

Reliable Automatic Recognition for Pitch-Shifted Audio

C. Bellettini[†] and G. Mazzini
ENDIF – University of Ferrara
Via G. Saragat, 1 – 44100 Ferrara – ITALY
Email: {carlo.bellettini, gianluca.mazzini}@unife.it
Telephone: (+39) 0532–974848
Fax: (+39) 0532–974840

Abstract—In this paper we present the difficulties involved in recognizing pitch-shifted audio, discussing its usage and implications. In the framework of audio fingerprinting techniques, we address the issue by proposing two complementary solutions, both of which can be exploited without affecting an existing reference database. As fingerprint algorithm, a well-known, robust method is employed in a modified version and providing insights on its peculiarities. Tests were carried out on a vast song library and exhibit an excellent success rate, up to considerable distortion magnitudes.

Index Terms—Audio fingerprinting, pitching, features extraction, music information retrieval.

I. INTRODUCTION

The availability of more computing power at steadily decreasing costs certainly plays a key role in the current, widespread interest in many applications related to signal processing. Indeed, chances are that what a decade ago would have been practical only by means of specialized hardware could now be accomplished in software (sometimes even on portable devices). From this point of view, automatic audio recognition is no exception.

Its complexity, however, arises a high number of issues, even leaving the vast field of speech recognition out. They range from unearthing peculiar features buried in a piece of audio, to store and retrieve them in an efficient way. Audio recognition may aim to discern whether or not two pieces are in fact the same [1], regardless of their “outer appearance” (i.e. coding, distortions). More generally, it comes into play when we are interested in reliably identifying an unknown excerpt of e.g. music, given a large set of references (a more thorough overview will follow in II). This would allow to easily metatag digital music, or to fill in a list of songs broadcast by a radio or played in a mall. There is also a strong demand for a way to spot illegal exploiting of copyrighted music.

According to the researcher’s background and goals, the topic can be addressed in many different ways. For solo instrumental music, a good solution can be that of identifying the single notes composing a piece [2], but for now it is unfortunately not effective in more general scenarios. A large choice of different audio features can be extracted [3], usually chosen on the basis of heuristic considerations, with a few exceptions [4]. It is also possible to combine more of them in one single identification model [5], in the hope of mutually compensate eventual weaknesses. Others instead put their efforts into the matching problem [6]: by borrowing gene-sequence matching algorithms from biology, an event-based audio sequencing method was proposed, regardless of the particular audio feature extracted.

On our side, we focused our attention on a simple yet very robust algorithm, whose basic operation was first described by Haitsma *et al.* in [7]. The distinctive feature it extracts is intimately related to the audio signal energy. More specifically, it takes into account how the energy difference among a set of sub-bands varies in time.

[†]Corresponding author. We are both grateful to Knowmark s.r.l. [13], for their financial and technical support and their valuable suggestions.

High robustness apart, this algorithm proves attractive also for the convenient output it produces (a bit matrix) and for its flexibility.

Our previous investigation [8] showed that improvements could indeed be attained by means of a careful tuning of the parameters. Through this algorithm and a few enhancements we successfully addressed the problem of recognizing pitch-shifted audio, which is ordinarily a demanding task. In the first instance, it displaces all frequency components, since the “pitch” can be defined as the perceived frequency of a sound. More will be explained later, in III.

The paper is organized as follows: section II introduces the system and the algorithms, while section III investigates the pitch-shift issue. Simulation results are reported in section IV and finally section V concludes the paper.

II. AUTOMATIC AUDIO RECOGNIZER

There exists a great variety of rationales and aims for an audio recognizing system [9] [10], ours is shown in Fig. 1. Blocks FA and SA stand for “fingerprint” and “search” algorithm respectively, while block DB is the reference database. First of all, FA allows to compute the fingerprint of a large number of songs, which will act as references. This lengthy, off-line stage must be performed only once, at least until FA does not change. Our library comprises almost 15.000 pieces in MP3 format, whose fingerprints make up the database.

The everyday use of the system is pictured in the lower branch. Some source provides the piece of audio we wish to identify and whose duration can range from a few seconds to many hours, depending on the particular application. In any case, 3 or 4 seconds are usually enough to perform a reliable identification. If a company is interested in a short commercial, another can rather prefer tracking down all the songs transmitted in a long radio or TV broadcast or recording.

