



Anomaly Detection Algorithm for Acoustics Phenomena

Thomas Barguil, Johann Laurent, Nicolas Bohelay, Dominique Heller

► To cite this version:

Thomas Barguil, Johann Laurent, Nicolas Bohelay, Dominique Heller. Anomaly Detection Algorithm for Acoustics Phenomena. The 2021 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE'21), Jul 2021, Las Vegas, United States. hal-03621200

HAL Id: hal-03621200

<https://hal.science/hal-03621200>

Submitted on 28 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Anomaly Detection Algorithm for Acoustics Phenomena

Thomas Barguil^{1,2} Johann Laurent¹ Nicolas Bohelay² Dominique Heller¹

¹ CNRS UMR6285 - Lab-STICC, Univ. Bretagne Sud, F-56100 Lorient, France

Email : {firstname.lastname}@univ-ubs.fr

² Wavetel, F-56260, Larmor-Plage, France

Abstract—The evolution of technologies in call centers towards communications via ethernet is at the origin of a certain number of perturbations. These perturbations can take different forms but the most important one is the acoustic phenomena. In this paper, we present an anomaly detection algorithm based on the One-Class Support Vector Machines (OC-SVM), for the detection of these acoustic phenomena. We are exploring different feature functions and seeking to find the best pairing with the OC-SVM to most effectively detect those acoustic problems that may pose a risk to consultants. Our experimental results show a good detection rate for amplitude levels equal or higher than -15 dB.

Index terms— acoustic phenomena, anomaly detection algorithm, OC-SVM

I. INTRODUCTION

During the past ten years, telecommunications networks have undergone major structural changes in order to move from switched and synchronous technologies (TDM) to packet-based technologies like the TCP/IP protocol, particularly in the context of voice services. Call centers have massively adopted the Voice over IP (VoIP) technology following unified communication trend (voice/data). It allows managing with flexibility, autonomy and scalability all their communication flows. A certain number of problem then appeared in conjunction with the development and implementation of this technology partially due the coexistence of VoIP with former technologies leading to mixed infrastructures (TDM/IP).

This leads two main problems which are the following :

- bad quality of telecommunications
- acoustic shocks during the communications

A bad quality or degradation of telecommunications is the most frequently encountered problem. This degradation have a minor impact on users (stress/tiredness) but they make communication difficult to understand and generate a low quality of service (QoS).

The acoustic shocks are more serious because they can cause health problems for workers in call centers. In fact, some communications are affected by severe degradation resulting in acoustic phenomena or howling that are difficult to bear for users (operator or call center customer). These health problems can vary from slight hearing loss or tinnitus to severe hearing disorders. In [1], the author details these physiological consequences of exposure to acoustic shocks.

To solve this degradation communication problems, a suppression system of these acoustic phenomena have to be

designed. Before these problems can be removed, they must be first detected. In this paper, we only focus on this detection part.

The rest of the paper is organized as follows : Section 1 presents a state of art on the howling detection. The features extraction and SVM/OC-SVM algorithms are described in sections 2 and 3 respectively. In section 4, we present an evaluation, carried out on a large database. Finally, we will conclude the paper and give some perspectives.

II. STATE OF THE ART

In the literature, there are several methods based on signal processing using a combined spectral and temporal analysis of the audio signal. As described in [2], most algorithms select frequency components having the largest magnitude to candidate howling components. Then, they use different features on these candidates in order to exhibit the temporal and spectral properties of the howling components. These properties may be the power ratio, the entire spectrum, or the persistence for a certain duration. To know if the component is a howling component or not, they compare the computed features to predefined thresholds, see [3], [4].

Signal processing methods have important flaws because of predefined detection threshold and the detection of multiple tones. Instead of declaring a fixed threshold for each parameters used, the solution is to use a method to automatically find these optimal thresholds. Another category of method use learning machine approach to solve this problems, has been found in the literature and is described in the next paragraph.

