Audio Search in a Large Audio Database

Larbi Guezouli1, Lahcene Guezouli[1], Hassane Essafi[2], M[ed] Amine Ouddan[3]
[1]LaSTiC UHL-Batna
1, rue Boukhlouf Med EL Hadi, Batna University
05000 Batna, Algeria
[2]CEA Fontenay-aux-Roses
92260 Fontenay-aux-Roses, France
[3]AdVestigo R&D
140 Bureaux de la Colline
92210 Saint-Cloud France
larbi.guezouli@univ-batna.dz, hessafi@cea.fr, aouddan@advestigo.com

ABSTRACT: *This paper proposes a new method of audio indexing and identification.*

This new method called CASIA (Calculation of Similarity Audio) is composed by two steps: indexing and identification.

In the indexing step, each audio document is pre-processed before the extraction of a set of basic audio descriptors, which characterize the temporal and the spectral information.

In the identification step, a query subsequence is pre-processed also and a set of audio descriptors is extracted. The comparison between query descriptors and descriptors of the database allows to generating the interference wave. This wave is used to calculate the similarity rate between the query and documents of database.

Keywords: CASIA, Information Retrieval (IR), Audio Processing, Sequence Fingerprint

Received: 20 November 2012, Revised 19 December 2012, Accepted 29 December 2012

1. Introduction

The search of data in a large database is very difficult, especially with audio data.

To search an audio sub-sequence in another audio sequence we must listen to the sequence from the beginning till the end. And to search this audio sub-sequence in a large database, we must listen to all audio sequences of the database. It must take a lot of time. There has been much interest in developing new and robust methods to do these searches quickly.

More precisely, our principal problem is to find all the audio segments present in the audio document query and which are similar to the protected audio documents. Many researches have been dedicated to the audio content-based retrieval [1-2, 4, 12, 15]. But the proposed methods are still time consuming and did not allow identifying with a high accuracy the reuse of segments of audio which are archived in a large database.

In our approach, the research of audio subsequence in a large audio database is done in two steps: The first step is dedicated to the characterisation of each audio signal of the database using a set of audio descriptors, while the second step is dedicated to the identification of audio document of the database that may contains portions of audio query. Like the characterisation step, during the identification step, the audio query is first characterised in order to extract descriptors which are compacted in a sequence structure called sequence fingerprint and that are used by CASIA method in order to measure the similarity between the query and the audio documents of the database.

Our sequence fingerprint process is decomposed into two levels: In the first level (low level) the audio signal is pre-processed and characterised using a set of audio temporal and spectral descriptors. The characterisation step has to describe an audio document as unique as possible in order to avoid false positive matches. To select the adequate descriptors, we have therefore evaluated their capacity to identify in real time of the documents contents of the audio dataset that have similar contents to that of the question [3].

The temporal descriptors are [13-14]: Energy, VSTD (Volume STandard Deviation), VDR (Volume Dynamic Range), VU (Volume Undulation), LER (Low Energy Ratio), ZCR (Zero Crossing Rate) and its derived descriptors like, HZCRR (High Zero CRossing Rate) [9], or statistics of ZCR. Spectral descriptors are based on Fourier Transform of the audio signal, for instance: FC (Frequency Centroid), BW (band Width), ERSB1, 2, 3 (Energy ratio in 3 subands), spectrum Flux [9], fundamental frequency or pitch, and other statistic descriptors based on fundamental frequency [8, 10-11].

In the second level (high level), the descriptors resulting from the low level are grouped into packet to capture the structure of the audio signal. The descriptors values are organized into a compact representation which is used to build sequence fingerprint characterizing an audio document with high level of discrimination.

The global objective of this work is the design of an environment for audio indexing and retrieval system.

2. Audio fingerprint extraction

2.1 Audio low level fingerprint extraction

As was mentioned in the introduction several audio descriptors have been developed in the literature. In this work, the descriptors have been analyzed, and six of them were selected to characterize our audio documents.

These descriptors have been computed considering audio frames of 2048 samples extracted from an audio signal sampled at 22050 Hz. The frames are overlapped with a window of 512 samples as shown in Figure 1.

