ROBUST FREQUENCY-BASED AUDIO FINGERPRINTING

Elsa Dupraz and Gaël Richard

Institut TELECOM, TELECOM ParisTech, CNRS-LTCI
37, rue Dareau - 75014 Paris, France

ABSTRACT

Pure frequency-based audio fingerprint systems have the capacity of handling very short fingerprints while being highly robust to perturbations such as additive noise or compression. However, these approaches are often complex and fail to identify time stretched signals. We propose in this paper two extensions of an existing system and test the robustness of the overall system in different conditions. It is shown that the search strategy adopted allows for a clear reduction of complexity with very limited degradation of performances and that the new system is robust to additive noise and speed changes up to 5%.

Index Terms— Audio Identification, Audio Fingerprint, noise and speed-change Robustness.

1. INTRODUCTION

Audio Fingerprinting consists in automatically identifying an audio excerpt (e.g. obtaining metadata such as artist name and song title using the audio signal only). This problem has received a great interest since it can be the cornerstone of multiple applications including audio track identification via cell-phone, radio broadcast monitoring, jingle detection for audio segmentation, copyright control .... Audio fingerprinting involves the extraction of a signature (or "fingerprint") for each audio document or part of it which are then stored in a reference database. An unlabeled audio excerpt is then identified by comparing its fingerprint with those of the reference database.

Several approaches already exist for audio fingerprinting (see [1] for a review). In most cases, the fingerprint is based on spectro-temporal information. For example, a number of methods are based on the analysis of the energy in subbands (such as, for example, the use of the energy flux modulation [2], spectral magnitudes [3] or the sign of the energy differences [4]). Other authors have also proposed more specific representations based on MPEG7 descriptors [5] or sinusoidal onset locations [6] as in the *Shazam* solution. Most of these approaches are capable of very high accuracies but suffer at varying degree to degradations such as additive noise, compression or speed changes. A traditional mean to overcome these problems is to use longer fingerprint duration but obviously this has two major drawbacks: a longer response feedback from the system and the impossibility to identify short events. This was one of the main motivations for the introduction of pure frequency-based audio fingerprinting [7, 8] where high accuracy can be obtained for very short duration (e.g. 97 % recall rate in a jingle detection task with $1s$ long fingerprints). This approach is therefore particularly attractive but has two shortcomings: first, it is rather complex and may become untractable for very large fingerprint databases; second, since it is based on the sole-frequency information a small shift of the frequency content dramatically impacts the performances (e.g. this approach is not robust to speed changes).

The aim of this paper is then to introduce specific contributions to first significantly reduce the complexity by following specific search strategies and second to obtain decent identification scores in presence of speed changes. The proposed algorithms are tested on two databases and compared to our own implementation of the original algorithm [7].

The paper is organized as follows: the overall architecture of the approach is summarized in next section. The new modules proposed, namely the speed change robustness unit and candidate pre-selection and ranking are then respectively detailed in sections 3 and 4. The experimental results are then exposed in section 5 before suggesting some conclusions.

2. SYSTEM OVERVIEW

2.1. Fingerprint extraction

The audio signal is first transformed to the Fourier domain using $64ms$ analysis windows with a 50% overlap. Then, predominant peaks (or sinusoidal components frequencies) are extracted for each frame and grouped together for L successive frames corresponding in our case to a duration of $1s$ for the unlabeled signal. To reduce the fingerprint size, it is then possible to only retain the n most salient peaks within these L successive frames ($L = 100$ in our case for the $1s$ unlabeled signal). The fingerprint is then an ensemble of time localized (e.g. frame numbers) frequency peaks.

Note that a fingerprint for the reference database can be extracted from part or from the entire audio file whereas a fingerprint for identification is extracted from only $1s$ of the

audio signal. This explains why a different number of peaks can be chosen for the reference and the unlabeled signals.

