Duplicate detection for for digital audio archive management: two case studies

Joren Six[1], Federica Bressan[2], and Koen Renders[3]

[1]IPEM, Ghent University, Belgium
[2] Stony Brook University, New York, USA
[3] VRT, Brussel, Belgium
joren.six@ugent.be

Abstract. This chapter focuses on identification of duplicate audio material in large digital music archives. The music information retrieval (MIR) problem to efficiently find duplicate in large collections is a solved problem. There are even off-the-shelf systems available to find duplicates. The applications of this technology, however, are still too unknown and underexploited.
This chapter describes duplicate detection and its many applications which include: meta-data quality verification, improving listening experiences, re-use of meta-data, informed noise cancellation, optimising storage space, and linking and merging archives. The applications of duplicate detection are illustrated with two case studies.

1. The first case study uses a collection of digitised shellac discs from the Belgian national public-service broadcaster. It shows a surprisingly high amount of duplicate material of around 38%. With some discs better preserved (and digitized) than others, linking duplicate material allows to redirect listeners to higher quality audio.
2. An archive of early electronic music is the focus of a second case study. The archive has been digitized twice. Segmentation timestamps and other meta-data, originating from first digitisation campaign, is reused to annotate higher fidelity digital audio from a second campaign.

The main contribution of this chapter is to highlight practical uses of duplicate detection. A secondary contributions are the findings detailed in the case studies. A third contribution is an evaluation of an updated fingerprinting system.

Keywords: Music Information Retrieval, Duplicate Detection, Acoustic Fingerprinting, Music Archives,Sound archive management, Case studies, Field recordings, MIR applications

1 Introduction

Many Music Information Retrieval (MIR) technologies have untapped potential. With every passing year the MIR-field presents promising technology and research prototypes [9,16]. Unfortunately, these academic advances do not translate well to practical use. For digital music archive management especially, MIR techniques are underexploited [5]. We see several reasons for this. One is that MIR technologies are simply not well known by archivists. Another reason is that it is often unclear how MIR technology can be applied in a digital music archive setting. MIR-researches see applications as 'self-evident' while a translation step is needed to enthuse end-users. A third reason is that considerable effort is needed to transform a potentially promising MIR research prototype into a working, documented, maintained solution for archivists.

In this chapter we focus on duplicate detection. It is an MIR technology that has matured over the last two decades [29,11,23,8,26]. It is telling that an overview paper [3] of about 15 years ago is still relevant today. Duplicate detection is regarded as a solved problem and the focus of the MIR-community shifted from active research to refinement. A broader impact and application of the technology, however, remains marginal outside big tech. Some of the applications of duplicate detection might not be immediately obvious since it is used indirectly to complement meta-data, link or merge archives, improve listening experiences, synchronization[24] and it has opportunities for segmentation.

The aim of the is chapter is threefold. The first aim is to be explicit about the use of duplicate detection for music archive management. The second aim is to present two case studies using duplicate detection: one determines the amount of unique material in an archive, the other case study is on reuse of meta-data. The third aim of this chapter is to present and evaluate the improved duplicate detection system that was used for the case studies. Let's first define duplicate detection.

1.1 Duplicate detection

The definition of duplicate detection in music is less straightforward than it seems at first due to many types of reuse. In [17] there is a distinction made between *exact*, *near* and *far* duplicates. For an exact duplicate the duplicate contains exactly the same audio as the original. Near duplicates are different only due to technical processing: e.g. by using another lossy encoding format, remastering or compression. Different recordings of the same live concert are also in the near duplicate category. Far duplicates span a whole range of reuse of audio material: samples, loops, instrumental versions, mashups, edits, translations and so forth.

Going even further, covers of songs reuse musical material of the original. In all cases a musical concept is shared between the original and the cover, but only in some cases audio is reused. There is a whole range of different types of covers: live performances, acoustic versions, demo versions, medleys, a remixes are only a few examples. A more complete overview can be found in [19]. In this work we focus on duplicates which contain the same audio material as the original.