Each frame is characterized using two kinds of descriptors: Frame descriptors, characterizes the entire elements of the frame. Sub-band descriptors, characterizes the sub-bands of the frame.

2.1.2 Frame descriptors

Volume: The most widely used and the easiest to compute is the volume frame feature. It allows detecting silent frames from not-silent ones.

$$V(n) = \frac{1}{n} \sum_{i=0}^{N=1} f_n^{\,2}(i) \qquad (1)$$

N: is the number of samples in the frame n.

i : is the i^{th} sample of the n^{th} frame of the signal.

ZCR (Zero Crossing Rate): To compute the ZCR of a frame we use the below formula that count the number of times that the audio waveform crosses the zero axes.

$$ZCR(n) = \frac{1}{2}\left(\sum_{i=1}^{N-1} |sign(f_n(i)) - sign(f_n(i-1))|\right)\frac{f_s}{N} \qquad (2)$$

$f_n(i)$: Amplitude of the i^{th} sample of the n^{th} frame.

f_s : Represents the sampling rate.

ZCR is the most indicative and robust measure to discern unvoiced or voiced speech.

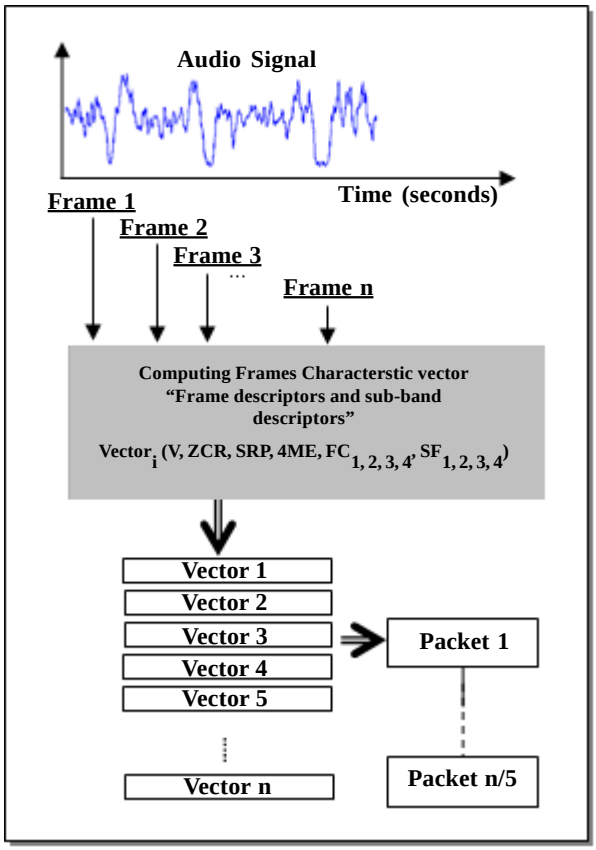


Figure 1. Sequence fingerprint extraction

SRP (Spectral Rolloff Point): We used the Spectral Rolloff Point as frequency domain feature [13] .It's defined as the 95th percentage of the power spectrum. This is useful to distinguish voiced from unvoiced speech. It is a measure of "*skewness*" of the spectral shape.

4ME (4 Modulation Energy): The volume contour of a speech waveform typically peaks at 4Hz. This feature was proposed by Shereir [13] to discriminate music from speech. It's defined as:

$$4ME = \frac{\sum_{i=0}^{N/T}\left(\sum_{j=0}^{T} W(j)\, C_n^2(j + i \times T))/T\right)}{\sum_{i=0}^{N} C_n^2(i)} \qquad (3)$$

$C_n^2(i)$: is the Fourier Transform of the volume contour.

$W(j)$: is a triangular window function centred at 4Hz.

N: Is the number of samples in a Frame.

T: Is the number of samples of the window W.

Speech frames usually have higher values of $4ME$ than music or noise frames.

2.1.2 Sub-band descriptors

In order to take into account the perceptual property of human ears, the entire frequency band of each frame is divided into four sub-bands, each consisting of the same number of critical bands correspond to cochlear filters in the human auditory model. When sampling rate is 22050 Hz, the frequency ranges for the four sub-bands are: 0-630 Hz, 630-1720 Hz, 1720-4400 Hz, and 4400-

11025 Hz. Then each sub-band k is characterized by two descriptors: *FC* (Frequency Centroid) and *SF* (Spectral Flux).

We denote by $S_n(k,i)$ the power of the i^{th} sample of spectrum of the sub-band k of the frame n.