A howling detection method is presented in [5], where the author details an algorithm to detect pure and multiple tones in an audio signal. This method is based on a SVM model with the use of Hidden Markov Model (HMM), method inspired by [6]. He uses a HMM with two states (Howling, No Howling), using the output of the SVM to prevent some discontinuities in the sequence of the decisions. As input to the SVM, it uses 3 different features : Local and Global Spectral Crest Factor (LSCF and GSCF) and a novel feature, the stability frequency. The stability frequency is to emphasize small variations of amplitude between two consecutive frames X_k^t and X_k^{t-1} together with high X_k^t for each frequency sub-band, see equation in [5]. The Table I and II shows the obtained results for the proposed method. The author uses 3 different datasets *PureOnSpeech*, *PureOnMixed* and *MultiOnMixed* containing

speech or mixed signals (speech and music) and two howlings types (Pure and Multiple tones).

TABLE I
DETECTION RATE AND FALSE ALARM RATE FOR THE PROPOSED METHOD ON THREE DATASETS

Datasets	Detection Rate (%)	False Alarm Rate (%)
<i>PureOnSpeech</i>	97.80	0.5
<i>PureOnMixed</i>	92.90	1.60
<i>MultiOnMixed</i>	89.90	1.60

TABLE II
DETECTION RATE FOR DIFFERENT STR ON THE PUREONMIXED TEST SET

Signal to Tone Ratio (dB)	Detection Rate (%)
[5 ; 0]	89.36
[15 ; 10]	80.56
[25 ; 20]	72.06

The Table I shows good results for any datasets. The Table II shows the results for the dataset, *SpeechOnMixed*. We have the results of detections according to different intervals of signal to tone ratio (STR). The STR is the ratio between the frame energy of the original signal and the tone. When the STR value increases, the signal has more gain than the tone. If the STR is equal to zero then the signal and tone have equal gains. We can notice, from these results, the relation of detection performance is dependent on the gain. We decide to propose a less complex system than the method proposed in [5], allowing a better implementation in a real system. The system that we proposed is more generic since it will not be able to learn all the anomalies but only learn the "normal" signal. The acoustic problems will only be present in the evaluation database. The architecture used is also less complex because it uses only 2 stages (Features and Detection Algorithm) instead of 3 (Features, SVM and HMM).

We propose to use a method based on SVM, called One-Class SVM. An anomaly detection algorithm who uses only two types of class (Normal, Anomaly). In our case, a "Normal" signal will consist of voice and music while an "Anomaly" will be the acoustic phenomena. We started from principle that a phenomena acoustic may be considered is an anomaly because the values in frequency and amplitude are different of speech or music.

In parallel, we use signal pre-processing to highlight different features of the signal in order to improve the detection performance.

III. FEATURES EXTRACTION

There are many possibilities to represent the speech signal for the speech recognition system, as presents in [7]. We have decided to use 4 features in your study : Spectrum, Mel frequency cepstrum coefficients (MFCC), perceptual linear prediction (PLP) and relative spectral linear prediction (RASTA-PLP). The purpose of speech features is to extract different representations of the signal from the audio signal and determine the best one for detecting anomalies. We will

explain these features below. The Fig. 1 to 4 show the representation of speech with a 2 second pure tone at 1500 Hz for each feature.

A. Spectrum

The spectrum is obtained by the fast Fourier transform. This algorithm is used to transform discrete data from the time domain into the frequency domain. This representation of the signal is the starting point for all other characteristics presented later.

B. MFCC : Mel-Frequency Cepstrum Coefficients

MFCC is a method based on human auditory behavior. The MFCC used the difference in frequencies that the human ear can distinguish. The signal is expressed according to the 'Mel' scale, which is based on the perception of pitches in equidistant intervals judged by observers. This scale uses a filter whose spacing is linear for frequencies below 1000 Hz and logarithmic for frequencies above 1000 Hz. Thus for each tone with an actual frequency, f , measured in Hz, a subjective pitch is measured on a scale called the 'Mel' scale (1).

$$f_{mel} = 2595 \times \log_{10}(1 + f/700) \quad (1)$$

where f_{mel} is the subjective pitch in Mels corresponding to a frequency in Hz. A Discrete Cosine Transform (DCT) is applied to the log filter bank energies to de-correlate the energies. This last step allows to obtain the MFCC coefficients. The set of coefficients is called acoustic vectors, see Fig. 2.

C. PLP : Perceptual Linear Prediction

The Perceptual Linear Prediction (PLP) is an algorithm developed by Hermansky in 1990 [8]. The PLP method is a combination of spectral analysis and linear prediction analysis. PLP techniques use concepts from the psychophysics of hearing to compute a simple auditory spectrum.

