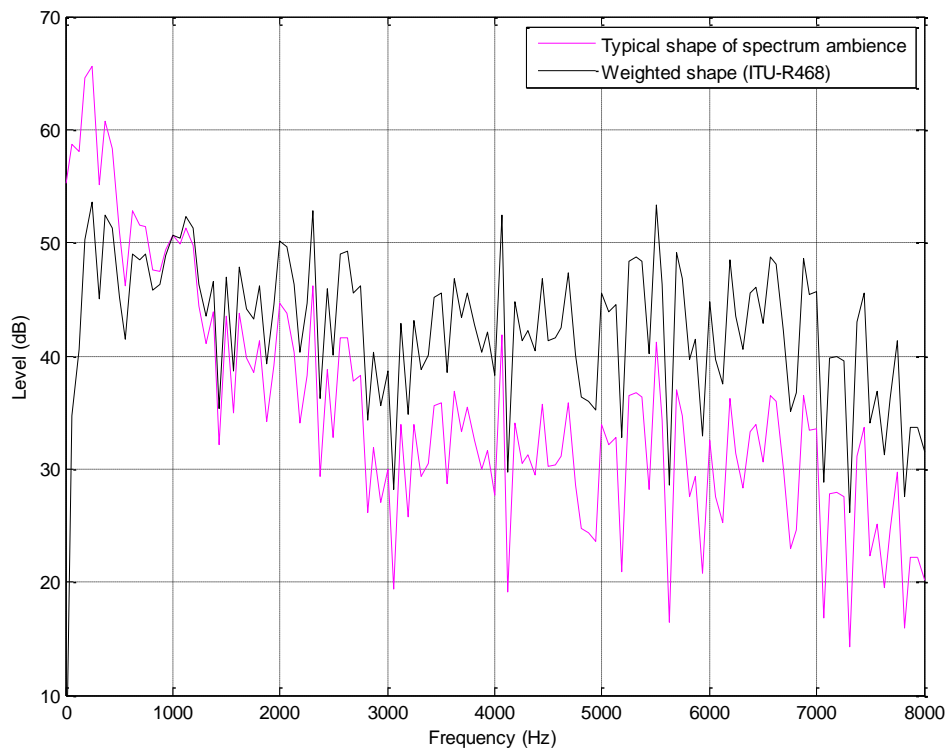


Figure 6- Typical weighted and unweighted long time spectrum shape of an ambience signal

Using each standardized weighting functions presented in the previous subsection, we measured the energies of ambiences and audio event signals. We used eighteen 10-minutes recordings from the Caretaker database

as ambiences, and 96 identified abnormal sounds from commercial databases as events: explosions, dogs, screams, gunshots, glass breakings, etc. Empirical results show that when using ITU-R468, it results in higher values of SNR. Indeed, that means that when we give a measure with that weighting, the event level appears louder than what it is in reality. When generating our signals at a target SNR using this weighting, the event’s volume will be lower than using other weightings. A-type weighting leads to a 1.9 dB overestimation and C-type leads to even slighter variations.

The overall results of SNR variation depending on the weighting are presented in Figure 7.

*Empirical results of energy measurements :*  
 - 18 ambiences: 10 minutes signals from a Rome subway station,  
 - 96 events: 1 second signals from 27 different categories.

Weighting	Flat	A-type	C-type	ITU-R468
Ambiences	58,53 dB	54,17 dB	58,38 dB	55,54 dB
Events	69,3 dB	66,78 dB	68,88 dB	70,63 dB
SNR	10,77 dB	12,61 dB	10,5 dB	15,09 dB

Figure 7- Empirical variations of noise measurements depending on weighting function

We give more detailed results in Figure 8. This figure represents, for each type of event the variation of SNR weighted measurement from a flat measure. That is equivalent to say that the zero corresponds to the unweighted measure. Events are sorted in order of increasing EER score, using the One-Class-SVM based detector presented in section 6.

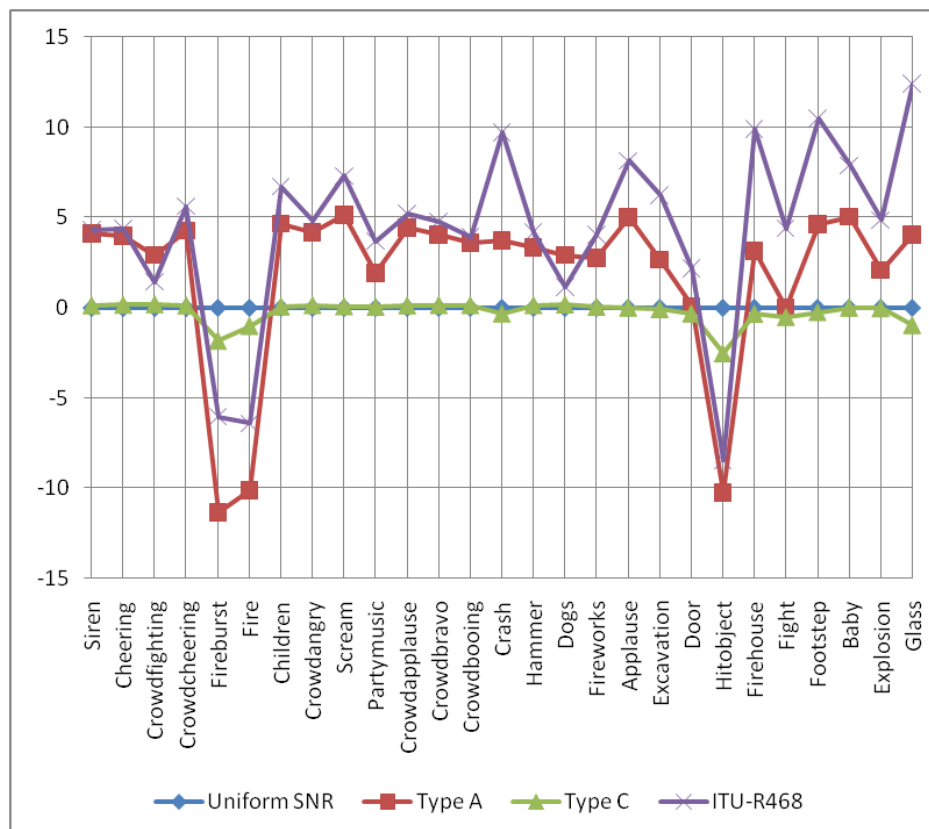


Figure 8- Mean weighted SNR variation from flat measurements depending on event type

### 5.3 Audio for surveillance simulation framework

In this subsection, we present our developments for generating evaluation database for audio surveillance systems. Indeed, as no specific database exists, we designed a framework in order to simulate adequate surveillance signals.

In the context of non-quiet environments, ambiances are complex compositions of hundreds of single events that are considered as normal: people talking, cars honking, trains arriving and departing, silences, etc. Ambiences might also present temporal structures at different scales: train arrivals, rush hours, weekdays / weekends rotations, seasons, etc. Though it is not possible to reliably simulate these signals, it is still possible to get records of representative sequences, but rarely including abnormal events.

We looked at several solutions for adding abnormal events to real ambiances. We summarize our conclusions in the following table. Additionally, “act and record” and “play and record” approaches cannot be realized during open hours of the area and we might not record really representative ambiances. It is easy to understand that we will not play explosions, fights, gunshots or screams while users are present. Then, we developed a set of tools in order to implement a “record and mix” method for generating signals.

Method	Description	Pros	Cons
Act and record	Record acted situations in real environment	Most realistic signals.	Time consuming. Some events unrealizable. Poor control of SNR.
Play and record	Play events in real environments through an audio restitution system	Events are affected by the acoustic of the environment. Large variety of events	Time-consuming. Limited control of SNR.
Record and mix	Add recorded ambiances and a selection of events.	Costless. Precise control of SNR. Precise events positions. Performs quickly. Large amount of events.	Acoustic specifications of the environment are not preserved.