1. *FC* is defined as the gravity center of the spectrum of the approximated area of the audio signal. It is calculated by *SFT* (Short Fourier Transform).

$$FC(n,k) = \frac{\sum_{i=0}^{N-1} iS_n(k,i)}{\sum_{i=0}^{N-1} S_n(k,i)} \quad (4)$$

FC is higher for musical area than a speech one. It's an important feature for characterizing timbre of music.

2. *SF* is defined as the average variation value of spectrum between two adjacent frames.

$$SF(n,k) = \frac{1}{N}\sum_{i=1}^{N}[\log(S_n(i,k)+\delta) - \log(S_{n-1}(i,k)+\delta)]^2 \quad (5)$$

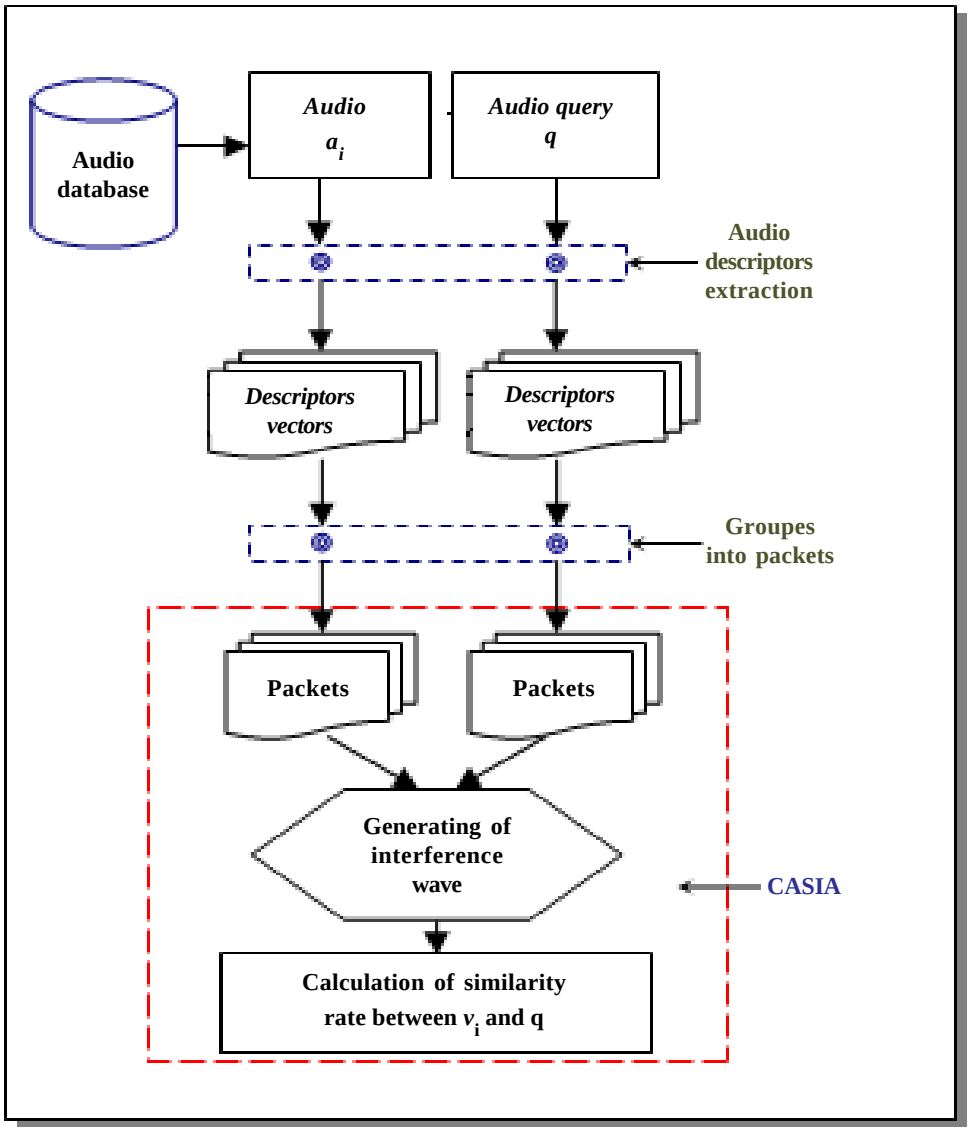


Figure 2. General schema of our approach

δ: A very small value to allow log operator to be usually defined.

