Real-Time Detection of Impulsive Sounds for Audio Surveillance Systems

Faycal Ykhlef[1], Sarah Ahmed Hamada[2], Farid Ykhlef[2], Abdeladhim Derbal[1] and Djamel Bouchaffra[1]

[1] Centre de Développement des Technologies Avancées, Division ASM, Algiers, Algeria
[2] University of BLIDA 1, LATSI and FUNDAPL Laboratories, Blida, Algeria
{fykhlef, aderbal, dbouchaffra}@cdta.dz[1]
{sarah.medhamada, ykhlefarid}@gmail.com[2]

Abstract. The monitoring of dangerous audio events is very important in surveillance systems. One of the most significant phase in audio surveillance is the detection of impulsive sounds (IS). It is considered as a preprocessing stage prior to the recognition phase. We propose in this paper an indoor audio monitoring software to detect IS in real-time. It is composed of three main stages: (i) audio acquisition, (ii) preprocessing module and (iii) sound detector. We have used MEMS microphone to acquire the audio data. The preprocessing stage aims at tuning the microphone sensitivity. It is used to mask the non-desired frequency components of the environment by adding white noise. The detection of IS is conducted using a thresholding scheme based on normalized form of power sequences. The proposed prototype is running under Windows 7 on an ordinary laptop. The results we have obtained are very promising.

Keywords: Impulsive sounds detection; power; real-time audio surveillance.

1 Introduction

The security of citizens in public environments is an important issue facing all the countries of the world. Therefore, setting up efficient surveillance systems has become essential in urban environments. In addition to video data, the third generation of surveillance systems includes additional sensors to provide extra information about anomalous events. Several types of sensors can be exploited. One can mention: temperature-meters, movement detectors, infra-red sensors, seismometers, and microphones [1]. In particular, the audio data captured by the microphones can be used to track-down the dangerous events which are happening outside the range of the camera view. In addition, audio data can be useful when the video information captured by the camera do not have enough clues to identify dangerous events especially when the climatic conditions become unfavorable. Acoustic events that may be identified as indicators of dangerous situations include but are not limited to: gunshots, screams, dogs barking, car accidents, alarms and glace breaking [2]. Theoretically, the main feature that is shared by all these acoustic events is a sudden energy increase. The detection and recognition of these events is a key phase for the implementation of an efficient surveillance system. The detection step consists in identifying the special acoustical events which are happening in the environment, typically impulsive sounds (IS). On the other hand, sound recognition consists in distinguishing between the different types of impulsive waves [3]. The sound detection module has to be permanently activated to ensure continuous monitoring of environmental events. The techniques used to achieve this goal have to be non-complex, capable of performing robust detection in noisy conditions and must operate in real-time. On the other hand, sound recognition methods exploit more complex schemes which are generally based on advanced machine learning paradigms [1], [4], [5]. Once an IS has been detected, the recognition stage is evoked to identify its exact type. Basically, the issue of sound detection can be addressed in two different ways: (i) thresholding methods and (ii) detection by classification [5]. Most of the thresholding methods are based on the comparison of a significant feature with a fixed threshold. For instance, one can mention: power measures [3], Teager Energy Operator (TEO) [6], and Chi-square distribution [7]. The detection by classification uses the same scheme as for the recognition issue. In fact, it is composed of two main steps: (i) feature extraction and (ii) classification. It can be considered as a two-class problem where the positive class is the “IS” and the negative one in the “non-IS” [14]. The approach of sound detection based on thresholding is less computationally demanding than detection-by-classification [5].

In this paper, we will only focus on the detection of IS. The recognition problem will be approached in our future works. As far as we are aware, many solutions reported in the literature for IS detection are focusing on the algorithmic aspects [2], [3], [4], [7]. The performance of these methods are generally evaluated offline using local databases. Few contributions are tackling the issue of real time IS detection. We can mention the studies reported in references [5] and [6]. K. Lopatka [5] proposes a system for the recognition of threatening acoustic events using supercomputing cluster. The detection stage uses an adaptive thresholding method. The recognition is based on support vector machines. A parallel processing scheme is introduced to tackle latency, delays and online decisions. The developed solution can be regarded as nearly real-time since the time needed to recognize the acoustic events is about 0.2s. The sound detection scheme has not been evaluated separately in this study.