Table 1: Comparisons of solutions for building audio test signals

The developed framework is presented in Figure 9. The main idea is to mix typical ambiances recorded from a place that is under surveillance with abnormal events. In order to qualify the surveillance system, we control the sound volume of events into the resulting audio file. To do so, the mean weighted energy of both ambiance and event signal are computed using a weighting function and from these values we determine the gain to apply on the event in order to raise a targeted mean SNR. The event is then duplicated and added over the whole ambiance signal at different positions.

In that context, events might be coming from commercial databases. However, the “act and record” and “play and record” solutions offer an attractive possibility for recording events. Doing so will insure that the whole simulation respects the environment acoustic and the acquisition system characteristics.

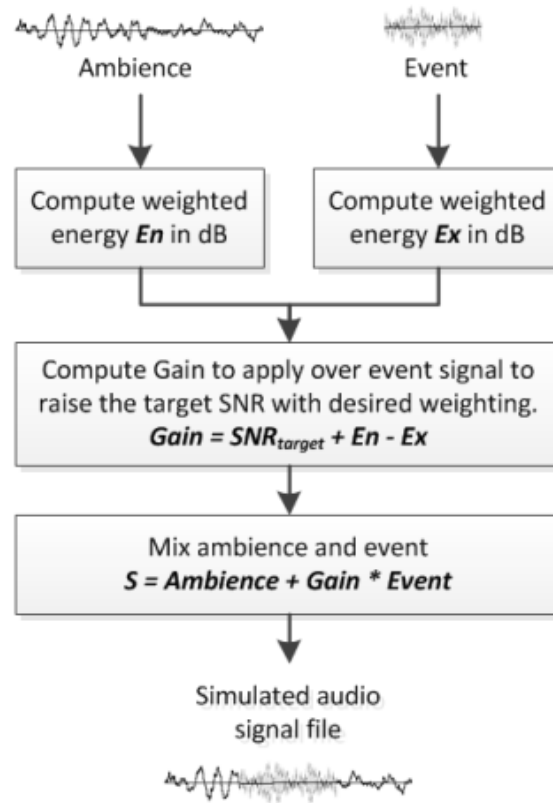


Figure 9- Simulation flowchart

## **6 Unsupervised abnormal audio detection**

### **6.1 GMM-based system**

A GMM based system has been studied and developed during this first year.

### **6.2 Evaluation of the GMM-based system**

The developed solution has been evaluated on real signals of ambience recorded in Rome metro station (CARETAKER IST FP6 project). A set of sound events has been used to simulate abnormal audio events, using the methodology described previously in section 5. The list and number of abnormal audio files are given in Table 2. The estimated duration of test signals and the various configuration parameters of the simulation chain are given in Table 3. DET curves are depicted on Figure 10 for the different tested SNR and for all the abnormal audio events, including the corresponding EER (Equal Error Rate) values. Detailed results are given for each class of abnormal audio events in Figure 11. DETware software tool developed by the NIST (National Institute of Standards and Technology) has been used for plotting, (Technology).

	Number of tests
Hammers	4
Fire	3
Fire-Burst	3
Sirens	3
Crowd-Fighting	3
Dogs	4
Crowd-Booing	3
Fire-Works	3
Crowd-Bravo	3
Explosions	4
Crowd-Appraise	3
Telephones	4
Crowd-Angry	3
Party-Music	4
Wood	9
Cheering	4
Screams	4
Children	4
Crowd-Cheering	3
Excavation	4
Applause	4
Fight	4
Doors	5
Earthquake	6
Fire-House	4
Crash	2
Foot-step	3
Glass-Debris	4
Baby	4
Hit-Objects	4
All Events	115

Table 2: List and number of abnormal audio events for GMM-based system evaluation

Number of ambience files for training	6
Number of ambience files for testing	6
Duration of each ambience file (in min.)	10
Number of SNR conditions (10,15,20,25,30 dB)	5

Duration of single audio event (in sec.)	1
Number of audio events per ambience file	50
Total duration of tested audio events (in sec.)	28750
Total duration of tested audio events (in hours)	7,99

Table 3: Summary of evaluation data set for GMM-based system evaluation

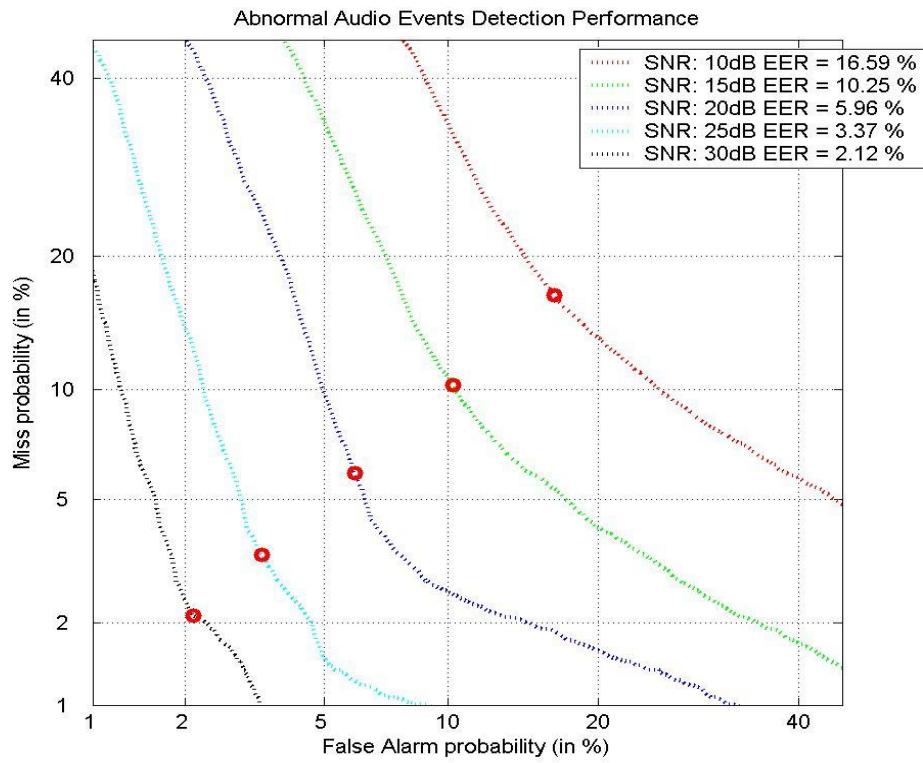


Figure 10: DET curve calculated on the complete set of abnormal events for different SNRs (10 dB, 15 dB, 20 dB, 25 dB, 30 dB)