The *SF* can be used to detect the musical area of the processed sound. Indeed the amplitude of *SF* of the musical sound is low compare to those of environment and speech.

The 12 extracted descriptors (4 frame descriptors and 2 for each of the four sub-bands) constituted the low level fingerprint that is used as input to produce the second level fingerprint.

2.2 Audio high level fingerprint extraction

In the second step the 12 characteristic vectors are grouped into packets (for the experiments, a packet encloses 05 characteristic vectors). The set of packets along the audio signal captures the audio information structure as shown in Figure 1.

The Table I summarizes the similarity between the three types of document (video, text and audio) on this level of fingerprint:

Document type	**Basic element of sequence fingerprint**
Text	Words
Video	Key images descriptors
Audio	Packets descriptors

Table 1. High level fingerprinting similarity between Text, Video and Audio documents

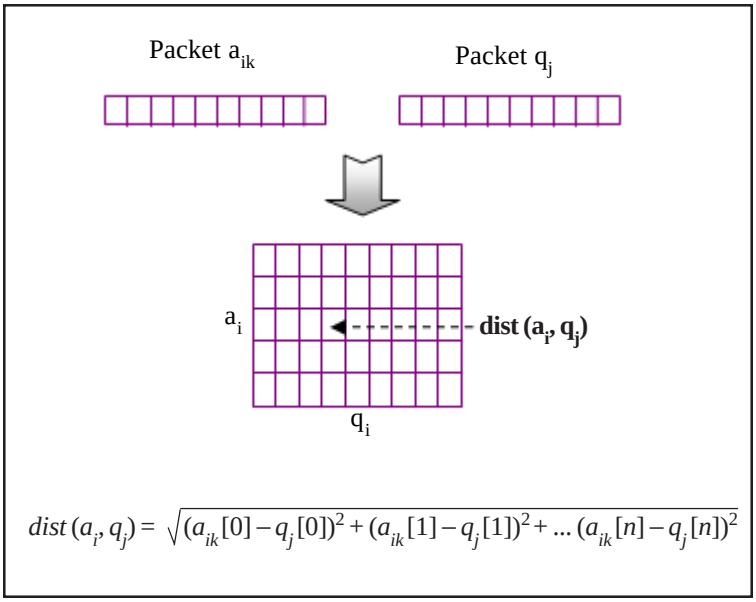


Figure 3. Calculation of distances matrix

CASIA method (CAlculation of the SImilarity of Audio documents) is inspired of our CASIT method (CAlculation of the Textual SImilarity).

CASIT method gave encouraging results. We used the same principle with the audio data. Figure 2 shows the general schema of this method.

The principal is to generate an interference wave [7] by using packets of audio sequences.

An intermediate step is to calculate a distances matrix. A case of this matrix consists to the Euclidian distance between a packet of the reference audio sequence and a packet of query sequence.

Figure 3 illustrates the calculation of distances matrix between packets of audio document and those of query document.

Figure 4 shows distances matrix, in binary form, calculated using two similar songs with noises in query song.

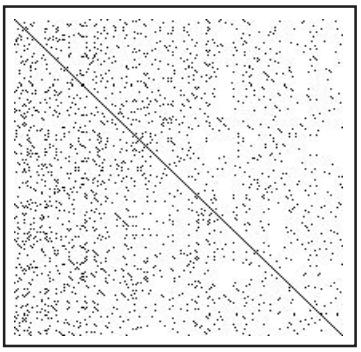


Figure 4. Distances matrix in practice

A point of binary form equals 0 if the corresponded distance is less than a threshold and 1 otherwise. The diagonal line shows the common parts between the two audio documents corresponding to each sequence matched packets. The other points represent the isolated pairs of packets whose descriptors are similar.