The bank of Bark filters performs the steps of frequency warping, smoothing and decimation. Firstly, the power spectrum $P(w)$ is warped along its frequency axis w into the bark frequency Ω as follow :

$$\Omega(w) = 6 \ln\{\omega/1200\pi + [(\omega/1200\pi)^2 + 1]^{0.5}\} \quad (2)$$

where w is the angular frequency in rad/s. The resulting warped power spectrum is then convolved with the power spectrum of the simulated critical-band masking curve. In PLP, the critical-band curve is given by :

$$\Psi(\omega) \begin{cases} 0 & \text{if } \Omega < -1.3 \\ 10^{2.5(\Omega+0.5)} & \text{if } -1.3 \leq \Omega \leq -0.5 \\ 1 & \text{if } -0.5 \leq \Omega \leq 2.5 \\ 10^{-1.0(\Omega-0.5)} & \text{if } 0.5 \leq \Omega \leq 2.5 \\ 0 & \text{if } \Omega > 2.5 \end{cases} \quad (3)$$

Subsequently, a convolution is performed among the speech auditory warped spectrum and the power spectrum of the simulated critical-band masking curve. The discrete convolution of $\Psi(\omega)$ with (the even symmetric and periodic function) $P(\omega)$ yields samples of the critical-band power spectrum.

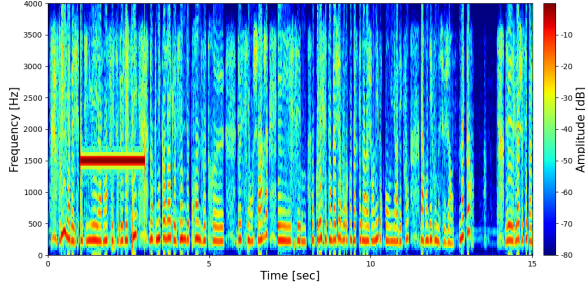


Fig. 1. Spectrogram

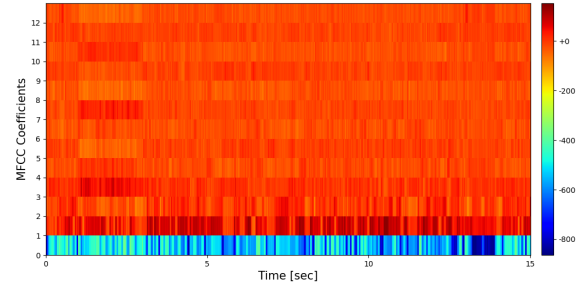


Fig. 2. MFCC

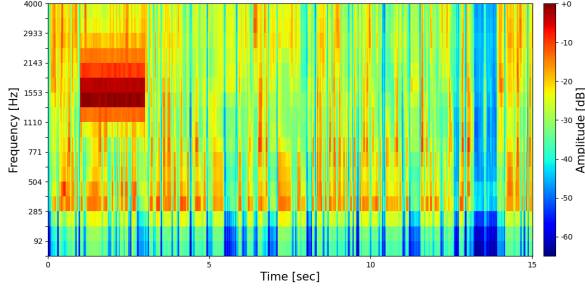


Fig. 3. PLP Spectrum

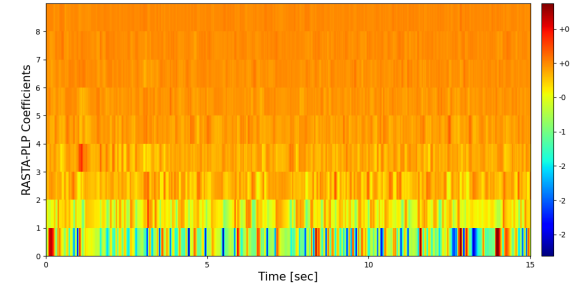


Fig. 4. RASTA-PLP Cepstrum

$$\Theta(\Omega_i) = \sum_{\Omega=-1.3}^{2.5} P(\Omega - \Omega_i) \Psi(\Omega) \quad (4)$$