This includes samples or mashups but excludes live versions which do not share audio with the original version, although they might sound similar.

The duplicates of interest share audio material, however duplicates should still be identifiable even when the audio is slightly changed. Evidently a match should still be found if volume has changed. Other modifications such as compression, filtering, speed changes, pitch-shifting, time-stretching, and similar modifications should be allowed as well. We and up at the following definition:

> *Duplicate detection allows to compare an audio fragment to other audio to determine if the fragment is either unique or appears multiple times in the complete set. The comparison should be robust against various modifications.*

Assuming a duplicate detection system is available, several applications become possible [17,21]. Such system is used in the following case studies. After the case studies a more technical part and evaluation follows which shows the limitations and strengths of the system actually used.

2 Case 1: Duplicates in the VRT shellac disc music archive

2.1 The VRT shellac disc archive

The VRT shellac disc archive is part of the vast music archive of the Belgian public broadcasting institute. The archive contains popular music, jazz and classical music released between 1920 and 1960 when the public broadcaster was called INR/NIR (Institut National de Radiodiffusion, Nationaal Instituut voor de Radio-Omroep). The total shellac disc archive contains about 100,000 discs of which a selection was digitized. Unique material and material with a strong link to Belgium was prioritized. Basic meta-data is available (title, performer, label) but, notably, a release date is often missing.

The digitized archive contains 15,243 shellac discs. Each disc has a front and backside, which are often not clearly labeled. Both sides of each disc are digitized at 96Khz/24bit with a chosen EQ curve without further post-processing. The digitization effort was led by Meemoo[1] which also provides long-term storage.

2.2 Determine unique material in an archive

The meta-data suggests that the shellac disc archive contains a significant portion of duplicate material. However, due to nature of meta-data, it is often

[1] *From the* `http://meemoo.be` *website: 'Meemoo is a non-profit organisation that, with help from the Flemish Government, is committed to supporting the digital archive operations of cultural, media and government organisations'.*

unclear whether the meta-data describe the exact same audio material or it describes a different rendition of the same song. The main problem is to **conclusively determine the amount of unique audio material in the archive**, or conversely the amount of duplicate material.

There are two opportunities by linking duplicate material. The first deals with sound quality. Some discs are better preserved (and digitized) than others. Linking duplicate material makes it possible to a potentially redirect archivists and listeners to a better preserved duplicate.

A second opportunity relates to meta-data quality. It is of interest to identify when meta-data is inconsistent and how the current meta-data standards for this archive and could be improved.

2.3 Detecting duplicates

The archive consists of 30,661 digital files with an average length of 168±5s. Duplicate detection found 11,829 files or 38.6% contain at least 10s of duplicate material. Most are exact duplicates but some are translations (see below).

The meta-data quality is quite high. If the title of a duplicate is compared with the original, 93% match using a fuzzy matching algorithm. To allow slight variations - differences in case use, additional white space, accents, word order - a Sørensen–Dice coefficient is determined between the original and duplicate title. Only if the coefficient is above a threshold the match is accepted. The performer meta-data field fuzzily matches for 83% of the cases: it is sometimes left blank.

One notable meta-data inconsistency results from the fact that the side of each disc was not clearly labeled. The concept of an 'A' side and a 'B' side was not yet established. During digitization and meta-data notation this is problematic. The cover gives the title of two works but it is unclear on which side the work is located. The meta-data often assigns both titles to each digital file. Since the order is not clear, for a duplicate the order of the titles can reversed (the sides are switched).