Both tasks can be accomplished by the same system. With respect to the former, the input is used “as is”, while for the latter we extract many short excerpts. The time interval between them must be carefully chosen, according to the particular context. For instance, if

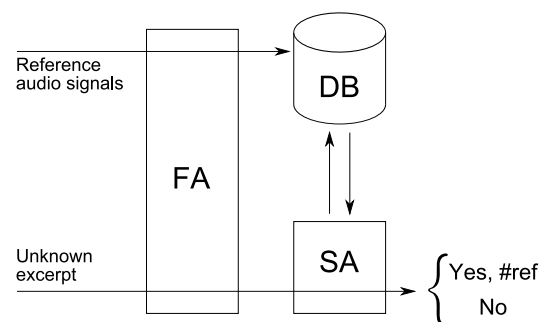


Fig. 1. Scheme of the audio recognizing system.

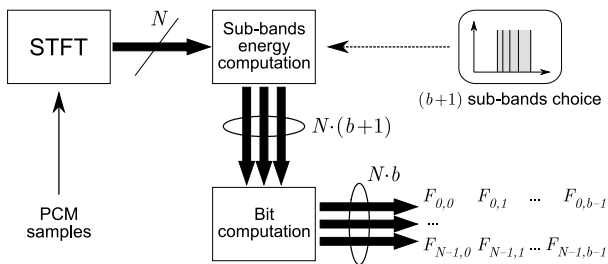


Fig. 2. The fingerprint algorithm (FA).

we are dealing with a sequence of songs, a one-minute spacing should be enough in order to hit every one of them at least once. Since this “stream approach” heavily relies on the basic single-excerpt case, it will be not further discussed here.

A. Fingerprint extraction scheme

Details and insights regarding our current implementation for FA can be found in [8] and will be summarized thanks to Fig. 2, where major blocks are depicted. The input is assumed to be a piece of audio in the form of PCM samples: this may imply one or more preliminary conversion stages, depending on the particular source available. When dealing with our MP3 library, for example, we introduced an appropriate decoder, whereas a microphone was used to digitize ambient music. The PCM samples could in principle have any sample rate or quantization level, but we kept the usual setting of 44.100kHz and 16bit, motivated by our concern on music audio. As a last pre-processing step, stereo inputs are converted to mono.

On the basis of what stated in the introduction, the employment of a short-time Fourier transform (STFT, quite common in the field of audio signal processing [2] [5]) is not surprising. The Hann window applied has a duration of about 372ms (which gives a very good frequency resolution), with an overlap between frames slightly above 96%. Such a large value not only alleviates to a great degree the negative impact of the necessary non-synchronization between a reference piece of audio and an unknown excerpt, but it can also prove valuable in facing the pitch-shift issue, as explained in III-A. On the other hand, should the overlap be smaller, the size of the database would be correspondingly reduced.

For each temporal segment n of the N available, a band meaningful for the human ear (up to 2kHz) is divided up into $b + 1$ sub-bands, each of them responsible for a computed energy contribution $E_{n,m}$ where $0 \leq m \leq b$ is the sub-band. The choice of sub-bands is discussed in III-A, while we fixed $b = 16$ in light of [8]. A vector of b bits is then computed according to the rule:

$$F_{n,m} = \begin{cases} 1 & \text{if } \alpha_{n,m} - \alpha_{n-1,m} > 0 \\ 0 & \text{otherwise} \end{cases}$$

where $F_{n,m}$ (for which $0 \leq m \leq (b - 1)$) is the m -th bit relative to frame n and α is the energy difference among contiguous sub-bands, namely $\alpha_{n,m} = E_{n,m} - E_{n,m+1}$. The whole input is processed one segment at a time and eventually we obtain a sequence of bit vectors, which is the actual fingerprint of the given piece of audio.

B. Basic match algorithm

As stated in II, the same algorithm is used both to build a reference database and to process an unknown excerpt. In the former case, full songs are processed, thus storing fingerprints much longer than those computed in the latter case. There comes the non-trivial task of establishing whether these are matched by some of the available references, possibly providing some reliability indication.

The bit matrix form induces to use the normalized Hamming distance as distance metric. Basically, the fingerprint to be identified is “slid” along all reference fingerprints, evaluating the distance for each possible alignment. Various strategies can be successfully applied to significantly shorten the time required for searching [7] [8], an essential point for real-world applications. Here we are far more concerned with the reliability of the algorithm, so we chose to stick to the brute-force solution. Nevertheless, the value $b = 16$ suggested to halve computation times, in exchange for little robustness, by comparing 32bits at a time.

While a minimum distance approach can always give a match, it is advisable to choose a distance threshold $T < 0.5$ beyond which the answer should be discarded as unreliable. More than one threshold would likewise correspond to different degrees of reliability. Heuristic considerations [7] suggest a single $T = 0.35$, which we verified [8] can be often regarded as appropriate. A high T will more easily provide an answer, but it will also lead to an increased false positive rate, which is unacceptable in some scenarios. On the other hand, too small a T could greatly weaken the usefulness of the system. These observations will be clearer after the presentation of the experimental results in IV. For a more insightful discussion, we indeed ranked the first 10 proposed matches, and not only the first one.