2.2.1 Generating interference wave

Using this distances matrix, we generate the interference wave as shown in Figure 5.

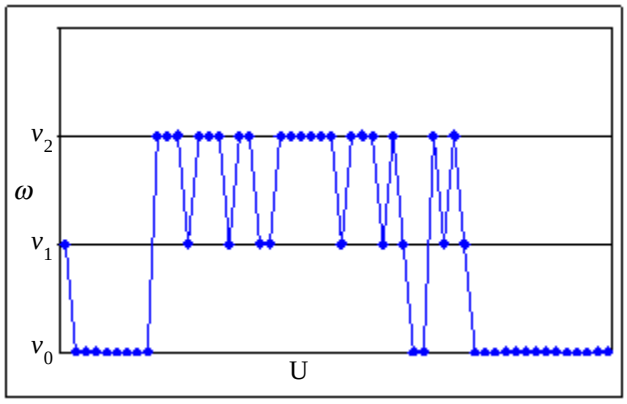


Figure 5. Interference wave

The interference wave is computed as follow:

$$Interference\ wave\ (u) = w \ \ where\ w \in \{v_0, v_1, v_2\}$$

The meaning of the three values of the elements of the estimator vector is:

- v_0 if $PQuest_j$ is different from $PBase_i$.

- v_1 if $PQuest_j$ matched with $PBase_i$ and $PQuest_{j\text{-}Radius}$ is different from $PBase_{i\text{-}Radius}$.

- v_2 if $PQuest_j$ matched with $PBase_i$ and $PQuest_{j\text{-}Radius}$ matched with $PBase_{i\text{-}Radius}$.

With:

- $PQuest_j$ is the j^{th} packet of question fingerprint.

- $PBase_i$ is the i^{th} packet of the base fingerprint.

- $PQuest_{j\text{-}Radius}$ is the behaviours packets of the j[th] packet with a fixed neighbouring radius.

- $PQuest_j$ is different from $PBase_i \Rightarrow (dist\,(PBase_i, PQuest_j) > threshold)$

- $Quest_j$ matched with $PBase_i \Rightarrow (dist\,(PBase_i, PQuest_j) > threshold)$

Therefore, instead of comparing two audio documents we compare two interference waves [5].

Next figures represent different cases of comparison between two audio sequences.

Figure 6 presents the interference wave of a comparison between two audio sequences. The question audio sequence contains noises. The final score of this case is 76%.

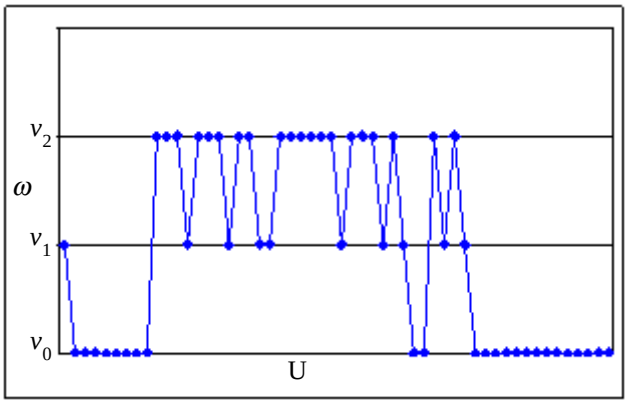


Figure 6. Comparison between two similar audio sequences with noises

Figure 7 presents the interference wave of a comparison between two similar audio sequences. The final score of this case is 99%.

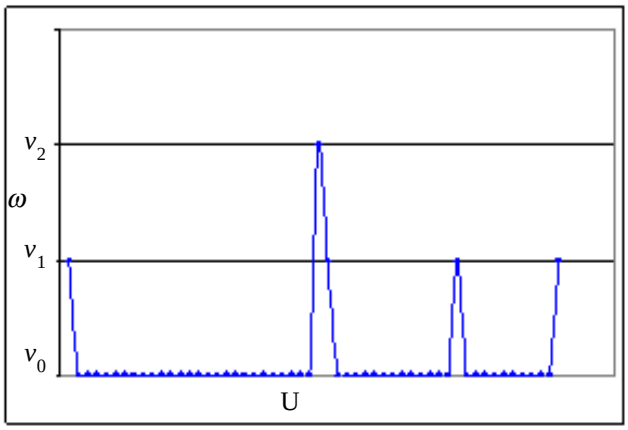


Figure 7. Comparison between two similar audio sequences

Figure 8 presents the interference wave of a comparison between two different audio sequences. The final score of this case is 1%.