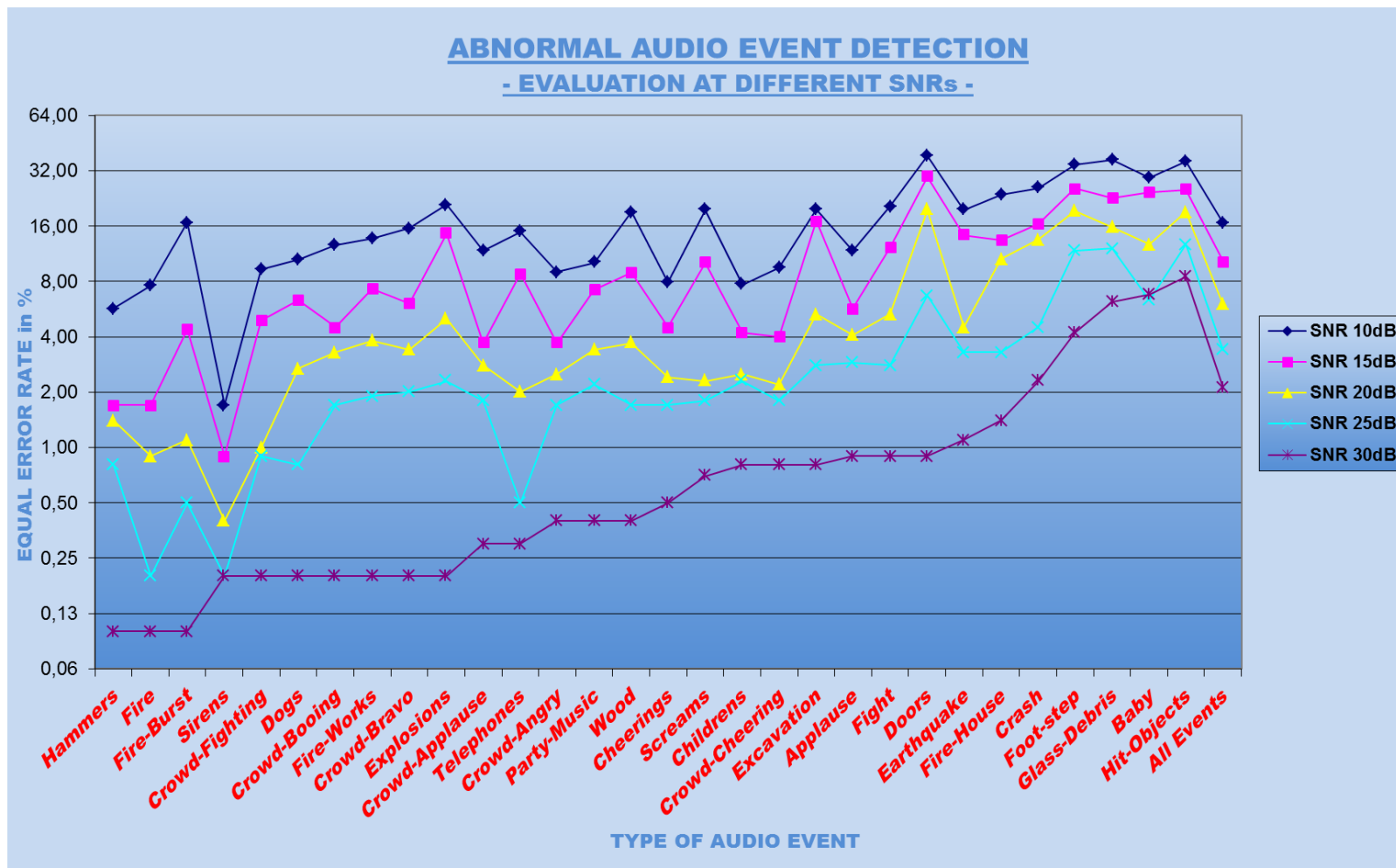


Figure 11: Evaluation of the proposed GMM-based system for abnormal audio event detection.

### 6.3 One class SVM-based system

Support Vector Machines are usually known as binary classifiers that might be extended to multi-class problems using heuristics such as One-vs.-All or One-vs.-One. In constructing such algorithm, we label points depending on the class they belong to and then perform the training confronting different classes by labels. These approaches lead to define one or more separation hyperplanes that are the frontiers between classes.

In the context of One Class Support Vector Machines (OC-SVM)(Schölkopf, Platt, Shawe-Taylor, Smola, & Williamson, 2001)(Tohmé & Lengellé, 2011), we consider that all given points from training data set belong to a single unique class, let say positive examples. Thus, instead of a separation hyperplane, our will is to define the contour of a region that encloses all the points. To stay in the framework of SVM, the basic idea is to project the data onto a hypersphere centred at the origin of a feature space using an appropriate Gaussian Radial Basis Function (RBF) kernel. Then, considering that origin as the only negative example, it becomes possible to solve the problem with traditional SVM tools.

This last algorithm has been proven to be efficient for rejecting aberrant data: outliers' rejection might be useful when considering no further analysis over the train signal when learning the detector. OC-SVM, applied in many domains including audio (Rabaoui, Davy, Rossignol, Lachiri, & Ellouze, 2007), has good generalization results and most of the time out-performs standard algorithms. Furthermore, this framework is able to model non-linear sets with ease and the hyperparameters are independent of the number of training data. As we have no a priori on neither the set of acoustic features to use nor the signals properties, OC-SVM appears to be a promising algorithm.

A SVM based system has been studied and developed during this first year.

subject to subject to As for the GMM approach, the OC-SVM solution has been evaluated on simulated signals. For an ease comparison, we now present the obtained results using equivalent tables and figures as in section 6.2.

	Number of tests
Siren	3
Cheering	4
Fireburst	3
Crowdfighting	3
Hammer	4
Children	4
Crowdcheering	3
Scream	4
Fire	3
Crowdapplause	3
Crowdangry	3
Dogs	4
Partymusic	4
Fireworks	3
Crowdbooing	3
Crash	2
Crowdbravo	3
Excavation	4
Applause	4
Footstep	3
Firehouse	4

<b>Door</b>	5
<b>Explosion</b>	4
<b>Baby</b>	4
<b>Glass</b>	4
<b>Fight</b>	4
<b>Hitobject</b>	4
<b>All Events</b>	<b>96</b>

Table 4: List and number of abnormal audio events for SVM-based system evaluation

Number of ambience files for training	6
Number of ambience files for testing	12
Duration of ambience files (in min.)	10
Number of SNR conditions (10,15,20,25,30 dB)	5

Duration of audio event (in sec.)	1
Number of audio events per ambience file	50
Total duration of tests (in sec.)	24000
Total duration of tests (in hours)	<b>6,67</b>

Table 5: Summary of evaluation data set for SVM-based system evaluation

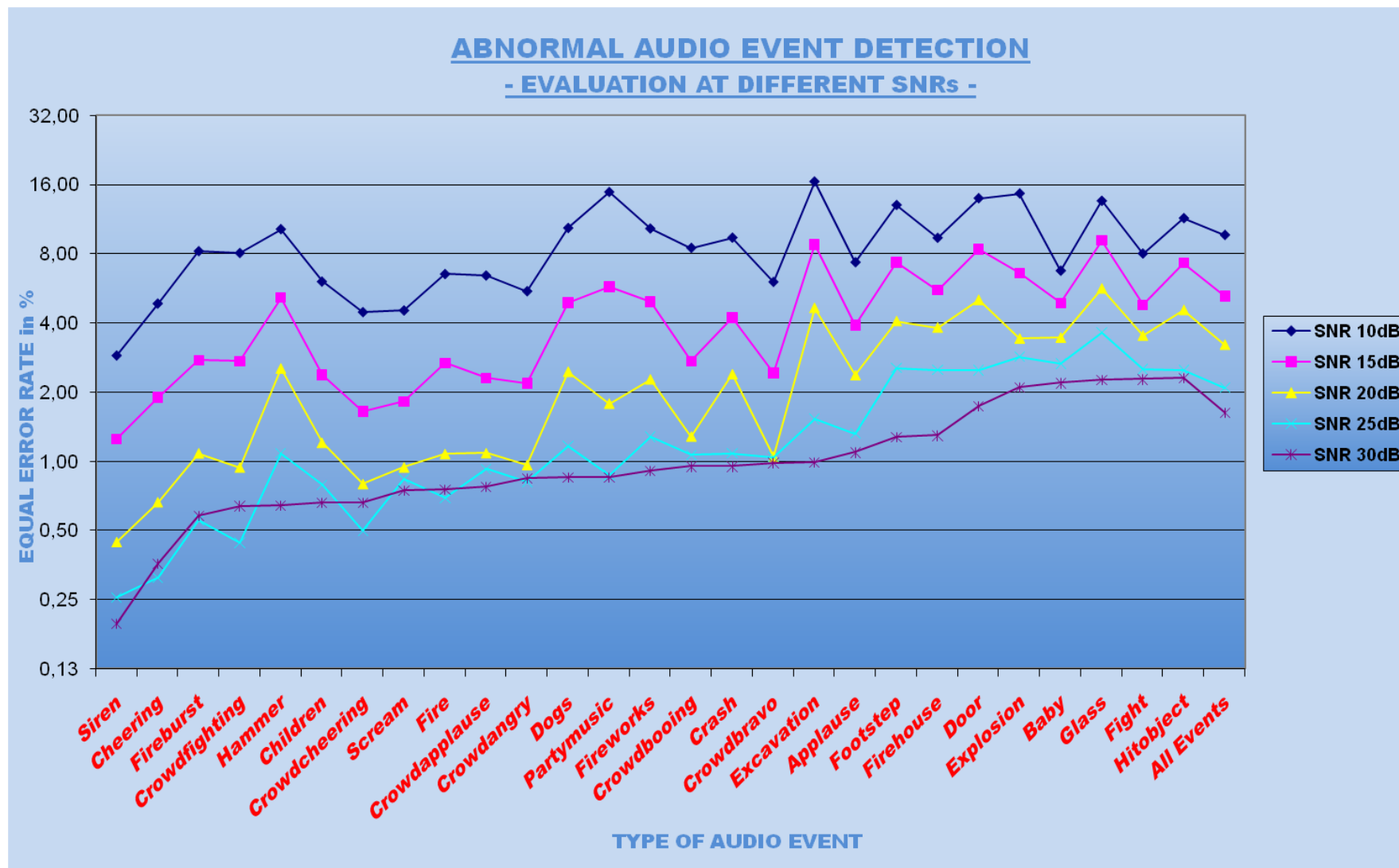


Figure 12: Evaluation of the proposed SVM-based system for abnormal audio event detection.