2.2. Fingerprint matching

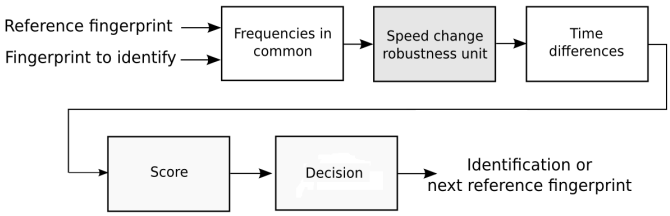

Fig. 1. Comparison between two fingerprints

To perform the audio identification, the fingerprint extracted from the unlabeled audio signal is compared to all fingerprints of the reference database. The main steps are summarized in Fig. 1 including the speed-changes robustness unit proposed in this paper. First, peak frequencies in common between fingerprints (with a given frequency tolerance) are looked for. Then a table containing time differences between peaks at the same frequency is established. The matching score is then obtained as the maximum number of time a given time difference is observed. The audio identification is granted when this score is above a pre-defined threshold (see [7, 8] for more details).

3. SPEED-CHANGES ROBUSTNESS

Speed changes are often used by broadcasters to comply with strict program schedule constraints. The effect on the audio signal of such processing is a constant shift (by a factor K) of all frequencies. The main idea of the speed change robustness unit proposed is to determine this constant shift K prior to the time differences computation (see Fig. 1). More precisely, it is first assumed that two peak frequencies (f_c and f_r respectively for the test and reference signals) can be associated if $f_r \in [f_c - T_f, f_c + T_f]$ where T_f is the frequency tolerance chosen. For each associated frequencies, the estimated frequency shift is computed as $\hat{K}_{c,r} = f_c/f_r$. An overall shift $\hat{K}$ between the two fingerprints is then assumed if the same value of $\hat{K}_{c,r}$ appears more often than a predefined threshold value. If this is the case, all peak frequencies of the unlabeled signal are multiplied by $\hat{K}$ prior to the fingerprint matching.

Note that this additional unit has a much lower complexity than the reference database search and has therefore little influence on the overall complexity.

4. COMPLEXITY REDUCTION

In the initial algorithm, complexity is in $O(N \times n)$ where n is the number of peaks in the fingerprint to identify. In fact, many reference fingerprints are compared whereas they do not have any chance to be chosen. We propose below two specific strategies in the framework of this frequency-based algorithm based on candidate pre-selection or ranking.

4.1. Candidate pre-selection

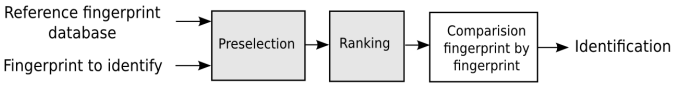

Fig. 2. Preselection and Ranking modules

The main idea is to eliminate reference fingerprint candidates that do not have a sufficiently high number of common frequencies with the test fingerprint. In a first step, the number of common peak frequencies are reduced by only counting those frequencies that appear more often in the reference than in the test candidate. Then, in a second step, the list of potential reference candidates is reduced by considering that the reference fingerprint must have at least a given percentage of common frequencies with the test fingerprint to be considered. For example, a preselection ratio of 30 % indicates that the selected candidates have at least 30% of their peak frequencies in common with the test fingerprint. Since a look-up table is used to find the frequencies in common, this approach leads to a significant decrease of complexity (which becomes of the order of $O(N_s \times n)$ where N_s is the number of reference candidates finally selected).

4.2. Candidate ranking

As it will be shown in the results, the previous hard selection threshold has a significant impact on performances. In fact, a candidate with a small number of common frequencies may still obtain a high score particularly in adverse conditions. This motivated the following alternative based on candidate ranking. In this approach, the reference fingerprints are ranked according to the number of frequencies they have in common with the fingerprint to identify. Then, the search is performed in the order of the ranking until the matching threshold is met. As a consequence, the more common frequencies a reference candidate has with the unlabeled song, the sooner it will be compared. This approach has two advantages. Firstly, since it is highly probable that the correct fingerprint is well placed in the ranking, the mean response time of the system significantly decreases. Secondly, if the good candidate obtains a high matching score with only a small number of common frequencies, it still has a chance to be retrieved while it would have been discarded with a hard pre-selection strategy.

5. EXPERIMENTAL RESULTS

5.1. Databases

Two databases, ID_303 and ID_1772, were used to evaluate the performances of the algorithm. Both databases contains 10 seconds long music excerpts from a variety of music genres (rock, pop, jazz, rap, metal and classical). All excerpts are sampled at 16 kHz. ID_303 is a small database of 303 files and is exploited in this paper to assess the robustness of the identification to noise and speed-changes. It is also used to highlight the impact of the candidate pre-selection. The other database, ID_1772, is significantly larger and contains 1772 files. This database is used to evaluate the performance of the complete system, should it be with or without candidate ranking. Although these databases are significantly smaller than those used in [9], their size are comparable to those of other studies and are sufficient to demonstrate the merit of the additional modules proposed since the original algorithm has already been thoroughly evaluated in [8] for a number of degradations.