Then, instead of comparing audio documents we compare interference waves [5-6].

This comparison is made by calculating the similarity rate between packets. This similarity is calculated by using the interference wave.

An intermediate step is to convert the interference wave to two interference vectors.

2.2.2 Interference vectors

From the interference wave, we calculate the two interference vectors $\boldsymbol{V_0}$ and $\boldsymbol{V_1}$. The vector $\boldsymbol{V_0}$ is built by using the sequences of $\boldsymbol{u} \in \boldsymbol{U}$ such as *InterferenceWave* $(u) = v_0$ and $\boldsymbol{V_1}$ is obtained by using the sequences of $\boldsymbol{u} \in \boldsymbol{U}$ such as *InterferenceWave*$(u) = v_1$.

The $\boldsymbol{n^{th}}$ element of $\boldsymbol{V_i}$ vector contains the number of n-grams of level $\boldsymbol{i}$, i.e. the number of sequences of $\boldsymbol{n}$ elements of level $\boldsymbol{i}$ in the interference wave.

number of 1-grams	number of 2-grams	number of 3-grams	…	number of n-grams

$\boldsymbol{V_i}$: Interference vector of the level $\boldsymbol{i}$

In the example of the Figure 5., the interference vectors are:

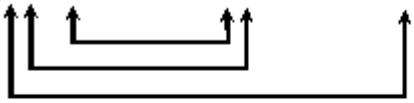
0	1	0	0	0	0	0	1	0	0	0	0	0	1

$\boldsymbol{V_0}$: Interference vector of the level **0**

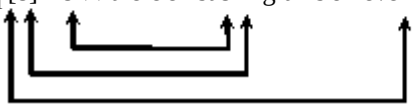
8	1

$\boldsymbol{V_1}$: Interference vector of the level **1**

$V_0[8] = 1 \Leftrightarrow$ there exist 1 8 -grams of level 0 in the interference wave



$V_1[8] = 8 \Leftrightarrow$ there exist 8 1 -grams of level 0 in the interference wave



2.3 Similarity calculation

The function ***sim*** defines the similarity rate between the query sequence $\boldsymbol{Q}$ and the database sequence $\boldsymbol{D}$. It is defined by using interference vectors $\boldsymbol{V_0}$ and $\boldsymbol{V_1}$ as follows:

$$sim = \frac{2 \times \sum_{j=1}^{n} \frac{1}{\lambda_j} \times V_0[j] + \sum_{j=1}^{m} \frac{1}{\lambda_j} \times V_1[j]}{2} \times 100$$

In this equation, $\boldsymbol{n}$ is the size of $\boldsymbol{V_0}$, $\boldsymbol{m}$ is the size of $\boldsymbol{V_1}$ and $\boldsymbol{l_j = T / j}$. ($\boldsymbol{l_j}$ the maximum number of j-grams in the sequence $\boldsymbol{D}$ and $\boldsymbol{T}$ is the number of packets of the sequence $\boldsymbol{D}$).

4. Results

Our retrieval process based on CASIA method, has been evaluated on archived database constituted of 5000 original audio documents (mp3 format) representing different musical genres that correspond to 21.0 Go. We downloaded 564 documents from the peer to peer network to constitute a query-database. Some titles of this base are present also in archived database.

The time of producing fingerprint of archived database documents is around 18 hours using PC with Intel Xeon processor CPU 2.8 GHz and 2.0 Go of RAM. The volume of the produced fingerprints (packets) is 236.4 Mo. The time processing of answering query from query-database is around 4 seconds.

To measure the pertinence of retrieval process, we have classified the obtained responses in five categories:

R1: The original of the query document is present in the database, and was identified by the system.