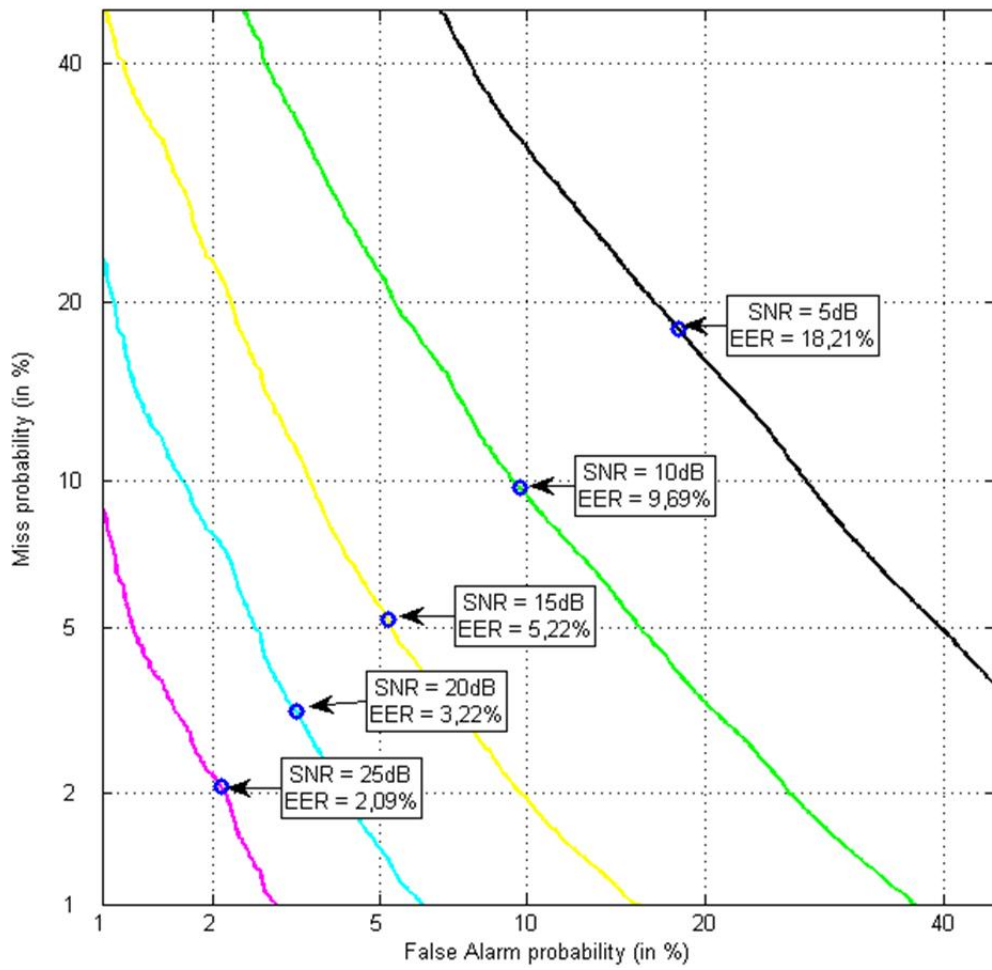


Figure 13: SVM based DET curves

## 7 Audio analysis based on PLSA

### 7.1 Probabilistic latent semantic analysis

The Probabilistic Latent Semantic Analysis model (Hofmann, 2001) (PLSA) was initially proposed by Hofmann as a probabilistic formulation of the linear algebra-based Latent Semantic Analysis (LSA) analysis (Deerwester, Dumas, Landauer, Fumas, & Harshman, 1990). This method suggests a new view of the concept of topics in text collection analysis (document analysis). The targeted goal is to decompose a document into a mixture of latent aspects (hidden aspects, hidden random variable or not observed random variable) defined by a multinomial distribution over the words of the vocabulary. Given a set of words and a PLSA model, one can generate documents as a mixture of the probability of observing a word given an aspect and the probability of observing an aspect given a document.

Documents are considered as a discrete random variable  $d_j \in D = d_1, \dots, d_N$ , where  $N$  is the number of documents. Words are represented by another discrete random variable  $w_i \in W = w_1, \dots, w_M$ , where  $M$  is the number of words. PLSA model is a statistical model that enables connections between a latent variable  $z_k \in Z = z_1, \dots, z_K$  where  $K$  is the number of latent topics with the occurrence (i.e. the number of occurrence of a word in a document) of word in a document. PLSA model is then defined as a mixture characterising a joint probability model over documents and words

$$P(w_i, d_j) = P(d_j)P(w_i|d_j) = \sum_{k=1}^K P(w_i|z_k)P(z_k|d_j)$$

Eq. 1

Under PLSA assumptions, the occurrence of a word  $w_i$  is independent of the document under consideration  $d_j$  given a topic  $z_k$ . The model defined in Eq. 1 is based on the following probability

- The probability of document  $d_j$ :  $P(d_j)$ ,
- The probability of observing the word  $w_i$  given the topic  $z_k$ :  $P(w_i|z_k)$ ,
- The probability of observing the topic  $z_k$  given a document  $d_j$ :  $P(z_k|d_j)$  (document specific conditional probability),

The PLSA model defines the conditional probability of a word in a document  $P(w_i|d_j)$  as the mixture of topic specific word distributions  $P(w_i|z_k)$ . The weights of this mixture are given by the distributions of topics in the document  $P(z_k|d_j)$ . The PLSA model is shown in Figure 14.

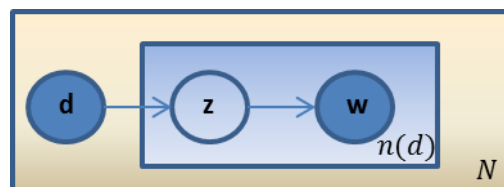


Figure 14 : PLSA Model with  $N$  the number of documents and  $n(d)$  the number of elements in the document  $d$ .

The parameters of the PLSA model are calculated using the maximum likelihood principle. Given a set of training document  $D_{train}$ , the log-likelihood of the model parameter  $P$  is given by

$$L(P|D_{train}) = \sum_{i=1}^M \sum_{j=1}^N n(w_i, d_j) \log P(w_i, d_j)$$

Eq. 2

The optimization of Eq. 2 is done using Expectation-Maximisation (EM) algorithm (Hofmann, 2001)(Dempster, Laird, & Rubin, 1977) over the training set  $D_{train}$ . As a result of this training phase, the topic distributions  $P(w_i|z_k)$  related to  $D_{train}$  are estimated. Optimizing the PLSA model means determining the topics which are common to each document of the training set and the mixture coefficients which specific to each document. The goal is estimate the PLSA model that provides high probability to the vocabulary words appears in the training corpus.

The goal of test phase is to calculate the weights  $P(z_k|d_j)$  of the trained topic for a test document  $d_{test}$  or  $P(z_k|D_{test})$  for a set of test documents  $D_{test}$ . As proposed in (Hofmann, 2001), this is done by running EM keeping the trained topic distributions  $P(w_i|z_k)$  fixed and maximising the likelihood of the words in the test document  $d_{test}$

$$L(W|d_{test}) = \sum_{i=1}^M n(w_i, d_{test}) \log \sum_{k=1}^K P(w_i|z_k)P(z_k|d_{test})$$

Eq. 3

The previous maximisation is equivalent to minimizing the Kullback-Leibler divergence between the measured empirical distributions  $P(W, D_{test})$  and the trained PLSA model.