III. THE PITCH-SHIFT ISSUE

The fingerprint algorithm described in II-A was proved robust against many kinds of signal degradations (see again [7] [8]). Nonetheless, as will be shown in IV, pitch-shifted audio poses a serious challenge for the system. In this paper, we speak of “pitch-shifted” audio when the pitch of the audio signal is raised or lowered, without affecting the tempo (time scale), i.e. the length of the piece is unchanged. Note that this is more than a simple frequency shifting, because it also involves a dilation in the frequency domain, in order to preserve the musical relationship of the harmonics [11].

The pitching issue can be commercially relevant, especially when monitoring ambient music, since it is not uncommonly found. For example, it may be needed in order to “melt” a sequence of various pieces, even if in these cases the distortion involved is usually not too strong, e.g. a couple of semitones (see IV for the definition) towards higher or lower frequencies. However, for testing purpose, we tried pitch shifts as high as 8 semitones, added or subtracted. In the following section we present two alternative solutions to face the problem, discussing their virtues and limitations.

A. A novel search strategy

The standard search approach computes the normalized Hamming distance between the fingerprint of an unknown excerpt and that of an equally-long sub-fingerprint. The latter is taken from the database, either trying each possibility in turn or according to an eventual optimization algorithm. This does not therefore substantially alter the basic behaviour, for it only provides smart starting points.

Taking up the picture of the unknown fingerprint slid along the time dimension of the database, we propose to extend the search to the frequency dimension, thus shifting by one or more bits the fingerprint of the excerpt prior to the comparison. Depending on the expected magnitude and direction of the distortion, the search should be appropriately limited, in order to save time. Since a frequency shifting is involved in pitching, we will show that this method allows to greatly improve the robustness of the recognizer, at least for not excessively strong distortions.

We can also consider a dual point of view and try to “de-pitch” the excerpt prior to the extraction of its fingerprint. Referring to the overall system, this filtering would fall into the pre-processing stage

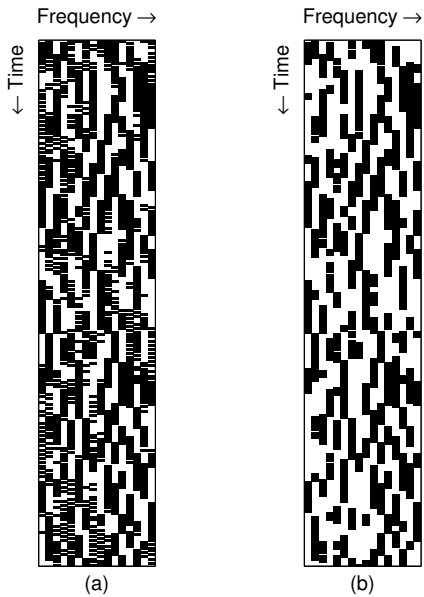


Fig. 3. The first 256 16-bit vectors of a sample fingerprint, represented before (a) and after (b) the mrl pre-processing, with $mrl = 3$. Note that $mrl = 1$ corresponds to an unprocessed fingerprint.

introduced in II-A. More efficiently, it should be performed after the STFT computation, whose large time-overlap parameter will greatly benefit the accuracy of the pitching operation [11].

On the positive side, this approach always compares full b -bit words, but in turn it requires a complex, non-negligible computational effort. Conversely, performing a bit-shift operation is simple and almost instantaneous, although the number of actually compared bits is linearly reduced by the magnitude of the bit shift.

There is another, more subtle issue, due to the choice of the frequency domain quantization in the fingerprint algorithm. At the fingerprint level, it actually maps a bit shift to a fixed frequency shift, so it accounts for at least two major drawbacks in the fingerprint-shift approach. Firstly, it allows to correctly handle pitch distortions only if their magnitude falls about evenly on quantization boundaries and secondly, it poses a limit on the maximum tolerable distortion. In other words, the bit shift cannot be too large, in order to retain a significant fraction of the available information.

B. Improved search

In Fig. 3a we report a sample, short fingerprint. Because of the high temporal correlation between the overlapping segments of the input, the bit matrix is mainly constituted by runs of 0s and 1s. We note anyway some “undecided” regions, where 0s and 1s tend to alternate. The negative impact on the overall system, in the long run, is twofold: not only does it lead to an increased distance between excerpts and possibly matching references, but also limits the robustness in case of pitch-shifted audio.

These thoughts suggested us to add little processing to both the unknown and the reference fingerprints, in order to keep only runs of a minimum configurable length mrl , as depicted in Fig. 3b. After this extra stage, the operation follows as usual with the distance metric computation. We will show that this caution allows, to a certain degree, to obtain an average inferior distance in case of match. At the same time, it does not reduce too much the distance with respect to non-matching entries. Finally, we point out that it does not cause the database to be rebuilt.