R2: The original of the query document is present in the database, and wasn't identified by the system.

R3: The original of the query document is present in the database, and was incorrectly identified by the system (false similar document).

R4: The original of the query document is not present in the database, and was identified by the system with a not similar document.

R5: The original of the query document is not present in the database, and wasn't identified by the system.

Table 2 summarizes the percentage of these categories from the total requests (the categories proportion of the total answer):

	R1	R2	R3	R4	**R5**
Percentage from the 564 requests	**54.66**	1.79	0.88	0	**42.62**

Table 2. Results of the 564 requests

The percentage of the positive requests (represented by R1 and R5) is: 97.28% and the percentage of the negative requests is: 2.67%.

5. Conclusion

We presented in this article how we index and identify an audio sequence using a new concept called interference wave.
The indexation of audio documents is based on the content of these documents, and the search takes into account the neighbourhood of packets, which increases the performances of our system.

We described the new concept of the interference wave. This latter is a three levels wave, it helps us in the calculation of similarity rate between query and database documents.

Our system is tested by using some queries and several corpuses. The percentage of the positive responses (represented by R1 and R5) is: 97.28% and the percentage of the negative responses: 2.67%.

These tests show that the performances of this new method are good.

References

[1] Chechik, G., et al. (2008). Large-scale content-based audio retrieval from text queries, *In*: Proceedings of the 1st ACM international conference on Multimedia information retrieval. 2008, ACM: Vancouver, British Columbia, Canada. p. 105-112.

[2] Ding, D., et al. (2012). Beyond audio and video retrieval: towards multimedia summarization, *In*: Proceedings of the 2nd ACM International Conference on Multimedia Retrieval. ACM: Hong Kong, China. p. 1-8.

[3] Essafi, H., Sayah, S., Ouddan, M. A. (2006). Robustness Evaluation of the Basic Descriptors for Audio Indexing. *In*: the 12th International Multimedia Modeling Conference.

[4] Foote, J. (1999). An overview of audio information retrieval. *Multimedia Syst.* 7 (1) 2-10.

[5] Guezouli, L. (2007). Gestion de documents plurimedia et recherche d'informations dans un système collaboratif, *In*: R&D AdVestigo. University of Denis Diderot, Paris VII.

[6] Guezouli, L., Essafi, H. (2010). CASIT: Content Based Identification of Textual Information in a Large Database. *In*:Advanced Information Networking and Applications Workshops (WAINA), IEEE 24th International Conference on.

[7] Guezouli, L., Essafi, H., Goossens, B. Identification par le contenu des informations textuelles dans une large base de documents multimédias. *In*: Traitement et Analyse de l'Information. Méthodes et Applications (TAIMA'03). Hammamet, Tunisie: La magrebine à l'impression.

[8] Lebossé, J., Brun, L., Pailles, J. C. (2007). A robust audio fingerprint extraction algorithm, *In*: Proceedings of the Fourth IASTED International Conference on Signal Processing, Pattern Recognition, and Applications. ACTA Press: Innsbruck, Austria. p. 269-274.

[9] Lu, L., Jiang, H., Zhang, H. (2011). A robust audio classification and segmentation method. *In*: Proceedings of the ninth ACM international conference on Multimedia. Ottawa, Canada: ACM.

[10] Ramona, M., et al. (2011). Audio Fingerprint: a Public Evaluation Framework Based on a Broadcast Scenario. *Journal of Experimental & Theoretical Articificial Intelligence*, (Special Issue on Event Recognition).

[11] Rossignol, S. (2000). Segmentation et Indexation des Signaux Sonores Musicaux. University of Paris 6.

[12] Salton, G., McGill, M. J. (1983). Introduction to modern information retrieval. McGraw-Hill.

[13] Scheirer, E., Slaney, M. (1997). Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. *In*: Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)- V. 2. *IEEE Computer Society*.

[14] Wang, Y., Liu, Z., Huang, J.-C. (2000). Multimedia Content Analysis Using Both Audio and Visual Cues. *IEEE Signal Processing Magazine*.

[15] Zhang, T., Kuo, C. C. J. (2001). Content-Based Audio Classification and Retrieval for Audiovisual Data Parsing. Springer.