This model has been initially proposed for document analysis and then fruitfully adapted to image and video analysis (Bosh, Zisserman, & Munoz, Scene classification using hybrid generative/discriminative approach, 2008),(Bosh, Zisserman, & Munoz, Scene classification with PLSA, 2006), (Monay, Quelhas, Gatica-Perez, & Odobez, 2009), (Varadarajan & Odobez, 2009) (classification, indexation, retrieval...). PLSA analysis has been also applied to audio document analysis. There are only few studies and mainly dedicated to music analysis (Cao, Baang, Liu, Li, & Hu, 2008), (Peng, Lu, Xiao, & J., 2009), (Smaragdis, From learningmusic to learning to separate, 2005), (Smaragdis, Shashanka, & Raj, Topic models for audio mixture analysis, 2009), (Zeng, Zhang, Li, Liang, & Zheng, 2009) (analysis, classification, indexation and recommendation).

The goal of our study is to apply this statistical framework to audio analysis and more precisely to audio surveillance applications.

## 7.2 Audio PLSA model formulation

PLSA models can be applied to discover the latent topics in audio documents. These models are based on a Bag of Words (BoW) representation obtained thank to unsupervised clustering. BoW representation does not take temporal information into account (time ordering of BoW elements). BoW is only based on a set of well adapted elements representing a data set. Bag of audio words (BoAW) are classically estimated as the cluster centroids coming from unsupervised clustering.

Suppose we have a set of audio documents  $d_j \in D = d_1, \dots, d_n$  with audio words from audio vocabulary  $w_i \in W = w_1, \dots, w_n$ . We can describe the audio documents by the BoW representations in  $m \times n$  co-occurrence matrix with  $n_{ij} = n(w_i, d_j)$ , where  $n(w_i, d_j)$  gives how often the audio word  $w_i$  occurs in an audio document  $d_j$ . The PLSA model is given as follow

1. Select an audio document  $d_j$  with probability  $P(d_j)$ ,
2. Pick a latent topic  $z_k$  with probability  $P(z_k|d_j)$ ,
3. Generate an audio word  $w_i$  with probability  $P(w_i|z_k)$ , where

$$P(w_i|d_j) = \sum_{k=1}^K P(w_i|z_k)P(z_k|d_j)$$

Eq. 4

The joint probability  $P(w_i, d_j)$  between an audio word and audio document is defined by

$$P(w_i, d_j) = P(d_j)P(w_i|d_j) = \sum_{k=1}^K P(w_i|z_k)P(z_k|d_j)$$

Eq. 5

Using Bayes rule, the joint probability is now given by

$$P(w_i, d_j) = P(d_j) \sum_{k=1}^K P(w_i|z_k)P(d_j|z_k)P(z_k)$$

Eq. 6

where  $P(d_j|z_k)$  is the probability of an audio document  $d_j$  in a topic  $z_k$ , and  $P(z_k)$  is the probability of latent topic  $z_k$ . The likelihood is the following

$$L = \sum_{i=1}^M \sum_{j=1}^N n(w_i, d_j) \log P(w_i, d_j)$$

Eq. 7

EM algorithm is used to compute optimal parameters. The E-step is given by

$$P(z_k|w_i, d_j) = \frac{P(z_k)P(w_i|z_k)P(d_j|z_k)}{\sum_{k'=1}^K P(z_{k'})P(w_i|z_{k'})P(d_j|z_{k'})}$$

Eq. 8

and the M-step is given by

$$P(z_k) = \frac{\sum_{i=1}^M \sum_{j=1}^N n(w_i, d_j)P(z_k|w_i, d_j)}{\sum_{i=1}^M \sum_{j=1}^N n(w_i, d_j)}$$

Eq. 9

$$P(w_i|z_k) = \frac{\sum_{j=1}^N n(w_i, d_j)P(z_k|w_i, d_j)}{\sum_{i'=1}^M \sum_{j=1}^N n(w_{i'}, d_j)P(z_k|w_{i'}, d_j)}$$

Eq. 10

$$P(d_j|z_k) = \frac{\sum_{i=1}^N n(w_i, d_j)P(z_k|w_i, d_j)}{\sum_{i=1}^M \sum_{j'=1}^N n(w_i, d_{j'})P(z_k|w_i, d_{j'})}$$

Eq. 11

Applying Bayes rules to Eq. 11, one can obtain the conditional probability of observing a topic  $z_k$  given a document  $d_j$  (posterior probability of a document to a topic)

$$P(z_k|d_j) = \frac{P(z_k)P(d_j|z_k)}{\sum_{k'=1}^K P(d_j|z_{k'})P(z_{k'})}$$

Eq. 12

At training time, PLSA model is optimised until convergence (maximum number of iteration or/and with stopping criterion based on log-likelihood). Eq. 8, Eq. 9, Eq. 10 and Eq. 11 are updated.

At test time, EM maximisation algorithm runs until converge (maximum number of iteration or/and with stopping criterion based on log-likelihood). Only Eq. 8 and Eq. 11 are updated because the trained topic distributions  $P(w_i|z_k)$  are kept fixed after training phase.

Finally after both of these two phases, PLSA provides for further analysis the following results

- Trained topic to word distributions  $P(w_i|z_k)$  (PLSA model),
- Posterior probability of a test document to a trained topic  $P(z_k|d_{test})$  or to a set of trained topic  $P(Z|d_{test})$ ,
- Log-likelihood of words in a test document  $L(W|d_{test})$ .

Depending on the targeted analysis, one may choose among these results the relevant ones. In the case of classification or training set analysis, posterior probability of a document to a topic  $P(z_k|d_{test})$  or to a set of trained topic  $P(Z|d_{test})$  may be jointly used with a classifier such as SVM or GMM based classifier (supervised or not depending of the targeted task). When abnormal document detection is the targeted task algorithm may be based on the log-likelihood of words in a document  $L(W|d_{test})$  tracking.

A PLSA model based analysis system has been studied and developed during this first year.

### 7.3 Audio PLSA analysis evaluation on real audio surveillance data

The first targeted task is the audio ambience changes tracking. The final goal of this task is to drive robust abnormal audio event detection such as those previously presented (GMM and SVM one class). These two systems requires to deliver good results that the trained ambience will fit, as better as possible, the test ambience. If it's not the case, some false alarms only due to ambience mismatch may occur. A best way to avoid these ambience mismatches is to check the test ambience likelihood with normal ambience models. This task can be achieved with PLSA based analysis. Assuming that durations of each significant daily ambience are known, one can build a collection of PLSA models. During test phases one can estimate if test ambiances fit well with expected models or if the previous set of trained normal ambiances is still well adapted to normal ambiances.

The data set encompasses audio signals collected in Torino XIII Dicembre metro station on 20<sup>th</sup>/21<sup>st</sup> of November 2010. Unfortunately this acquisition session was done just for setting up the audio acquisition

chain. We just collect 2.5 hours on 20<sup>th</sup> and 4 hours on 21<sup>st</sup>. The microphone is located in the entering platform (roof) close to the ticket vending machines and access turn styles (Table 6).

Date	Time period	Audio signal name
20/10/2010	16h25-16h55	Turin_20_pm_1
20/10/2010	16h55-17h25	Turin_20_pm_2
20/10/2010	17h25-15h55	Turin_20_pm_3
20/10/2010	15h55-18h25	Turin_20_pm_4
20/10/2010	18h25-18h55	Turin_20_pm_5
21/10/2010	7h00-7h30	Turin_20_am_1
21/10/2010	7h30-8h00	Turin_20_am_2
21/10/2010	8h00-8h30	Turin_20_am_3
21/10/2010	8h30-9h00	Turin_20_am_4
21/10/2010	9h00-9h30	Turin_20_am_5
21/10/2010	9h30-10h00	Turin_20_am_6
21/10/2010	10h00-10h30	Turin_20_am_7
21/10/2010	10h30-11h00	Turin_20_am_8

Table 6: Audio signals description

For PLSA test purpose we also included in the data base the following additional signals

- Speech signal (French male speaker – 30 minutes)
- Musical piece (Keith Jarret, "Concert in Koln" part one – 25 minutes),
- Uniform random audio words matrix co-occurrence.