IV. TEST RESULTS

A main database of almost 15 000 pieces was built. For convenience, however, it was partitioned into two sections: the smaller comprises about 4 500 pieces by Italian performers only, while the remaining 10 500 are all relative to non-Italian, western music. They will be respectively denoted by IT and NIT. The rationale is that some kind of extra information is often available, for example whether the pieces are by local or foreign artists. A vast choice of genres was selected for both IT and NIT.

Following our discussion at the end of III-A, we actually computed two fingerprints for each piece, so to evaluate the behaviour of the system with respect to two different band quantizations. The most straightforward approach is to log-equally divide a meaningful bandwidth, justified by the fact that the human ear has an approximate logarithmic response. The base used for the logarithm is 10 and the performed subdivision will be denoted by EQL.

Another solution is that of exploiting a quite often used tuning system, at least as far as the western world is concerned: equal temperament, or 12-TET. Leaving out the historical reasons which drove his adoption, in this paper we only point out that it is built from a basic tone in the neighbourhood of 440Hz (called A4 or La4). In our case, this exact value was used. Each of the 12 available notes is then obtained by multiplying (or dividing) the frequency of the adjacent tone by a factor of $2^{1/12}$ and this interval is called “semitone”. The iterated procedure leads to the musical scale (C, C \sharp , D, ... B) or (Do, Do \sharp , Re, ... Ti), probably known to the reader, where C \sharp and D \flat (and so on for analogue couples) have actually the same frequency. The interval between a tone and its doubled-frequency counterpart is called “octave”, and is such that we perceive two sounds an octave apart as having the same pitch. In conclusion, 12-TET is a tuning system where we can perceive as much as 12 unique pitches. Note that 12-TET is not related to the Bark scale.

In the EQL case, the spanned bandwidth can be freely chosen regardless of b and we selected $300 \div 2000$ Hz, according to [7]. On the contrary, given $b = 16$, the rules at the basis of 12-TET fix the preservable bandwidth. In order to include the most meaningful fraction, we chose the interval $740 \div 1976$ Hz, i.e. F \sharp 5 \div B6, thus spanning an octave and a third.

A. Important parameters and terminology

The tests were carried out on both undistorted and pitch-shifted excerpts, whose duration is as low as 3.333s, corresponding to exactly 256 16-bit vectors. The pitch-shifted excerpts are all randomly extracted from a couple of random pieces (one for IT and one for NIT), whose pitch was manually altered using the open source tool *Audacity* [12]. These pieces were also fingerprinted, in their entirety, into the database. The pitch shifts tested varies from -8 to $+8$ semitones, with a step of 1 semitone, and involve an approximate percent frequency displacement summarized by table I. On the basis of our previous discussion, a 12-semitone pitch shift, i.e. an octave, would correspond to a 100% frequency shift.

In these sections we speak of “success” when the recognizer gives as first match the correct reference, while in IV-C we will show up to which degree is this information reliable enough (threshold method). The “success rate” is thus computed by averaging over the number of trials, at least 50 per each point. By “distance in case of success (error)” we mean the normalized Hamming distance between the excerpt and the (non) matching sub-fingerprint extracted from the appropriate database and proposed as first match, while the “2nd choice” distance is representative of the second-ranked match. Moreover, we recall that the acronym “mrl” stands for “minimum run-length”, whose role has been discussed in III-B. Finally, when we

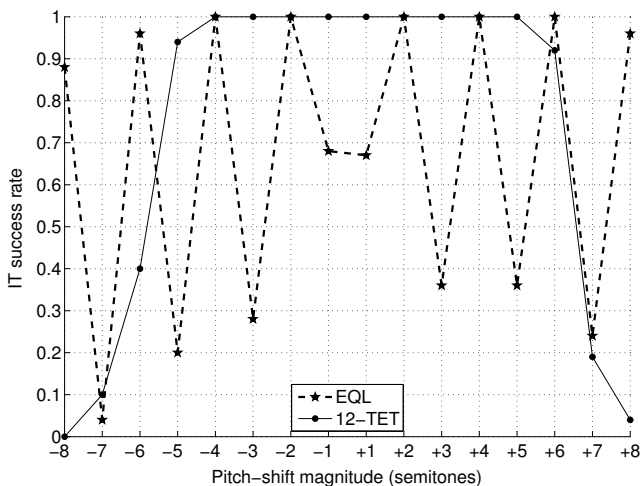


Fig. 4. **IT** database success rate for different band quantizations, as a function of the applied pitching (semitones).