Another more specific finding is that popular orchestral songs were released for multiple markets. The orchestral backing is the same but the sung part is translated, sometimes by the same singer. The practice of dubbing and re-releasing popular hits in another language is much less common now. An example is file *CS-00069022-01* by Jean Walter. One contains *Rêve d'un Soir/Toi Toujours*, the Dutch version is titled *Hou van Mij (Loving You)/Ik Had Een Droom (I Kissed A Dream)*. Note that the order is reversed and that the titles are translated freely. The Dutch version also refers to an English original. Discrimination between a noisy exact duplicate and a translated version is not possible by taking only into account the duplicate detection results.

Meta-data and classical music is generally problematic. Unfortunately, this is also the case for the shellac disc collection, which contains some classical works. It is clear that shoehorning classical composers, performers, soloists, works and parts into a title and performer framework is a source of many of the identified inconsistencies.

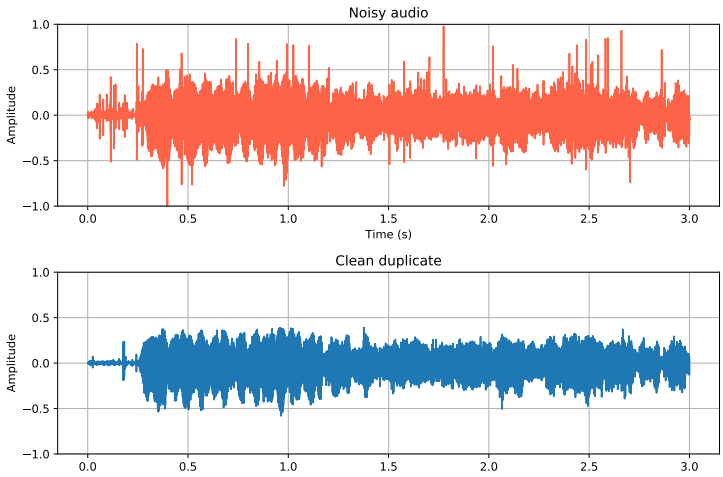

Fig. 1: A fragment from two digitized discs with the same audio material. The one below is clearly less affected by pops and cracks. Identifying and linking duplicates allows listeners to find the best preserved copy.

One of the opportunities of linking duplicates is shown in figure 1. It shows two discs with the same audio material. One disc deteriorated much more than the other. The amount of pops, cracks and hiss on the digitized version of the deteriorated disc makes it hard to listen to. The much better preserved disc resulted in more ear-friendly digital audio. To listen to the noisy and cleaner version, consult the supplementary material[2]. Going one step further is active denoising. There are technique which use duplicate material to interpolate samples from several sources with the aim to reconstruct and denoise gramophone discs [27]. Modern deep-learning techniques to denoise historic recordings [10] also show a lot of promise.

2.4 Some observations

This case study shows clear advantages of duplicate detection within an archive. Especially if the archive is expected to contain many duplicates. It is possible to link audio with others that contain the same recorded material. By post processing these lists of duplicates it becomes possible to:

- Identify the amount of unique material within the archive;
- Link low quality recordings to better preserved duplicates;
- Confront meta-data fields from a recording and its duplicates to get insights into meta-data quality;

[2] Supplementary material can be found at `http://0110.be/l/duplicates`. It contains audio examples, software and raw data on duplicates.

– Find surprising links between recordings that share the same recorded material but are clearly different. E.g. a shared orchestral backing of a which the sung part is translated.

The case study also showed a limitation of duplicate detection technology. Discriminating between a near exact noisy duplicate and a translated version of a song with the same orchestral backing is not possible using only the results of duplicate detection. Additional techniques: meta-data analysis, song-to-text conversion with language recognition might be employed to solve this automatically.

3 Case 2: Meta-data reuse for the IPEM electronic music archive

3.1 The IPEM music archive

The Institute for Psychoacoustics and Electronic Music (IPEM) has an archive of tapes originating from 1960-1980. IPEM is part of Ghent University in Belgium. To get an idea about the contents this archive, it helps to sketch a short history of IPEM and the broader context.