The detection and recognition system of acoustic events reported by R. Levorato [6] is developed in the Network-Integrated Multimedia Middleware and is operating in real-time. It uses a computer equipped with a sound card and a wired acoustical microphone. The TEO was exploited to detect impulsive events. The recognition is based on Gaussian mixture models. The entire system (detection and recognition) has been evaluated using four types of IS: gunshots, screams, broken glasses and barking dogs. The detection phase has not been evaluated separately since the main purpose of this study was the recognition of environmental sounds. Real time IS detection is an important issue in environmental sounds recognition. It can be considered as a preprocessing phase prior to the recognition stage. In fact, the elaboration of an audio surveillance system does not rely only on the efficiency of the algorithms; the software and hardware constraints have to be taken into account to achieve better performance.

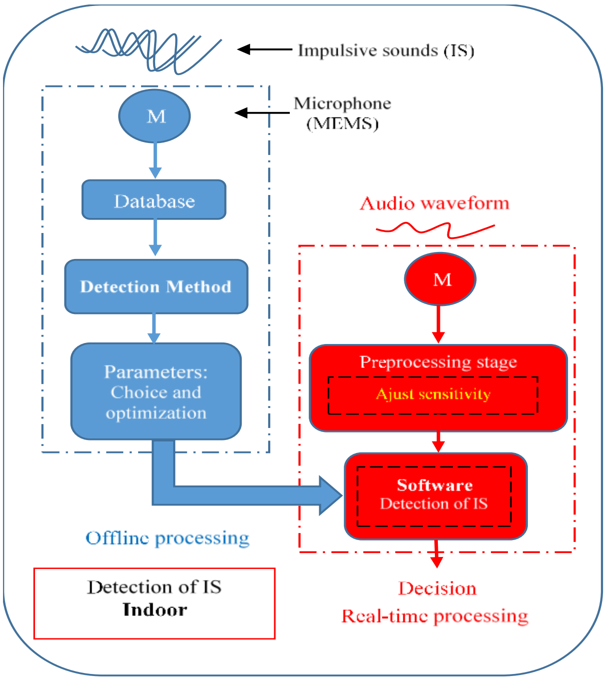


Fig 1. Design methodology and realization of the software

In addition, the sensitivity of acoustical sensors and the quality of audio data may loom large in strengthening sound detectors. Therefore, our concern in this paper is to elaborate an IS detector regardless of their exact type. We have adopted a thresholding scheme. We have used an algorithm based on normalized version of power sequences to detect sudden changes of acoustical power. A special attention was given to the microphone sensitivity in the design of our prototype. Therefore, we have proposed a preprocessing module in order to tune the microphone sensitivity by using Gaussian white noise. The software we have conceived is running under Windows 7 on a laptop equipped with Core i5 processor and 6G of RAM. The acquisition of audio data is achieved wirelessly using MEMS audio sensor. The software can detect IS under noisy conditions in an indoor environment. The detector includes: (i) offline and (ii) real-time processing (Fig.1). Our main contributions in this paper are twofold: (i) the optimization of the detector parameters for real time operation in indoor environment and (ii) the design of a preprocessing stage for microphone sensitivity tuning.

2 Design methodology

2.1 Offline processing

The goals of offline processing are fourfold: (i) the choice of an adequate audio sensor, (ii) the construction of an audio database (iii) the implementation of IS detector and (iv) the optimization of its algorithmic parameters.