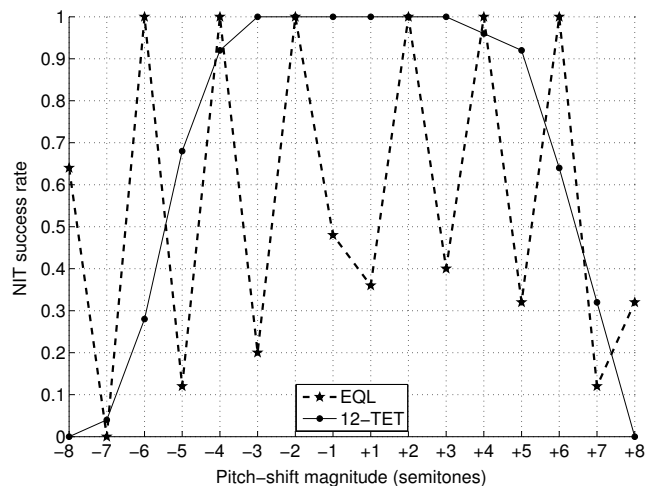


Fig. 5. **NIT** database success rate for different band quantizations, as a function of the applied pitching (semitones).

talk in terms of distortion magnitudes, without further specifications about its sign, both positive and negative pitch shifts are implied.

B. Success rate

First of all, we carefully tested both databases (IT and NIT), verifying a thorough reliability in the undistorted case, with a success rate in excess of 99.9%, regardless of the band quantization chosen. For practical uses, we can say that every one of the 15 000 pieces was correctly identified with respect to its relative database, and only by means of 3.333s random excerpts. A number of ambient, noiseless recordings were also made and tried in the same way with similar results. A standard, home-quality microphone was used, directly connected to the PC.

In general, we observed that the average distance for a correct match in the EQL case is regularly $6 \div 7\%$ lower than that computed for the 12-TET case. Indeed, they are about 0.15 and 0.22 respectively, well below the proposed 0.35 threshold. Should no distortions occur, EQL can be a better choice.

If we add some pitch shift, though, the outcome may be much different. The key point is that the basic search strategy, being it optimized or not, is unable to identify even slightly pitch-shifted excerpts. Extremely rare hits may be seen in the EQL case for 1 semitone distortion, but none in the 12-TET case. We now show exceptionally good improvements thanks to the search algorithms presented in III-A.

Let's first focus on the fingerprint-shift search strategy. As we can see in Figs. 4 and 5 (for IT and NIT respectively), a careful choice of the sub-bands is essential. In the EQL case, the behaviour alternates between an excellent success rate and a very poor one. On the other hand, 12-TET performs very well up to quite a strong pitch shift. With

+1	+2	+3	+4
+5.95%	+12.24%	+18.92%	+25.99%
+5	+6	+7	+8
+33.48%	+41.42%	+49.83%	+58.74%
-1	-2	-3	-4
-5.61%	-10.91%	-15.91%	-20.63%
-5	-6	-7	-8
-25.09%	-29.29%	-33.26%	-37.00%

TABLE I
PITCHING/FREQUENCY-DISPLACEMENT APPROXIMATE MAPPING.

the smaller database, we note a steep drop-off beyond a variation of 4 or 5 semitones, possibly due to the significant frequency dilation involved (compare III). A similar trend is visible in the NIT case, but slightly anticipated in respect of a growing distortion magnitude.

On the contrary, the “de-pitch” approach, if perfectly performed, leads to quasi-perfect results in terms of success rate, even for deviation of 8 semitones. This is comparable to the undistorted case and thus not graphically reported. The only drawback is a significantly increased distance, which is at most in the neighbourhood of an additional $5 \div 6\%$. Note also that the impact of the particular algorithm used for the pitching and the de-pitching may be non-negligible.

If the exact pitch shift is not known or guessed, though, we are exactly in the same case of having a pitch-shifted excerpt and employing the basic search strategy. That is, without a very good estimation of the pitch shift involved, or without enough (and computationally expensive) trials and errors, the “de-pitch” solution is quite useless. A concerted strategy could be thought of, but at a great expense in terms of search time. From now on, only the bit-shift approach will be brought into discussion.

A further improvement may be often attained by pre-processing the interested bit matrices prior to the distance evaluation, in order to keep only runs of 1s which have a minimum fixed length. This is clearly highlighted by Figs. 6 and 9 (note the middle-compressed scale, for clarity), where the curves suggested that the best values for mrl falls within the interval $2 \div 4$, slightly in favour of 2. Similar conclusions can be drawn from the dual Figs. 7 and 10, which focuses instead on the dependence from mrl , where curves for lower distortions are omitted, being almost constant to 1. Again, the NIT database appears less robust than IT.