The extracted features are 20 LFCC (without C0) extracted with 0.4 second temporal window (50% overlap). All the Torino audio features have been normalised (mean and covariance). Mean and covariance have been estimated on this whole data feature set. Since speech and music signal comes from another sources (data base or music track), they have been individually normalised (mean and covariance). Speech is plotted in red, music in blue and Turin\_20\_am\_1 in black (Figure 15).

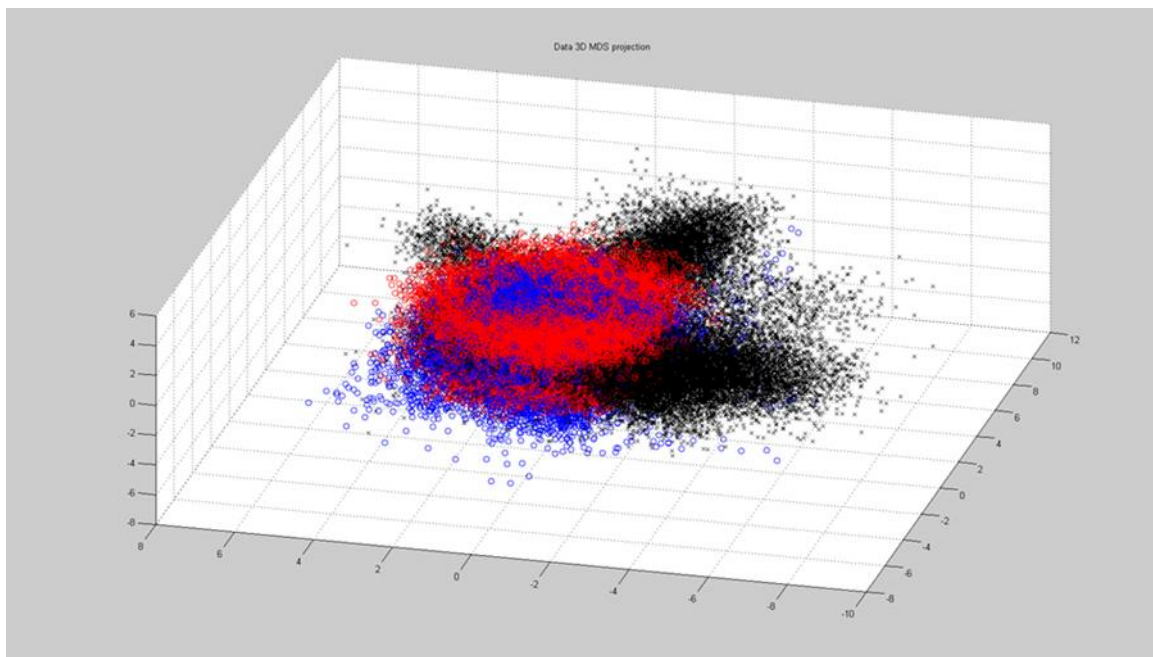


Figure 15: Evaluation data

Audio vocabulary has been trained with Turin\_20\_am\_1. PLSA model is based on this audio vocabulary. Three document sizes  $d$  have been evaluated 5 seconds, 10 seconds and 15 seconds. Several numbers of topics  $K$  from 3 to 15 have been also evaluated. Presents results of the training phase with Turin\_20\_am\_1,  $K = 3$  and  $d = 5s$ . Document log likelihood is normalised as suggested in (Varadarajan & Odobez, 2009).

The following figure (Figure 16) presents PLSA model training results with

- Trained topic to word distributions  $P(w_i|z_k)$  (PLSA model),
- Posterior probability of a trained document to a trained topic  $P(z_k|d_{test})$  or to a set of trained topic  $P(Z|d_{test})$ ,
- Log likelihood of the model during the training step
- Normalised Log-likelihood of words in a test document  $L(W|d_{test})$ .

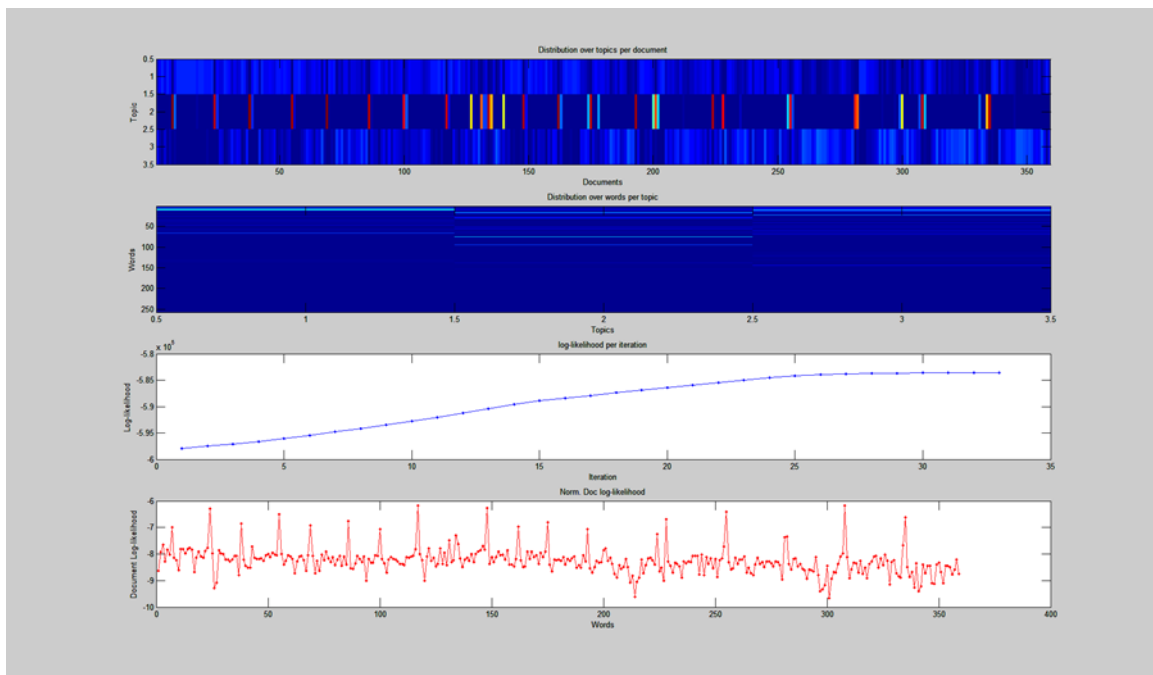


Figure 16: PLSA model training ( $K=3$ ,  $d=5s$ )

We have then evaluated this model against other 30 minutes Torino sound ambiance, speech, music and random co-occurrence matrix based data. The following figures (Figure 17 ) shows test results for speech, music and random co-occurrence matrix.