Audio sensor (microphone): The audio sensor which is used to acquire data plays an important role in the detection process. Several specifications need to be taken into account. We can mention: decibel scale, frequency response, signal to noise ratio, polar response, noise level, sensitivity, dynamic range, and sound pressor level (SPL) capability [8]. Special focus should be addressed to dynamic range, sensitivity and SPL capacity in sound detection. One of the most appropriate sensors for audio monitoring are those produced by Buel & Kjaer sound and vibration [9]. Unfortunately, we were not able to purchase such microphones due to their high pricing. To cope with this problem, we have exploited another type of sensors entitled Micro-Electro-Mechanical Systems (MEMS) microphones. These sensors are usually embedded in smartphones and smart electronic devises. They are congruous for systems that compel a very high dynamic range and tight sensitivity matching [10]. As far as we are aware, the exact type of microphones which are embedded in smartphones are unluckily not provided within the smartphone technical guide. However, an overview of the specifications can be found on the following website [10]. We have used Samsung Galaxy Ace III smartphone in our experiments. It is equipped with an omni-directional microphone and offers high quality, sensitivity and maximum SPL capability (around 94dBSPL). We have exploited Wo-Mic software to transform our smartphone to be a wireless-microphone for our computer [11]. Mypublic WIFI has been used to connect the smartphone into the laptop [12].

Database: The procedure proposed to optimize the parameters that influence the IS detector requires the use of a benchmark composed of multiple sequences that contain impulsive waveforms. The recording of sounds need to be conducted using the same microphone that we plan to use in real time processing phase. We have downloaded 200 audio files of gunshots from sounddogs website [13]. The sampling frequency (Fs) of these files is 11025Hz. After that, the dynamic range of all these files has been adjusted. Silence sections have also been eliminated from the audio waveform. We have used a desktop computer equipped with high quality loudspeakers to play the audio files. The microphone (integrated in the smartphone) has been placed at a distance of 4 m from the loudspeakers in indoor environment (the surface of the room is about 30 m^2) and connected wirelessly to a laptop. The progress of the experiment is given as follows. The audio files which are saved on a desktop computer are played one after the other using MATLAB software. The silence duration between each file has been fixed to 3s. The acoustic waveforms, which are generated by loudspeakers, are simultaneously recorded using the microphone. The obtained audio sequence is saved on the hard disk of the laptop. This experimentation is repeated 3 times to obtain sequences of IS recorded respectively at 70, 80 and 90 dBSPLs. These audio sequences are separately saved as $\mathbf{seq_{(1)}}$, $\mathbf{seq_{(2)}}$ and $\mathbf{seq_{(3)}}$. The variation of sound pressure is carried out by changing the volume of the loudspeakers and measuring its SPL using a professional sound level meter. The three audio sequences are manually tagged to pinpoint the starting instants of impulsive events (Marks) (Fig. 2). These instants are saved respectively as $\mathbf{ref_{(1)}}$, $\mathbf{ref_{(2)}}$ and $\mathbf{ref_{(3)}}$ and will be exploited later to compute the detection errors.

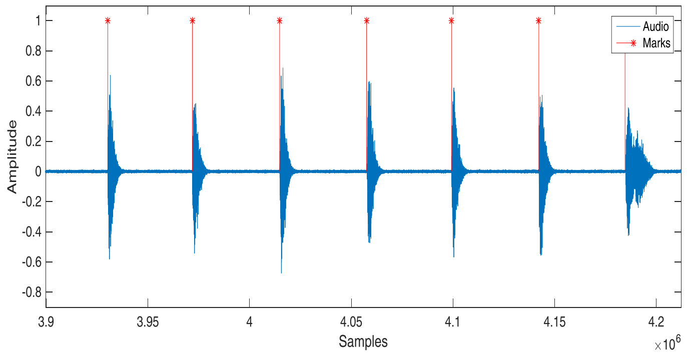


Fig. 2 Instants of beginning of impulsive events (selected section:70 dBSPL)

IS Detector: The method we have used to detect IS is originally proposed by Dufaux [3]. It is based on the power sequences of audio waveforms. The author has used this scheme as a preprocessing stage to recognize audio environmental events.