Overall, we observe a high symmetry in the presented results, with respect to the sign of the distortion. Nonetheless, as it grows, the recognizer appears to be more robust against positive pitch shifts, being +1 the approximate centre of symmetry. We also point out that the relationship between robustness and frequency deviation is not trivial, as we can see by referring to table I: a higher value actually leads to a better success rate in case of 12-TET. Analogue considerations will hold when the distance metric is involved. Even if a larger database loses in success rate from a few percent to more than 20%, its performance is anyway comparable for low to medium pitch shifts.

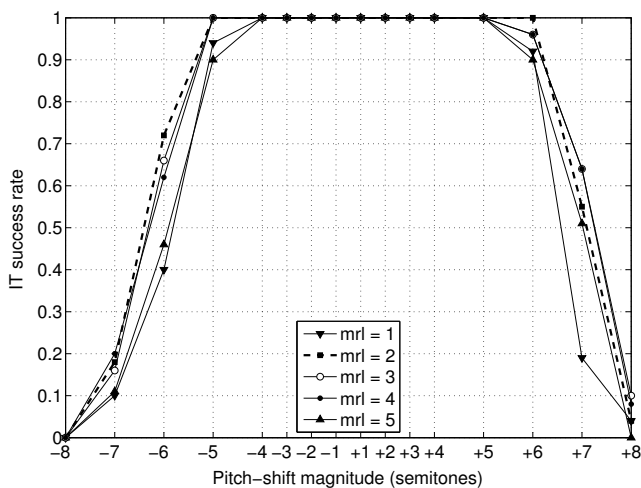


Fig. 6. IT database success rate as a function of the applied pitching (semitones) and mrl (bits).

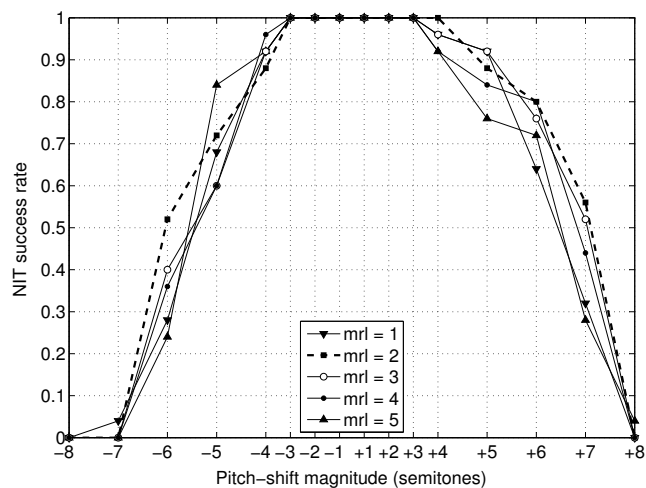


Fig. 9. NIT database success rate as a function of the applied pitching (semitones) and mrl (bits).

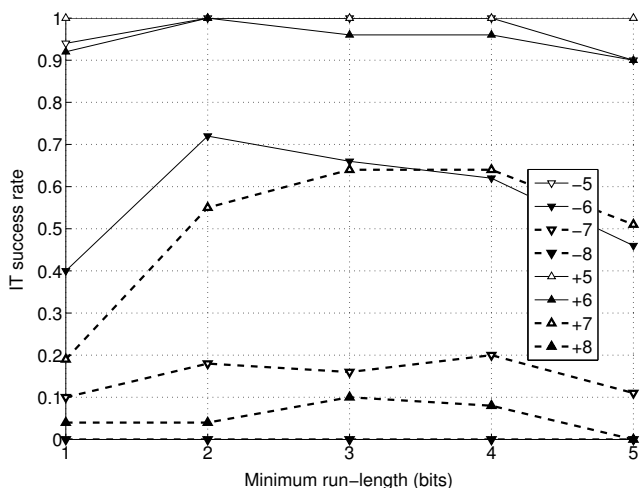


Fig. 7. IT database success rate as a function of mrl . Only lowest-successful pitching values (semitones) are shown.

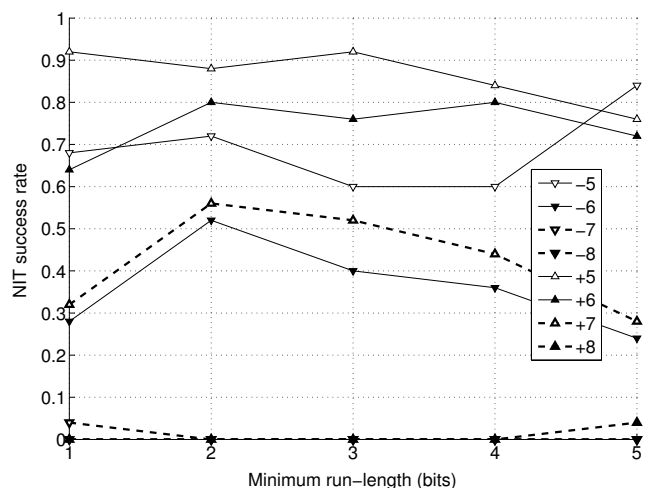


Fig. 10. NIT database success rate as a function of mrl . Only lowest-successful pitching values (semitones) are shown.

