University of Padua at MusiCLEF 2011: Music Identification Task

Emanuele Di Buccio, Nicola Montecchio and Nicola Orio

Department of Information Engineering, University of Padua, Italy
{emanuele.dibuccio,nicola.montecchio,nicola.orio}@dei.unipd.it

Abstract. This paper reports on the participation of the Information Management System Research Group of the University of Padua to the Music Identification task of the MusiCLEF Laboratory in CLEF 2011. The system under evaluation is FALCON, an open-source engine for content-based cover song identification written in Java that applies classic techniques derived from textual Information Retrieval to music identification. The obtained results show how such approach yields satisfying results using little computational resources.

1 Introduction

Automatic identification of music documents has become an essential component of many popular web services. Audio fingerprinting techniques are widely used in order to identify copies of a recording, which can differ from the original because of data compression, A/D conversion or environmental noises [1]. In most major video sharing websites, the background music of user generated videos is automatically identified in order to either remove the video (for copyright issues) or suggest where to purchase the original music (advertising)[1].

Audio fingerprint approaches aim at the identification of a particular *performance/recording* rather than *work*: even alternate takes of a composition are considered different items instead of being regarded as different instances of the same piece. This assumption imposes significant restrictions to the range of possible differences between copies of the same recording, allowing audio fingerprinting techniques to be particularly efficient.

Aside from audio fingerprinting techniques, past research on music identification has focused, for obvious commercial reasons, on popular music; the research field is commonly referred to as *cover song* identification, a more general name for this research field being *version identification* [2]. The problem however is also of interest for other genres: in classical music there is especially a vast number of interpretations of the same work and version identification technology can be beneficial for many music libraries and archives that aim at the preservation and dissemination of classical music. For a comprehensive review of previous approaches to the version identification problem the reader can refer to [2], where

[1] See, for instance, http://www.youtube.com/t/copyright_my_video

the author provides a review based on a functional block decomposition of previously proposed systems.

The experience gained in the development of text search engines shows that in most cases simple and efficient techniques can be generally employed in various retrieval tasks, with results that are often comparable with more complex and less efficient approaches. Following this idea, we developed a music identification engine named FALCON[2]. The fact that our software is implemented on top of Apache Lucene[3], an open source text search engine, substantiates the claim that the problem of music identification can be modelled as a more general retrieval task. In this paper we study its adaptability to classical music, making use of the dataset prepared for the *Music Identification* task of the MusiCLEF 2011 benchmarking activity. The collection is constituted by circa 7 thousand recordings for which the MusiCLEF Laboratory organizers provided precomputed descriptors. Testing FALCON on this test collection has allowed us to investigate the scalability of our proposed approach, thus extending previous investigations carried out on popular music [3].

2 Methodology

Our approach is based on a two-level bag-of-features hierarchy; the input descriptors (chroma features) are transformed into hashes which are subsequently grouped into a "set of hash sets" representation for each recording. The cardinality of set intersection is adopted as a similarity measure. The methodology, described in detail in [3,4], can be summarized as follows.

audio content analysis - this step consists in: (i) *chroma feature extraction* from audio waveforms, (ii) *key-finding* — the most probable keys of a recording are estimated in order to preserve transposition invariance — and (iii) *hashing* of the transposed chroma features into a sequence of integer values which form the output of this phase.

indexing - each sequence of hashes is segmented into a set of possible overlapping segments. Hashes are interpreted as terms in a textual document and segments as passages constituted by sets of terms. This representation can be easily stored in an inverted index: the set of all the distinct hashes appearing in the sequences obtained from the recordings in the collection constitutes the index vocabulary; each item in the posting list is associated to an hash and retains information about the frequency of occurrence of the hash in a specific segment.

querying - the similarity between a query Q and a recording D is computed as

$$S(Q,D) = \sqrt[|Q|]{\prod_{q \in Q} \max_{d \in D} \Big\{ \sum_{t \in q \cap d} \min\Big(\frac{\mathrm{hf}(t,d)}{|d|}, \frac{\mathrm{hf}(t,q)}{|q|} \Big) \Big\}} \qquad (1)$$

[2] http://ims.dei.unipd.it/falcon
[3] http://lucene.apache.org/

where q and d denote two segments of a query and a collection recording, respectively of length (number of hashes) $|q|, |d|$; similarly, $|Q|, |D|$ denote the number of distinct segments for Q, D and $\mathrm{hf}(t, d)$ denotes the frequency of the hash t in the segment d. Formula 1 can be interpreted as follows: the first step (inner summation) computes local similarity between segments as the (normalized) number of terms they have in common; the second step aggregates the contributions of all the query segments, by computing the geometric mean of the best local similarities.