5.2. Experiments and evaluation protocols

All experiments are conducted using the following protocol:

1. *Building the reference fingerprint database:* For each $10s$ musical extract, a single fingerprint is generated. It is computed using $64ms$ overlapping analysis windows. The overall fingerprint size is fixed to 500 peaks (namely 50 peaks per second).

2. *Identifying an audio segment:* For each extract, a segment of $1s$ is randomly selected. A fingerprint is computed on this segment using 100 peaks (e.g. twice as much peaks per seconds than for the reference). This fingerprint is then compared to the reference fingerprint database. The identification is positive (resp. negative) if the original $10s$ musical extract is retrieved. The recognition rate then corresponds to the ratio of correct identification. Note, that it is a slightly conservative approach since if the matching is found within the correct $10s$ extract it is counted as positive (e.g. the capacity of the algorithm to precisely localise the corresponding segment is not evaluated).

In this paper, we limit our evaluation to the impact of the proposed additional modules (pre-selection, ranking and speed-changes robustness). The first two experiments are dedicated to the evaluation in terms of recognition rate and complexity of the candidate pre-selection and ranking modules. In addition, the *mean response time* which corresponds to the CPU time needed to retrieve the original file from its $1s$ fingerprint, is computed. All running time values were obtained on an Intel Core 2 Duo, $2,4Ghz$, $1,98Mo$ RAM using a non-optimized Matlab implementation. The third experiment is dedicated to the robustness analysis of our algorithm to additive white Gaussian noise. To that purpose, white Gaussian noise is added to all test signals (with a Signal-to-Noise ratio of $+20dB$ or $+10dB$). The $1s$ fingerprints extracted on these noisy signals are then compared to the original fingerprint references, e.g. computed on the original $10s$ clear reference signals. The performances, with and without the candidate pre-selection module, are then compared to the matched condition (no noise). In the last experiment, the efficiency of the speech-changes robustness unit is evaluated. To that aim, the tested signals are accelerated or slowed up by a factor of 5% (which is comparable to the value used in [10]). And, as for the previous experiment, the $1s$ fingerprints extracted on these speed changed signals are then compared to the original fingerprint references and the performance are computed with or without the use of the proposed speed-changes robustness unit.

Note that, the larger database, ID_1772, is only used for the evaluation of the candidate ranking module due to complexity burdens of our implementation when candidate selection or ranking is not used.

Preselection	0%	30%	40%	50%	60%
Recognition rate	99%	97%	95%	93%	82%
Mean response time	$7.9s$	$6.7s$	$3.1s$	$2.5s$	$1.2s$

Table 1. Preselection evaluation on ID_303

	Without ranking	With ranking
Recognition rate	90%	98%
Mean response time	$9.1s$	$2s$

Table 2. Candidate ranking figures on ID_1772

5.3. Results

Effect of preselection: As highlighted in Table 1, preselection alone is not efficient and leads to degraded accuracies for low pre-selection ratio. This is due to the fact that a correct fingerprint pair may possess a rather low number of common frequencies, especially in adverse conditions. As a consequence, a hard pre-selection threshold based on the number of common frequencies may eliminate the good candidate.

Candidate ranking: The performance of our algorithm with the candidate ranking module are compared in Table 2 to the performance of the original algorithm with 50% preselection. The candidate ranking proves to be very efficient and clearly outperforms the pre-selection strategy. The complexity of this proposal is also very advantageous and allows for a mean identification response time of $2.0s$ on ID_1772 (compared to $9.1s$ without the ranking module but with 50% preselection). Indeed, our proposal, with the pre-selection or ranking modules, has a complexity in $O(N_t \times n)$ where N_t is the number of candidates tested. The reduction in complexity is then due to the fact that N_t is much smaller than N.