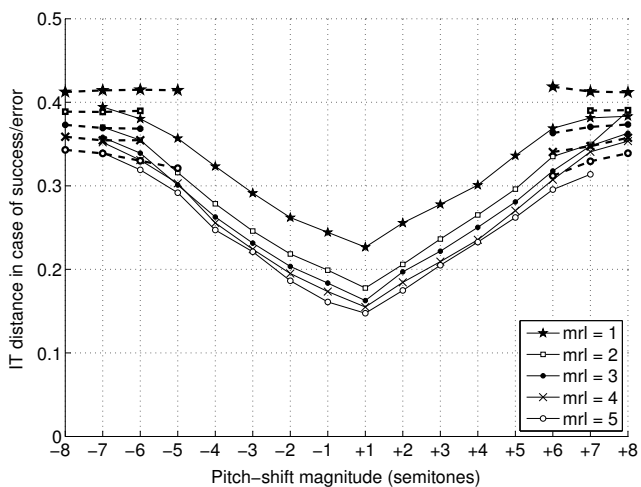


Fig. 8. IT database average distance in case of success (solid line) or error (dashed line), as a function of mrl .

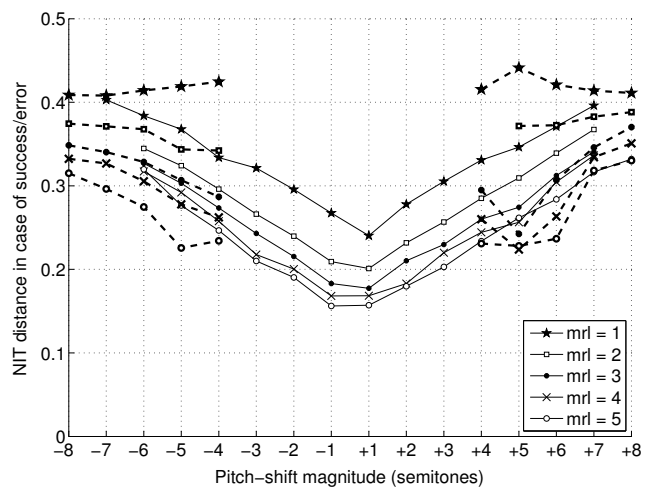


Fig. 11. NIT database average distance in case of success (solid line) or error (dashed line), as a function of mrl .

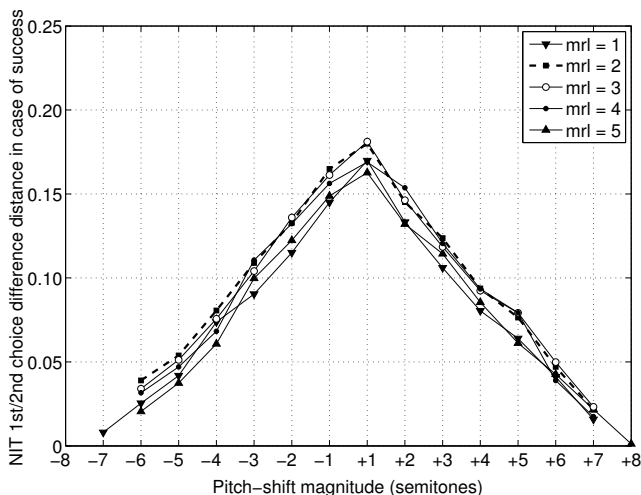


Fig. 12. NIT database average distance difference between first- and second-ranked proposed matches, when the first is successful.

C. Computed distance

A most important metric to investigate is the average distance computed in case of success (ds) or error (de). Indeed, both can help in fixing (a) reliability threshold(s) and are a soft indication of the overall robustness of the recognizer. In Figs. 8 and 11, we observe that ds varies with a roughly linear law, depending chiefly on the magnitude of the distortion, though positive pitch shifts lead always to inferior, better values. It therefore behaves similarly to the success rate metric, to which it is closely connected.

If the distortion grows considerably, it is possible to estimate also de , for errors occur. In these regions, we observe that the trend of the larger database, namely NIT, heavily departs from that of the smaller, i.e. IT. In particular, in the IT case ds smoothly tends towards de , which appears somewhat constant. On the contrary, when NIT is used, de can even be significantly inferior to its ds counterpart. Once a threshold is fixed, say $T = 0.35$, this can be really troublesome, since we cannot discern a reliable, below-threshold correct match from an incorrect guess.