The filters bank outputs Θ are scaled with an equal-loudness pre-emphasis weight giving a matrix Ξ with equalized values as follows :

$$\Xi[\Omega(\omega)] = E(\omega) \Theta[\Omega(\omega)] \quad (5)$$

The function $E(\omega)$ is an approximation of the non-equal sensitivity of human hearing at different frequencies and simulates the sensitivity of hearing at about the 40-dB level. This particular approximation is given by :

$$E(\omega) = \frac{(\omega^2 + 56.8 \times 10^6) \omega^4}{(\omega^2 + 6.3 \times 10^6)^2 \times (\omega^2 + 0.38 \times 10^9)} \quad (6)$$

Then, a cubic-root amplitude compression is applied on the equal-loudness preemphasis as follow :

$$\Phi(\Omega) = \Xi[\Omega]^{0.33} \quad (7)$$

In next step, the obtained equalized spectrum is treated with Linear Prediction (LP) in order to obtain the predictor coefficients, see Fig. 3.

D. RASTA-PLP : *RelAtive SpecTrAl* - PLP

Relative Spectral PLP (RASTA-PLP) is developed by Hermansky in 1991 [**hermansky_rasta-plp_1991**]. RASTA-PLP is an extension of PLP and the only difference, is that a band-pass filter is added at each sub band. The added filter improves

the robustness of the PLP feature as it smooths out short-term noise variation and removes constant offset in the speech signal.

The process involved in RASTA-PLP includes calculating the critical power spectrum as in PLP. Then, the time trajectory of each transformed spectral component is introduced to a band pass filter. In the next step, the filtered speech is transformed using expanding static non linear transformation. Next, an equal loudness curve adjustment and application of intensity-loudness power law, in order to mimic the human auditory system. Finally, the RASTA-PLP coefficients are computed by performing consecutively on the output of previous stage the inverse Fourier transform, the linear predictive analysis and the cepstral analysis, see Fig. 4.

We will now present our unsupervised anomaly detection algorithm, OC-SVM. This algorithm is based on the principle of the SVM classification algorithm, so a brief description will be made before presenting the OC-SVM algorithm.

IV. AN UNSUPERVISED ALGORITHM FOR ANOMALY DETECTION

Support Vector Machines (SVMs) are interesting in anomaly detection because of their ability to provide non-linear classification through a kernel function. We briefly introduce the basic concepts of SVMs then focus on OC-SVM that we adopted in this work.

A. Support Vector Machines

The concept of SVMs is to find a decision boundary between two classes. This algorithm maps the training sample

vectors onto a higher dimensional space and then determining an optimal separating hyperplane. Let us consider a training data set composed of M samples of D -dimensional feature vectors x_i , associated with a class label y_i , such that: $x_i \in R^D$, $y_i \in \{-1, +1\}$. The algorithm searches for an optimal hyperplane that separates the data with a maximal margin by solving the optimization problem :

$$\min_{w, b, \xi_i} \frac{1}{2} \|w\|^2 + C \sum \xi_i \quad (8)$$

subject to :

$$y_i(\langle w, x_i \rangle + b) \geq 1 - \xi_i, \quad (9)$$

where $\langle w, x_i \rangle$ denotes the dot product between normal vector to the hyperplane w and a feature vector x_i , b the offset and the ξ_i are the slack variables to handle the case where data are not separable. The parameter C is a constant determining the trade-off.

When a linear decision boundary is not sufficient, we can project the data onto a higher dimensional space using a transformation function $\Phi(x)$ where it is possible to define a linear hyperplane. Since the solution is formulated as a dot product, we can define the kernel function such that $k(x, x_i) = \langle \Phi(x), \Phi(x_i) \rangle$. Solving this problem, we obtain the following decision function :

$$f(x) = \text{sign}\left(\sum_{i=1}^M \alpha_i y_i k(x, x_i) + b\right), \quad (10)$$

where the α_i are the Lagrange multipliers associated with the optimization problem. The label of a new input is obtained by looking at the sign of $f(x)$.

B. One-Class Support Vector Machines

One-Class SVMs are used to separate the data of one specific class, the target class, from other data. They are trained with examples without anomaly, i.e. data points from the target class.