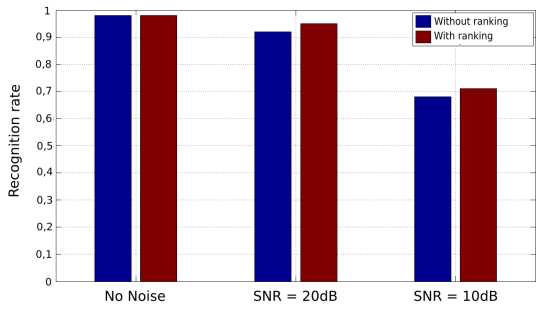

Fig. 3. Robustness to noise on `ID_303`

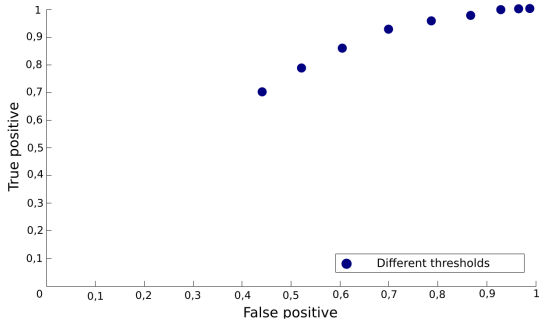

Fig. 4. Evaluation of the orignal algorithm [7] with different thresholds for speed change of 5% on `ID_303`

Robustness to noise: Figure 3 displays the results obtained by our algorithm in presence of additive white Gaussian noise for a pre-selection ratio of 10 %. It can be first observed that the performances remain acceptable even for a rather high level of noise (about 70 % correct identification for $1s$ fingerprint and a SNR of $10dB$). Then, it can be seen that the robustness to noise is not impacted by our ranking module since it even performs slightly better than the original algorithm with 10 % preselection.

Robustness to speed changes: The original algorithm (e.g. without speed change unit) performs poorly in the case of speed changes between the reference and test fingerprints. Indeed, Figure 4 which provides the identification results (False positive vs True positive) in the case of 5% speed changes, shows that no satisfying operating point (or threshold) can be found. Indeed, high true positive leads at the same time to high false positive. On the opposite, our proposal with the speed change robustness unit proposed allows for very satisfying performances with 1% speed changes and only limited degradations for 5% speed changes for an operating point leading to 98% in the case of no temporal modification (see Table 3).

Speed-change	True positive	False positive
0%	98%	2%
1%	96%	4%
5%	89%	11%

Table 3. Robustness to speed changes on `ID_303`

6. CONCLUSION

Frequency-based audio fingerprinting systems are efficient for short duration fingerprints but traditionally suffers from two major drawbacks: a rather high complexity and low robustness to speed changes. The purpose of this paper was then to propose two efficient means to reduce the complexity and to allow for higher performances to speed changes since those are frequently used by broadcasters. Future work will be dedicated to the scale up of this approach to very large databases and to the improvement of the algorithm in adverse conditions (such as low SNR for background music identification or highly reverberant environments).

7. REFERENCES

[1] P. Cano, E. Battle, T. Kalker, and J. Haitsma, "A review of audio fingerprinting," *Journal of VLSI Signal Proc.*, vol. 41, 2005.

[2] J. Laroche, "Process for identifying audio content," *US Patent N WO88900*, 2001.

[3] J. Pinquier and R. Andre-Obrecht, "Jingle detection and identification in audio documents," in *ICASSP*, 2004.

[4] J. Haitsma and T. Kalker, "A highly robust audio fingerprinting system," in *ISMIR*, 2002.

[5] E. Allamanche, J. Herre, O. Hellmuth, O. Froba, and M. Cremer, "Audioid: towards contet-based identification of audio material," in *110th conv. of the AES*, 2001.

[6] A. Wang, "The shazam music recognition service," *Comm. of the ACM*, vol. 49(8), march 2006.

[7] M. Betser, P. Collen, and J.-B. Rault, "Audio identification using sinusoidal modelling, and application to jingle detection," in *ISMIR*, 2007.

[8] M. Betser, *Modelisation sinusoidales et applications a l'indexation sonore*, Ph.D. thesis, Telecom ParisTech, 2008.

[9] C. Bellettini and C. Mazzini, "Reliable automatic recognition for pitch-shifted audio," in *ICCCN*, 2008.

[10] J. Haitsma and T. Kalker, "Speed-change resistant audio fingerprinting using auto-correlation," in *ICASSP*, 2003.