Caution should then be taken with respect to the choice of mrl , even if it does help in lowering the evaluated distance. This is clear in the presented figures, so we did not report the dependence of the metric from mrl , which is constituted by slowly-decreasing, quasi-parallel straight lines, as mrl grows. Hence, and enlightened by IV-B, $mrl=2$ can be regarded as very appropriate, since it improves both the distance and the overall success rate, but without falling into the difficulties discussed. Moreover, the gain in distance is particularly high when passing from $mrl=1$ (i.e. no pre-processing) to $mrl=2$, while it becomes less and less noticeable in subsequent steps, so there is no urge to further increase the value.

We recall that our system does not propose one single match, but tracks down instead the first, best 10 matches. It is thus possible, in particular, to evaluate the distance difference between the first and the second choice, as depicted in Fig. 12 (NIT). The IT case, not reported here for lack of space, is fully similar, but the metric is slightly higher (and thus better), with an improvement in the neighbourhood of some percent.

As expected, for weaker pitch shifts both cases present substantially spaced values, while they tend to smoothly converge as distortion grows. Though not graphically represented, we also observed that all non-first proposed matches, i.e. from the second to the tenth, were almost equal in distance and appeared as randomly picked. In

conclusion, as far as mrl is concerned, $mrl=2$ appears by far the best choice, supporting our previous feelings on the topic. By evaluating the dependence from mrl , we drove similar considerations.

V. CONCLUSIONS AND FUTURE WORK

Starting from the difficulties involved in automatically recognizing pitch-shifted audio, we proposed two effective solutions. They are based on a robust and thoroughly tested fingerprint algorithm, partly modified to meet the new requirement, and on a reference database. Since our adaptation is especially relative to the search strategy, our approach allows the use of an eventual, already-built database.

In reference to the specific fingerprint algorithm, we discussed the choice of the sub-bands, noting their key role in the effectiveness of the proposed solution. In particular, if the pitch-shift distortion is not homogeneous with respect to the band subdivision, the success rate may be very low. On the contrary, a matched subdivision results always in a very good performance, at least when the pitch shift is not too strong. Finally, we showed the benefits of an additional processing stage on the extracted fingerprints, which can come at little computational costs. All tests were performed on a large MP3 collection.

Future work will focus on optimizing the search strategy in the pitch-shift case and evaluating the robustness of the method when intermediate pitch shifts occur, even if we are fairly optimistic on the basis of preliminary trials. A further step will be that of theoretically analyse the computational complexity of the proposed solutions, in order to provide more motivated cost/benefit trade-offs.

REFERENCES

- [1] A. Sinitsyn, "Duplicate song detection using audio fingerprinting for consumer electronics devices", *IEEE 10th International Symposium on Consumer Electronics*, St. Petersburg, Russia, June 2006.
- [2] G. Velickic, E.L. Titlebaum, M.F. Bocko, "Musical note segmentation employing combined time and frequency analyses", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004.
- [3] P. Cano, E. Batlle, T. Kalker, J. Haitsma, "A review of algorithms for audio fingerprinting", *International Workshop on Multimedia Signal Processing*, US Virgin Islands, December 2002.
- [4] C.J.C. Burges, J.C. Platt, S. Jana, "Extracting noise-robust features from audio data", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, USA, May 2002.
- [5] A. Ramalingam, S. Krishnan, "Gaussian Mixture Modeling Using Short Time Fourier Transform Features for Audio Fingerprinting", *IEEE International Conference on Multimedia and Expo*, Amsterdam, The Netherlands, July 2005.
- [6] P. Cano, E. Batlle, H. Mayer, H. Neuschmied, "Robust Sound Modeling for Song Detection in Broadcast Audio", *Proceedings of the 112th AES Convention*, Munich, Germany, May 2002.
- [7] J. Haitsma, T. Kalker, J. Oostveen, "Robust audio hashing for content identification", *International Workshop on Content-Based Multimedia Indexing*, Brescia, Italy, September 2001.
- [8] C. Belletini, G. Mazzini, "On audio recognition performance via robust hashing", *International Symposium on Intelligent Signal Processing and Communication Systems*, Xiamen, China, November 2007.
- [9] D. Jang, S. Lee, J.S. Lee, M. Jim, J.S. Seo, S. Lee, C.D. Yoo, "Automatic commercial monitoring for TV broadcasting using audio fingerprinting", *AES 29th International Conference*, Seoul, Korea, September 2006.
- [10] A. Ribbrock, F. Kurth, "A full-text retrieval approach to content-based audio identification", *IEEE Workshop on multimedia signal processing*, St. Thomas, Virgin Islands, USA, December 2002.
- [11] <http://www.dsdimension.com>
- [12] <http://audacity.sourceforge.net>
- [13] <http://www.knowmark.it>