In the feature space F , OC-SVM method basically separates all the data points from the origin by a hyperplane and it maximizes the distance of this hyperplane to the origin. This results in a binary function which captures the region of the input space where the training data lives. Thus the function returns +1 in a “small” region (capturing the training data points) and -1 elsewhere. The quadratic programming minimization function is slightly different from the original stated :

$$\min_{w, \xi_i, \rho} \frac{\|w\|^2}{2} + \frac{1}{\nu n} \sum_{i=1}^n \xi_i - \rho \quad (11)$$

subject to :

$$(w \cdot \Phi(x)) \geq \rho - \xi_i, \quad \xi_i \geq 0 \quad (12)$$

Schölkopf et al. [9] has reformulated SVMs to take the new regularization parameter ν who is always between $[0;1]$. The

term ν is a user predefined regularisation parameter that governs the trade-off between the size of the hypersphere and the fraction of datapoints falling outside the hypersphere. ν characterizes the solution in a nice interpretable way: (1) sets an upper bound on the fraction of outliers, e.g. the training examples regarded out-of-class and (2) sets a lower bound on the number of training examples used as support vectors.

Again by using Lagrange techniques and using a kernel function for the dot-product computations, the decision function becomes:

$$\begin{aligned} f(x) &= \text{sign}((w\Phi(x)) - \rho) \\ &= \text{sign}\left(\sum_{i=1}^n \alpha_i k(x, x_i) - \rho\right) \end{aligned} \quad (13)$$

OC-SVMs thus create a hyperplane characterized by w and ρ which has maximal distance from the origin in the feature space F , hence, separating all the data points from the origin.

We choose the Radial Basis Function (RBF) kernel : $k(x, x_i) = \exp(-\gamma \|x - x_i\|^2)$, corresponding to the nonlinear kernel.

In this following section, we will discuss the evaluation stage. This section will be in 3 parts: the description of the database, the method used and the analysis of the results obtained.

V. EVALUATION

A. Datasets

Our database are generated under the same conditions that the method proposed in [5]. These sets are created from a database sampled at 8 kHz. It's the most common sampling rate used in audio communications. The database contains speech sentences with different speakers and music. The silences have been removed to keep only the speech or the music. Each file is normalized to have a mean level of -26 dB FS (Full Scale). This value is equivalent to 92 dB SPL (Sound Pressure Level) where the ear begins to be sensitive to high frequencies. In our study, our database will only be based on the *MultiOnMixed* database of the proposed method in [5].

Pure tones are generated randomly with a fundamental frequency ranging from 900 Hz to 4 kHz. This interval is chosen because it corresponds to the interval of the frequencies of the medium (900-1500Hz) and high (1500-4000Hz) sounds. These high-gain sounds can have serious consequences for hearing. For multiple tones, we considered square and multiple harmonic signals which were generated in the same way as the pure tones. The level of a howling is drawn randomly within the range of -40 to 0 dB FS. Each howling is two seconds long.

TABLE III
COMPOSITION OF THE DATASETS

Datasets	Howling type	N	Duration
<i>Evaluation</i>	Pure	417	1400min
<i>Evaluation</i>	Multi	140	484min
<i>Training</i>	None	0	13min

TABLE IV
INFERENCE FEATURES, TRAINING AND EVALUATION FOR EACH FEATURES

Steps	Spectrum		MFCC		PLP		RASTA_PLP	
	Overall (s)	W (s)	Overall (s)	W (s)	Overall (s)	W (s)	Overall (s)	W (s)
Features	74.5730	6.588e-6	100.9881	8.922e-6	169.3596	1.496e-5	886.7694	7.834e-5
Training	4366.6745	0.0547331	1190.4696	0.01492166	922.5154	0.01156305	710.54	0.008906097
Evaluation	152.9156	1.35e-5	3.5336	3.122e-7	17.6530	1.560e-6	51.2664	4.530e-6

In order to train our algorithm, we also created one set (including speech and music only) without tones. OC-SVM must train with normal samples to find the anomalies. We separated the evaluation dataset in two datasets and each one has a different howling. These datasets are summarized in table III, according to their type of howling, number of tones (N) and overall duration.

We will now present the method used for the training and the evaluation.

B. Method

The audio signal is, first, segmented into frames of 20 ms with an overlap of 10 ms. For each frame, a Fast Fourier Transform (FFT) is computed using 256 samples equivalent to 20 ms of samples at 8 kHz sampling. The result of the FFT gives directly the spectrum of the signal. For the MFCC, the signal is decomposed into 20 rectangular bands spaced in the mel scale. The PLP and Rasta-PLP decompose the signal into 18 rectangular bands spaced in the Bark scale. The OC-SVM is applied on the training set for each feature. We used the Lib-SVM library [10] as implementation of the OC-SVM. A grid-search procedure is performed to find the best OC-SVM's parameter, η , by ten folds cross-validation (CV). This approach, called k-fold CV, consists to split the training set into k smaller sets. The following procedure is followed for each of the k "folds":

- A model is trained using k-1 of the folds as training data
- the resulting model is validated on the remaining part of the data

The Fig. 5 shows a cross-validation example by 5-folds.