As far as we are aware, this method has never been employed before for real time detection of audio events. The tasks of detection and recognition of sounds in reference [3] were conducted offline. We have optimized the algorithmic parameters of this method to exploit it for real time detection of IS. It was implemented on MATLAB software. The detection process is summarized as follows.

Algorithm 1:

1. Computation of the k^{th} power block e(k)

$$e(k)=\frac{1}{N}\sum_{n=0}^{N-1} x^2(n+kN) \qquad (1)$$

x(n): n^{th} sample of the audio waveform which is sampled at Fs,
k: is the index of blocks. It varies from 0 to $+\infty$,
N: is the length of power blocks.

2. Framing of the power sequence $e_{win}(j/k)$

This step consists in creating a power sequence e_{win} of length L.

$$e_{win}(j) = e(i) \qquad (2)$$

The variation of ‘i’ and ‘j’ indexes are related to ‘k’ values. According to ‘k’, we can distinguish two states:

2.1. Transient state (k<L): $\quad i, j = 0 \text{ to } k-1 \qquad (3)$

2.2. Permanent state (k ≥L): $\quad i = k-L+1 \text{ to } k \qquad (4)$

$$j = 0 \text{ to } L-1 \qquad (5)$$

3. Normalization of the power sequence $e_{norm}(j)$

$$e_{norm}(j) = \frac{e_{win}(j) - \min_j\left(e_{win}(j)\right)}{\max_j\left(e_{win}(j) - \min_j\left(e_{win}(j)\right)\right)} \qquad (6)$$

$$j = 0{:}\,L-1,\ e_{norm}(j) \in [0,1]$$

4. Computation of the variance var(k)

$$var(k) = \frac{1}{L-1}\sum_{j=0}^{L-2}\{e_{norm}(j) - \bar{e}_{norm}(k)\}^2 \qquad (7)$$

$\bar{e}_{norm}(k)$ represents the mean of the first L-1 values of $e_{norm}(j)$

5. Decision: **if** $var(k) \leq Th$, **then** the sound is impulsive, **otherwise**, no special event is detected

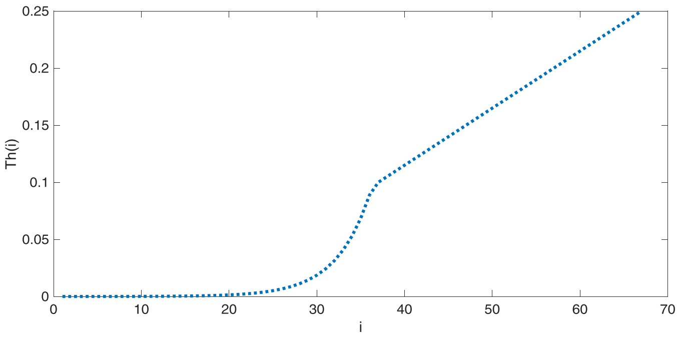


Fig 3. Variation of the decision threshold

Choice and optimization of parameters: The parameters of algorithm 1 that have to be optimized to improve the real time detection efficiency are: (i) sampling frequency Fs, (ii) block length N, (iii) power sequence length L, and (iv) decision threshold Th.

The Fs has to be chosen so that the spectral components of impulsive events will be covered. As a rule of thumb, the higher the Fs, the better the audio quality. However, increasing the value of Fs increases the number of samples of the audio waveform which leads to an increase of the computational complexity of the power. In our experiment, we have chosen a low sampling rate of 11025 Hz since the audio data we have downloaded are sampled at this exact rate. The value of N has to satisfy the following constraints: (i) it has to be quite low to reduce the computational complexity and be appropriate for real time execution, (ii) conversely, if the chosen value is too low, too many unnecessary details in the power sequence may arise; which can disturb the detection process. We have conducted several empirical tests to find the best choice by taking into account the real time execution and the decision exactitude. We have found that a value of 220 samples (20ms) is an appropriate choice. The length of the energy sequence L should be chosen so that the duration of the detected event is sufficiently high to generate a waveform signal that can be recognized in the second stage of the audio surveillance system. This value depends also on the length of the block. A value of L equals 30 provides an impulsive waveform of 0.6 s duration if N is set to 220 samples (20ms). This value is considered to be appropriate for the software and hardware configuration of the laptop we are using. The selection of the decision threshold Th is a very important step in the detection of impulsive events. Based on our experimental study, we have found that the value of Th can vary between 10^{-5} and 0.25. These boundaries are obtained by using the values of parameters described previously. The upper bound of this interval denotes the variance of the power sequence when no IS are occurring. The lower bound denotes the blocking threshold for which no impulsive event is detected whatever its intensity. The selection of an optimal Th depends on two main criteria. (i) The sound pressor level of audio data (ii) and the type of environmental disturbances (noise). The satisfaction of these criteria requires the use of audio data which are originating from several acoustical sources. It is very difficult to create scenarios that encompass all acoustic events.