The procedure is repeated multiple times according to the most probable keys detected in the audio analysis step; for each recording in the collection, the highest similarity value is preserved.

3 Experiments

3.1 Test Collection and Setup

The test collection comprises 6679 recordings of classical music works, which add up to more than 572 hours of music. Of these recordings, 2,671 are associated to works that are represented at least twice in the data base, forming 945 *cover sets*[4]. The audio descriptors were extracted using the MIRToolbox package [5]. More details on the collection, and in general on the MusiClef campaign, are available in [6].

All experiments were repeated twice, one time using a key-finding algorithm to preserve independency to transposition, and one time without such strategy. The other parameters were set according to the values reported in [3], except the segment overlap which was set to 50% the length of each segment (15s).

3.2 Identification Accuracy and Computational Load

In FALCON, the trade-off between accuracy and speed of retrieval privileges the latter, as the architecture was designed with large collections in mind. It is therefore important to measure the computational resources required by the software. All experiments were run on a machine with a 3.4 GHz dual-core processor (4 logical cores), 24 GB of RAM and a 7200 RPM hard disk.

Table 1 shows the results of our experimentation. As can be seen, the software can index about 3.5 hours of music per second and is able to perform a typical query in under 3 seconds. The average query time is different because the key-finding algorithm repeats a query multiple times (in this case 3) in parallel; each thread is completely independent of the others.

[4] The term "cover set" is commonly used to define a set of recordings of the same piece of music.

Table 1: Accuracy and timing for 667 queries over 6679 recordings.

	no transposition	with transposition
total indexing time	163s	166s
average query time	1.79s	2.32s
MAP	0.6754	0.6826
MRR	0.7617	0.7771

4 Concluding Remarks

This paper reports our contribution to the Music Identification task of the MusiCLEF Laboratory in CLEF2011. Our experimental results show how a simple approach based on text retrieval techniques can yield satisfying results using little computational resources.

These results suggest that the methodology implemented in FALCON can be adopted as the first of a two-step methodology. The identification performed by FALCON would aim at providing candidate versions of the query recording at the top k rank positions. Then identification could be refined by means of more sophisticated but also more resource consuming music alignment techniques, that when exploited in isolation and on the entire collection are hardly scalable.

References

1. Cano, P., Batlle, E., Kalker, T., Haitsma, J.: A review of audio fingerprinting. Journal of VLSI Signal Processing Systems **41** (November 2005) 271–284
2. Serrà, J.: Identification of versions of the same musical composition by processing audio descriptions. PhD thesis, Universitat Pompeu Fabra, Barcelona (2011)
3. Di Buccio, E., Montecchio, N., Orio, N.: A scalable cover identification engine. In: Proceedings of the international conference on Multimedia. MM '10, New York, NY, USA, ACM (2010) 1143–1146
4. Miotto, R., Orio, N.: A music identification system based on chroma indexing and statistical modeling. In: Proceedings of the 9th International Conference on Music Information Retrieval. ISMIR (2008) 301–306
5. Lartillot, O., Toiviainen, P.: A matlab toolbox for musical feature extraction from audio. In: Proceedings of the 10th International Conference on Digital Audio Effects, Bordeaux, France. (2007)
6. Orio, N., Rizo, D., Lartillot, O., Miotto, R., Montecchio, N., Schedl, M.: Musiclef: A benchmark activity in multimodal music information retrieval. In: Proceedings of the 12th International Conference on Music Information Retrieval (to appear). ISMIR (2011)