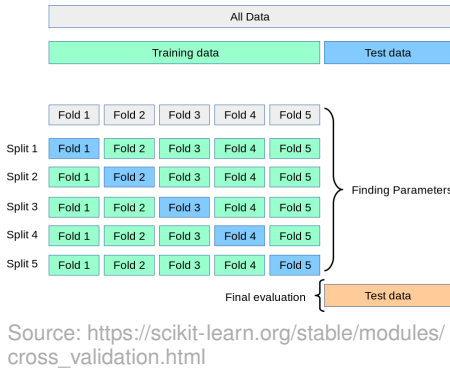


Fig. 5. 5-folds Cross-validation Example

The Table IV shows the execution time for the evaluation dataset for Features and Evaluation on your set up. The training is based on the Training dataset only. The execution

time is given in seconds by windows (W) and for the overall datasets.

The computer setup is presented in Table V. OC-SVM algorithm is coded in python 3.7.6. The learning and deployment of OC-SVM rely on the widely used Scikit-Learn, open source ML framework. For the features MFCC, we used Librosa, python package for music and audio analysis. The Features PLP and RASTA-PLP are self coded.

TABLE V
SET UP

Set up	Computer
Hardware	Intel i7-9700 8 cores at 3.0 GHz 24 GB of DDR4
Operating System	Linux Debian 10

C. Results

The results presented in Table VI, gives the accuracy rate, the false alarm rate and the anomaly detection rate for each feature. The accuracy rate corresponds to the set of labels predicted for a sample that exactly matches to the corresponding set of real label (Normal or Anomaly). The anomaly detection rate is the number of correctly predicted anomalies out of the total number of anomalies to be detected in the database. The False Alarm rate is calculated as the ratio between the number of anomalies wrongly categorized as normal and the total number of actual anomaly. To have an efficient system, the accuracy and detection rate must be as close to 100% as possible and the false alarm rate must be close to 0%.

TABLE VI
COMPARISON ANOMALY DETECTION , FALSE ALARM AND ACCURACY RATE

Features	Accuracy (%)	False Alarm (%)	Detection (%)
Spectrum	92.29	7.21	69.29
MFCC	95.81	1.21	9.62
RASTA-PLP	96.07	0.53	1.67
PLP	95.17	3.67	57.26

The features in cepstral domain (MFCC and RASTA-PLP) provide good prediction results (95-96%) and a low false alarm rate (0.5-1.2%). However, we can notice that they have a very bad anomalies detection rate. The RASTA-PLP detects only 1% of the anomalies, i.e. 5 anomalies out of the 557 anomalies generated in the database. For the MFCC, the results are better (9-10%) but not sufficient. It detects about 50 anomalies out of 557. From these results, we can conclude that these features are not effective to detect this kind of anomaly.

For the features in the spectral domain (Spectrum and PLP), they provide better results. For the accuracy rate, the PLP provides similar results as the RASTA-PLP and MFCC equal to 95%. We can observe a decrease in accuracy of 3% compared to the PLP (92%). The detection rate is better for these features : 57% for PLP and 69% for spectrum. With the spectrum, we detect 384 anomalies out of 557 against 317 for the PLP. We can observe that the increase in the detection rate leads an increase in the false rate. We have 3.67% for the PLP and twice as much for the spectrum 7.21%. The Spectrum and PLP have encouraging detection results without degrading the performance of the accuracy and false alarm rates. In the following results, we will focus on the anomaly detections for the spectrum and the PLP. We will see the behaviour of the detection rate in function of the gain for the two types of anomalies generated: Pure and Multiple. The results are shown in Fig. 6.