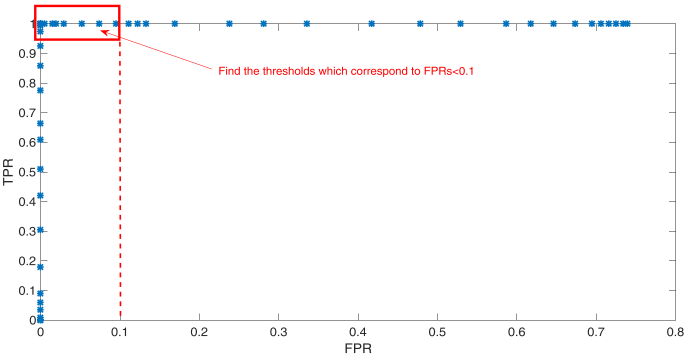


Fig. 4. Threshold selection (Example of seq(1) : 70dBSPL)

Therefore, we have used in our experiment the audio database described below to extract a sub-optimal threshold value. In order to address the first constraint, we have considered a set of three audio sequences which are composed of impulsive events as described before. The levels of pressure which are considered in our experiment are (i) 70, (ii) 80 and (iii) 90 dBSPLs. The Th value has to be estimated based on these audio sequences. The approach we have proposed in order to estimate this parameter is given as follows.

Algorithm 2:

Step 1 (Variables declaration):

seq$_{(i)}$, **ref**$_{(i)}$, **Th_sel**$_{(i)}$ and **Th_fin** are vectors,
Th, Fs, N, L, i and S are scalars,