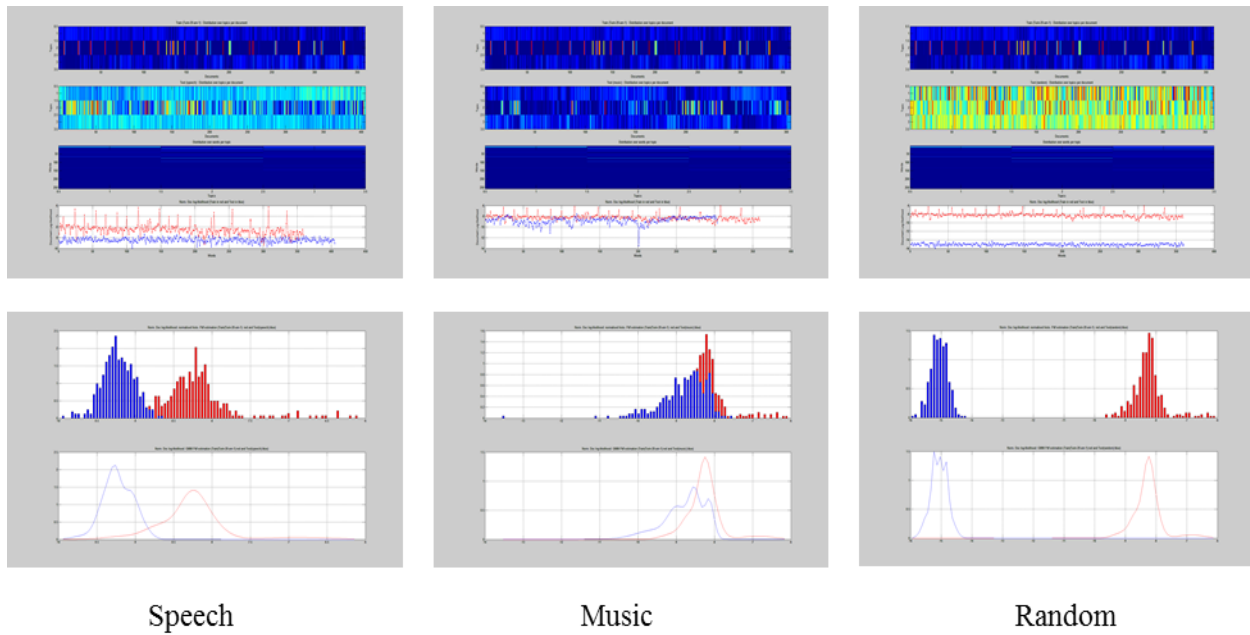


Figure 17: PLSA test with Speech, music and random co-occurrence matrix ( $k=3, d=5s$ )

For each test signals the same results as in are presented in the up part. In the bottom, normalised log likelihood of documents has been plotted (train in red and test in blue). Difference between ambiances is clearly done.

We also made some tests to evaluate training PLSA model adaptation with other ambiance related to other time periods. The results given with DET curves are presented with Figure 18. EER results are not the best to analyse our results. Low EER characterises ambiances which from the trained one. Speech is the more different. Since some short parts of training material encompass music, one can understand that music document log likelihood distribution is to close the trained one. This conclusion refers only to this particular evaluation. General conclusions about Torino metro ambiance based PLSA model and music signals cannot be inferred only with this single method (only one kind of music).

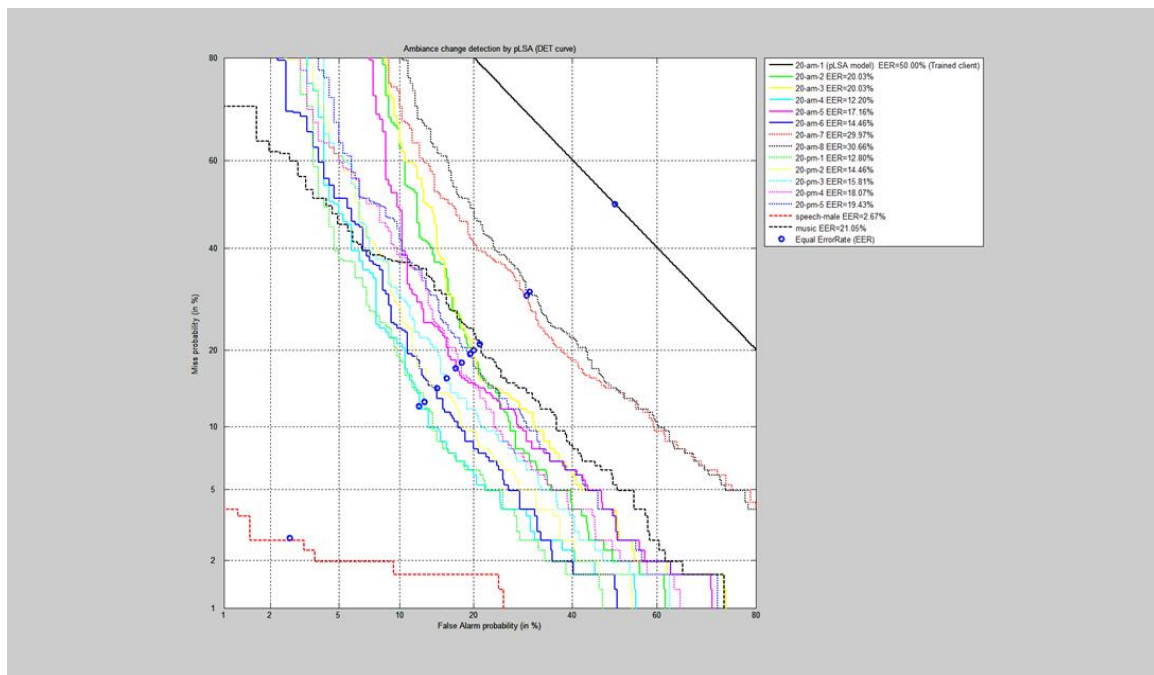


Figure 18: Ambiance changing over different time periods

PLSA model has been trained on the first kind of daily audio ambiance characterized by three main properties (7h30-8h00)

- Few metro users and mainly workers going to their offices – very quiet ambiance,
- People in charge of turn style controls not yet working (GTT security team),
- No music played inside the station.

These two facts are important because pressed workers are mainly silent and the persons in charge of security who are very noisy have not yet began their jobs.

During the day ambiance evolves. From 8h00 to 9h30, security team, students and child appear in the station with more discussions. Maximum Ambiance difference appears from 9h00 to 9h30 with morning maximum usage of the station. From 10h00 to 11h00, ambiance is another time modified and is close the trained one. Differences with afternoon are mainly due to a significant modification: music is played inside the station.

All these differences between ambiances have been highlighted thanks to PLSA analysis (see Figure 18 and Table 6 for time period details). Exhaustive tests have to be carried out before concluding about PLSA analysis added value. We need more audio signals to validate the PLSA analysis based method aiming to track ambiance changes over long time period. First results are very encouraging and will be improved during the project life.

## 8 Conclusion

A complete audio based surveillance system have been studied and developed during this first project year. A state of the art audio feature extraction tool is available but restricted to consortium usage only. All of them and at different levels are based on state of the art bricks and also used innovative solutions.

GMM based and SVM based abnormal audio event detection system, encompassing two different stages, are real time. They have been implemented using pure C ANSI and will be easily embedded in the VANAHEIM system during the first project integration phases. PLSA based audio ambiance changes tracking require more real audio ambiances for full performance evaluation. First results are very encouraging and this system will be improved during the second year of the project.

Studies and developments also include software tools supporting exhaustive evaluation of algorithms. This tool based on relevant mixture of both real audio ambiance and simulated abnormal events coming from professional audio base (entertainment post-production such as Sound FX database). It proposes metrics for SNR estimation well defined and well accepted by the audio community. The automatic test signals generation allow a quick and robust way to evaluate performances. After discussion with GTT, it has been agreed that several abnormal audio event acquisition sessions will be done in the Torino metro station. The plan is to play abnormal events inside the station, collect them with the installed 6 microphones in order to come out with simulated audio events but this time totally in phase with acoustic and electrical properties of the project test site. This real site dedicated abnormal events will be used for more significant algorithm performances evaluation.

The goals of the next year are the following

- Continue studying on GMM based and SVM based system in order to enhance current performances,
- Improve PLSA based analysis with first finalizing the audio ambiance changes tracking system and second applying this new concept to other task such as already done in video analysis,
- Begin studies on audio based stream selection.