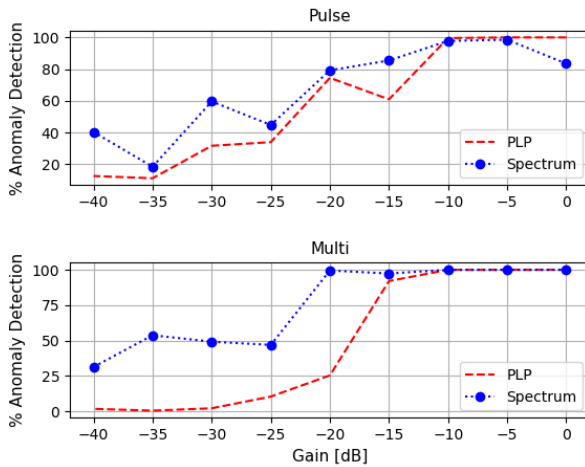


Fig. 6. Comparison of Anomaly Detection rate for Spectrum and PLP according to the gain applied on the tones (Pulse and Multiple)

Fig. 6 shows the anomaly detection rate of pure and multiple tones for the spectrum and PLP. For pure tones, the anomaly detection is very good when the anomalies is higher than -15 dB and stay good to -20 dB for PLP and spectrum. For multiple tones, we have differences between spectrum and PLP. The PLP detects all anomalies higher than -20 dB and higher than -25 dB for the spectrum. There is a significant difference between spectrum and PLP for gains below -15 dB. The spectrum detects approximately 50% of anomalies between -25 and -35 dB and 25% at -40 dB. While the PLP, the detection rate tends rapidly to zero as the gain decreases from -20 dB. Our priority is to detect 100% of the anomalies or to get as close as possible on high gains (from -15 dB). These high gains can have serious consequences for hearing. So from these results, we decide to retain the PLP feature, although the spectrum has higher detection rates than the PLP because the spectrum only detects 80% of the anomalies at 0 dB for pure tones.

VI. CONCLUSION AND PERSPECTIVES

We proposed a robust algorithm for the detection of the most dangerous acoustic phenomena for users. This method is based on an anomaly detection algorithm with a signal representation used in speech processing. Evaluation carried out on a very large database showed that our algorithm combined with the PLP is able to detect loud tones efficiently. Frequencies that have low level of gain can be less detected since their impact on the users are less dangerous.

In our future work, we will try to develop a mechanism to increase the algorithm robustness to improve the detection rate for low level frequencies. Detection of these frequencies can be used to prevent potential signal degradation. It will also be necessary to implement an online system that will be able to detect and remove these problems in real time.

REFERENCES

- [1] Alain Londero et al. "A Case of Acoustic Shock with Post-trauma Trigeminal-Autonomic Activation". English. In: *Frontiers in Neurology* 8 (2017).
- [2] Soudeh A. Khoubrouy and Issa Panahi. "A method of howling detection in presence of speech signal". en. In: *Signal Processing* 119 (Feb. 2016), pp. 153–161.
- [3] Kashyap Patel and Issa M. S. Panahi. "Efficient Real-Time Acoustic Feedback Cancellation using Adaptive Noise Injection Algorithm". In: *2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*. July 2020, pp. 972–975.
- [4] Stefan Kuehl et al. "Acoustic Howling Detection and Suppression for IP-Based Teleconference Systems". In: *Speech Communication; 13th ITG-Symposium*. Oct. 2018, pp. 1–5.
- [5] J. Flocon-Cholet et al. "A robust howling detection algorithm based on a statistical approach". In: *2014 14th International Workshop on Acoustic Signal Enhancement (IWAENC)*. Sept. 2014, pp. 65–69.
- [6] Julien Faure, Alexandre Guerin, and Claude Marro. *Method and device for detecting acoustic shocks*. en. May 2015.
- [7] Urmila Shrawankar and V. M. Thakare. "Techniques for Feature Extraction In Speech Recognition System : A Comparative Study". In: *arXiv:1305.1145 [cs]* (May 2013).
- [8] Hynek Hermansky. "Perceptual linear predictive (PLP) analysis of speech". In: *The Journal of the Acoustical Society of America* 87.4 (Apr. 1990), pp. 1738–1752.
- [9] Bernhard Schölkopf et al. "Support vector method for novelty detection". In: (Jan. 2000), pp. 582–588.
- [10] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A library for support vector machines". en. In: *ACM Transactions on Intelligent Systems and Technology* 2.3 (Apr. 2011), pp. 1–27.