Step 2: Loop

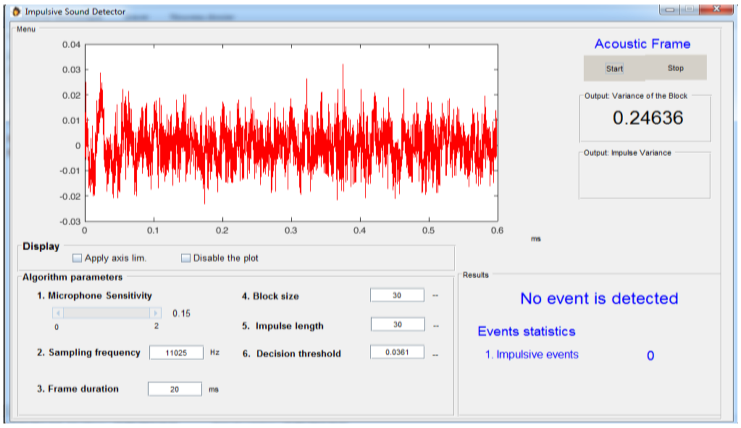
for i=1 to 3 do

1. Select the ith audio sequence: **seq**$_{(i)}$.
2. Select the IS starting instants: **ref**$_{(i)}$: The length of **ref**(i) is equal to 200 samples.
3. Generate threshold scales: Th $\in [10^{-5}, 0.25]$; Parameters: Fs=11025Hz, N=220 and L=30.
   We have found that Algorithm 1 is very sensitive to small variations of Th when its values fall within the interval $[10^{-5}, 0.1]$. Beyond 0.1, Th $\in [0.1, 0.25]$, the method becomes less sensitive. Therefore, logarithmic and linear scales have been used respectively for the first and the second intervals (Fig. 3).
4. Detect IS within **seq**$_{(i)}$ using *Algorithm 1*: The number of scales we have used is S=67.
   Estimate the detection errors: (i) Compute true positive rates (TPR) and false positive rates (FPR) using the set of thresholds generated in 4 [5], (ii) plot the ROC curve R(i).
5. Select a set of candidate thresholds (Fig. 4): Thresholds of which the FPRs are below 0.1 are selected **Th_sel**$_{(i)}$

end

Step 3: Decision threshold

The decision threshold Th that will be used in the real time detection phase is found by applying the following two steps: **(i):** extraction of the common values between these three vectors **Th_sel**$_{(1)}$, **Th_sel**$_{(2)}$ and **Th_sel**$_{(3)}$. The resulting set of values are saved in **Th_fin** vector.
(ii): computation of the median value of **Th_fin**. The decision threshold we have found in our experiment is Th=0.0361.



Output: Variance of the Block: 0.24636

Output: Impulse Variance

□ Apply axis lim.
□ Disable the plot

1. Microphone Sensitivity: 0.15 (0 – 2)
2. Sampling frequency: 11025 Hz
3. Frame duration: 20 ms
4. Block size: 30 --
5. Impulse length: 30 --
6. Decision threshold: 0.0361 --

No event is detected

Events statistics

1. Impulsive events 0

Fig. 5 Real-time detector of impulsive sounds

The parameters of *Algorithm 2* (thresholds and number of scales) have been empirically estimated by taking into consideration the properties of the audio acquisition device and the computer specifications given above.

2.2 Real-time processing

The real time processing block is a monitoring scheme that is permanently activated. It is composed of three main stages: (i) microphone, (ii) preprocessing module and (iii) IS detector.

Audio sensor: The acquisition of data is performed in real time using the same microphone as the one used the offline processing phase.

Preprocessing module: The main mission of the preprocessing module is to ensure the proper functioning of IS detection in different environmental conditions. Therefore, we are not obliged to readjusted the parameters of algorithm 1 that have been set previously when the environmental conditions change. Only one parameter in this module that has to be readjusted: the detection sensitivity. The proposed solution consists in adding a white noise y(n) following a normal distribution with mean μ and variance σ^2. The probability distribution of y is:

$$p(y) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(y-\mu)^2}{2\sigma^2}} \quad (8)$$

Therefore, equation (1) have to be substituted by the following equation:

$$e(k) = \frac{1}{N}\sum_{n=0}^{N-1}(x^2(n+kN) + y(n)) \quad (9)$$

The rest of *algorithm 1* remains unchanged. In this way, the detection sensitivity can be tuned by changing the variance of y(n). μ is set to zero since it does not affect the detection efficiency. Thus, the variance σ^2 corresponds to the detection sensitivity or microphone sensitivitiy.

IS Detector: The third stage uses algorithm 1 to detect IS**.** The parameters we have selected in the previous section are used in the real time phase to achieve better detection performance.

3 Software & results

We have implemented our prototype using MATLAB software on a laptop equipped with Core i5 processor and 6 G of RAM. The software is running on Windows 7 operating system. The audio waveform is acquired in real time using the microphone of the smartphone (connected wirelessly into the laptop). The audio surveillance system is permanently activated in an indoor environment. Once an impulsive event occurs, the software launches a visual signal and records the time of its occurrence. The system also records the number of detected impulsive events. In addition, it offers the possibility of readjusting the algorithmic parameters of the detector (Fig. 5). It is worth noting that the performance evaluation of real time sound detectors is not an easy task. In our experiment, we have planned a simple scenario which consists in real field trials using actual IS. The sounds of test were restricted to hands clap. The software was tested in indoor environment under noisy conditions (same room as described in sec. II). Noises are originating from two main sources: (i) people speaking in the hall of the building, and (ii) distant field construction sounds. The first step consists in setting up an empirical value of noise variance (microphone sensitivity threshold) which reduces the miss detection errors. As a rule of thumb, the higher the level of noise is, the larger the noise variance is requested. In our experiment, the sensitivity threshold was set at 0.15. The scenario is given as follows “a researcher is asked to clap his hands one time each 15s for a duration of 15 mn”. The performance of the detector is summarized in table1.