## 9 Bibliographie

- (ISO), I. O. (2003). *Acoustics - Normal Equal-Loudness-Level Contours*. ISO Standards.
- Blum, A. L. (1997). Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence Journal*, 97 (1-2), 245-271.
- Bosh, A., Zisserman, A., & Munoz, X. (2008). Scene classification using hybrid generative/discriminative approach. *IEEE trans. Pattern Analysis and Machine Intelligence*, 30 (4), pp. 712-727.
- Bosh, A., Zisserman, A., & Munoz, X. (2006). Scene classification with PLSA. *ECCV*, 4.
- Cao, Y., Baang, S., Liu, S., Li, M., & Hu, S. (2008). Audio visual event classification via spatio temporal audio words. *IEEE International conf. on Pattern Recognition*.
- Commision, I. E. (2003). *Sound Level Meters - Part 2: Pattern Evaluation Tests*. 29-Electroacoustics, IEC.
- Corporation, B. B. (1968). *The Assessment of Noise in Audio-Frequency Circuits*. Engineering Division Research Report.
- Dangauthier, P.-C. (2007). *Fondations, Méthode et Applications de l'Apprentissage Bayésien*. PhD Thesis Report, Institut National Polytechnique de Grenoble - INRIA Rhône-Alpes, Grenoble.
- Das, S. (2001, June). Filters, Wrappers and a Boosting-Based Hybrid for Feature Selection. *Proceedings of the 18th International Conference on Machine Learning (ICML)*, 74-81.
- Dash, M., & Liu, H. (1997). Feature Selection for Classification. (Elsevier, Ed.) *Intelligent Data Analysis - An International Journal*, 1 (3), 131-156.
- Deerwester, S., Dumas, S., landauer, T., Fumas, G., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the society of information science*, 41 (6), pp. 391-407.
- Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algoorthim. *Journal of Royal Statistical Society*, 39 (1), pp. 1-38.
- Fletcher, H., & Munson, W. (1933). Loudness, Its Definition, Measurement and Calculation. *Journal of Acoustic Society of America (JASA)*, V, 82-108.
- Glass, J. R., & Zue, V. W. (1988). Multi-Level Acoustic Segmentation of Continuous Speech. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 215-218.
- Guyon, I., & Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3, 1157-1182.
- Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning*. PhD Thesis Report, The University of Waikato, Hamilton - New Zealand.
- Hofmann, T. (2001). Unsupervised learning by probabilistic semantic latent semantic analysis. *Machine learning*, 42, pp. 177-196.
- Husson, J.-L., & Laprie, Y. (1996). A New Search Algorithm in Segmentation Lattices of Speech Signals. *Fourth International Conference on Spoken Language Processing (ICSLP)*, 4, pp. 2099-2102.
- Khan, A., & Yegnanarayana, B. (2004). Latent Semantic for speaker recognition. *ICSLP*.
- Kim, H.-G., Moreau, N., & Sikora, T. (2005). *MPEG-7 Audio and Beyond - Audio Content Indexing and Retrieval*. (J. W. Ltd., Ed.)
- Kohavi, R., & John, G. H. (1997). Wrappers for Feature Subset Selection. *Artificial Intelligence Journal*, 97 (1-2), 273-324.
- Kohonen, T. (2001). *Self-Organizing Maps*.
- Kudo, M., & Sklansky, J. (2000). Comparison of Algorithms that Select Features for Pattern Classifiers. (E. Science, Ed.) *Pattern Recognition*, 33, 25-41.
- Liu, H., & Motoda, H. (2008). *Computational Methods of Feature Selection* (Taylor & Francis Group, LLC ed.). (V. K. Editor, Ed.)
- Monay, F., Quelhas, P., Gatica-Perez, D., & Odobez, J. (2009). Contextual classification of image patches with latent aspect models. *EURASIP Journal on Image and Video Processing, Special Issue on Patches in Vision*.
- Nakariyakul, S. (2008). On The Suboptimal Solutions Using The Adaptive Branch and Bound Algorithm for Feature Selection. *Proceedings of the 2008 International Conference on Wavelet Analysis and Pattern Recognition*, 1, 384-389.
- Nakariyakul, S., & Casasent, D. P. (2007). Adaptive Branch and Bound Algorithm for Selecting Optimal Features. (E. Science, Ed.) *Pattern Recognition Letters*, 28, 1415-1427.
- Pal, S. K., & Mitra, P. (2004). *Pattern Recognition Algorithms for Data Mining*. (C. & CRC, Ed.)

- Peeters, G. (2004). *A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project*. Technical Report, IRCAM, Analysis/Synthesis Team.
- Pena J.M., L. J. (1999). An empirical comparison of four initialization methods for the K-means algorithm. *Pattern Recognition Letters* , 20, pp. 1027-1040.
- Peng, Y., Lu, Z., Xiao, & J. (2009). Semantic concept annotation based on audio PLSA model. *ACM* .
- project, C. (2006-2008). CARETAKER : Content Analysis REtrieval technologies to apply Knowledge Extraction to massive Recording.
- Pudil, P., Ferri, F., Novovicova, J., & J., K. (1994). Floating Search Methods for Feature Selection with Nonmonotonic Criterion Functions. *Proceedings of the 12th IAPR International Conference on Pattern Recognition* , 2, 279-283.
- Rabaoui, A., Davy, M., Rossignol, S., Lachiri, Z., & Ellouze, N. (2007, September). Improved One-Class SVM Classifier for Sounds Classification. *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)* , 117-122.
- Rissanen, J. (1983). A Universal Prior for Integers and Estimation by Minimum Description Length. *Annals of Statistics* , 11 (2), pp. 417-431.
- Rissanen, J. (1978, September). Modelling by Shortest Data Description. *Automatica* , 14 (Issue 5), pp. 465-471.
- Rissanen, J. (1986). Stochastic Complexity and Modelling. *Annals of Statistics* , 14, pp. 1080-1100.
- Rissanen, J. (1989). Stochastic Complexity in Statistical Inquiry. *Series in Computer Science* , 15.
- Rissanen, J. (1984, July). Universal Coding, Information, Prediction, and Estimation. *IEEE Transactions on Information Theory* , pp. 629-636.
- Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A., & Williamson, R. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation* , 13 (7), 1443-1471.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys* , 34, pp. 1-47.
- Smaragdis, P. (2005). From learning music to learning to separate. *Mitsubishi Electric Research Laboratories (TR2005-134)*.
- Smaragdis, P., Shashanka, M., & Raj, B. (2009). Topic models for audio mixture analysis. *NISP workshop on Applications for topic models* .
- Somol, P., & Pudil, P. (2004). Fast Branch & Bound Algorithms for Optimal Feature Selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , 26 (7), 900-912.
- Technology, N. I. *DET-Curve Plotting Software*. [available on-line: <http://www.itl.nist.gov/iad/mig/tools/>], NIST Information Technology Laboratory.
- Tenmoto, H., Kudo, M., & Shimbo, M. (1998). MDL-Based Criterion Selection of the Number of Components in Mixture Models for Pattern Classification. *Advances in Pattern Recognition, Springer* , pp. 831-836.
- Theodoridis, S., & Koutroumbas, K. (2009). *Pattern Recognition Fourth Edition*. (E. Inc., Ed.)
- Tohmé, M., & Lengellé, R. (2011). Maximum Margin One-Class Support Vector Machines for Multiclass Problems. (E. Science, Ed.) *Submitted to Pattern Recognition Letters* .
- Union, I. T. (1986). *Measurement of Audio-Frequency Noise Voltage Level in Sound Broadcasting*. Recommendation, ITU, Broadcasting Service.
- Varadarajan, J., & Odobez, J. (2009). Topic models for scene analysis and abnormality detection. *ICCV-VS* .
- Xiong, Z., Radhakrishnan, R., Divakaran, A., & Huang, T. S. (2004). Effective and Efficient Sports Highlights Extraction using the Minimum Description Length Criterion in Selecting GMM Structures. *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)* , pp. 1947-1950.
- Zeng, Z., Zhang, S., Li, H., Liang, W., & Zheng, H. (2009). A novel approach to musical genre classification using probabilistic latent semantic model. *ICME* .

## 10 Glossary

AFE	Acoustic Feature Extraction
BIC	Bayesian Information Criterion
BoW	Bag of Words
BoAW	Bag of Audio Words
EER	Equal Error Rate
EM	Expectation-Maximisation algorithm
GMM	Gaussian Mixture Model
LFCC	Linear Frequency Cepstral Coefficients
MDL	Minimum Description Length
MFCC	Mel Frequency Cepstral Coefficients
PLSA	Probabilistic Latent Semantic Analysis
SNR	Signal to Noise Ratio
SVM, OC-SVM	Support Vector Machine, One-Class Support Vector Machine