Table 1: Overall performance of the proposed IS detector

	TPR	**FPR**
Scenario results	85%	15%

4 Conclusion and future works

The main goal of this study is to design an efficient real time IS monitoring software. It is composed of three main stages: (i) acquisition sensor, (ii) preprocessing module and (iii) audio impulsive event detector. The software uses a microphone of a smartphone (MEMS-type) to acquire the audio data. The preprocessing module aims at tuning the microphone sensitivity to tackle any change in the environmental conditions (acoustical background noises). It consists in adding Gaussian white noise to the acquired waveform so that the undesired frequency components originating from the acoustical environment will be hidden. The tuning parameter is the noise variance.

The detection of impulsive events is based on the variability of the normalized power sequence through a variance analysis. The optimization of the detection parameters has been conducted offline using locally stocked IS. According to our experimental results, we have found that the software performs well in indoor environments. The advantages of our prototype are twofold: (i) a noncomplex solution, and (ii) easily adaptable to environmental conditions. The disadvantages of our solution can be summarized as follows: (i) the detection performance depends mainly on the hardware specifications of the laptop and the number of simultaneous running applications, (ii) consecutive impulsive events detection can only be achieved if the time offset between these events is less than 0.6s. We will focus our future works on five goals: (i) the optimization of the block length (ii) the exploration of Wireless Sensor Networks for distributed solution, (iii) the optimization of the sensor dynamic range, (iv) the global evaluation of the system by taking into account the reverberation effects, the size of the monitored area and its nature (indoor and outdoor), and (v) the proposition of real time impulsive sound recognizer.

References

1. M. Valera and S.A. Velastin, “Intelligent distributed surveillance systems: a review,” IEE Proc.-Vis. Image Signal Process., vol. 152, no. 2, pp. 192-204, April 2005.
2. P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, “Audio surveillance of roads: A system for detecting anomalous sounds,” IEEE Trans. Intell. Transp. Syst., vol. 17, no. 1, January 2016.
3. A. Dufaux, "Detection and recognition of impulsive sound signals," Phd Thesis, Institute of Microtechnology, Neuchatel University, Switzerland, 2001.
4. P. Foggiaa, N. Petkovb, A. Saggesea, N. Strisciuglioa and M. Vento, “Reliable detection of audio events in highly noisy environment,” Pattern Recognition Letters, vol. 65, pp. 22-28, 2015.
5. K. Lopatka, "Adaptive system for recognition of sounds indicating threats security of people and property employing parallel processing of audio data streams," Phd thesis, Gdansk University of Technology, Poland, 2015.
6. R. Levorato, “GMM classification of environmental sounds for surveillance applications," Master thesis, University of Padova, Italie, 2010.
7. A. Talal, U. Momin and A. Muhammad, “Improving efficiency and reliability of gunshot detection systems,” in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'13), Vancouver, BC, Canada, 2013.
8. J. Eargle, The Microphone Book, Elsevier, second edition, 2005.
9. bksv: http://www.bksv.com, last accessed: 2018/07/05.
10. St: http://www.st.com, last accessed: 2018/07/05.
11. Orange: http://www.wirelessorange.com/womic, last accessed: 2018/07/05.
12. Publicwifi: www.mypublicwifi.com, last accessed: 2018/07/05.
13. Sounddogs: https://www.sounddogs.com, last accessed: 2018/07/05.
14. N. Almaadeed, M. Asim, S. Al-Maadeed, A. Bouridane and A. Beghdadi, “Automatic detection and classification of audio events for road surveillance applications,” Sensors (Basel). vol 18, 1858, 2018.