After the second world war, several European broadcasting cooperations started electro-acoustic music production studios. In these studios composers and engineers collaborated to create new instruments and sounds. One of the main drivers behind these investments was the belief that avant-garde music was the logical next step in the western art-music tradition. Two early examples are the 'Colonge Studio für elektronische Musik des Westdeutschen Rundfunks', Germany (1951) and the 'Studio di Fonologia Musicale RAI di Milano', Italy (1955). The insight that avant-garde music studios were not commercially viable warranted institutional backing. In this context BRT (Belgishe Radio en Televisie) and Ghent University ("Rijksuniversiteit Gent" at the time) jointly started IPEM in 1963.

The IPEM production studio was active between 1963 and 1987 [15]. It produced about 450 works of around 100 composers which ended up on about 1000 magnetic tapes [14]. A typical tape cover can be seen in Figure 2. In the late 1980s electronic music production became more and more accessible thanks to the introduction of cheap electronic instruments. The need for institutional backing started to appear anachronistic. After a difficult transition IPEM established itself as a research center in systematic musicology and is now part of Ghent University.

3.2 Merging two digitized archives

The archive has been digitized twice. The first digitization campaign was around 2001 [14]. This resulted in a database with high-quality meta-data and set of audio CD's. Unfortunately, the sound cards used at the time (SEK'D ARC 88, 16bit, 48kHz) yielded audio not up to standard for long term preservation. The

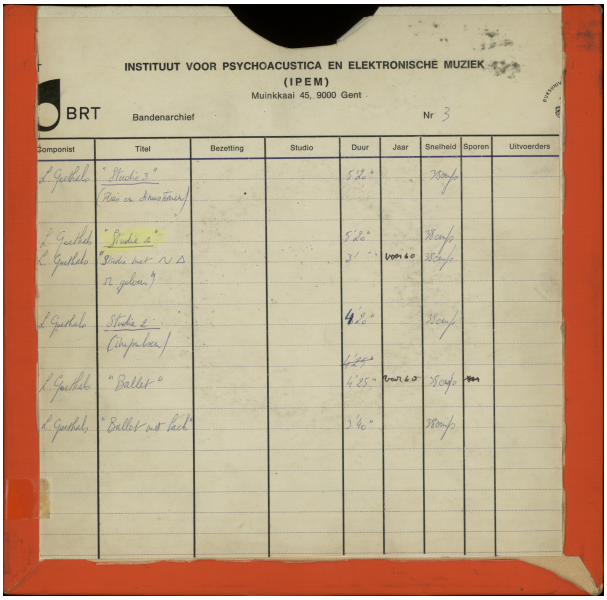

Fig. 2: The cover of a typical magnetic tape in the IPEM archive. The meta-data includes work titles, composers, additional comments and technical meta-data about the tape.

choice of writable CDs as long term storage media was also questionable, due to their limited shelf life. The meta-data, however, that was organized in a relational database is still relevant today.

The second digitization campaign was organized in 2016 by Meemo. Much better sound cards were available and digitization was done at 96kHz and 24bit. Long-term storage is done on redundant LTO8 magnetic tapes. However, less effort was spent on detailed meta-data. Complete magnetic tapes were stored in a single audio file together with a list of works on each tape. Each work has a title and a composer. Rough estimates of the duration of the work and additional meta-data (performers, context of the recording) are often missing. Notably, it is unclear where each work starts and stops in the unsegmented digital audio file.

Segmentation of the tapes with avant-garde electrouacoustic works is not always trivial. A large part of the tapes do have clear boundaries between works but for others, expert listeners and contextual information is required. This is the case, for example, for works with different parts without a shared sound language. Another example is where silences on tape should be filled by an acoustic instrument during performance (which is not recorded on tape). The meta-data on the duration of the work and the duration of audio on tape can differ widely in such cases.

The problem in the case of the IPEM music archive is to **connect high quality meta-data and segmentation of a previous digitization to high quality unsegmented audio of a second digitization**.

3.3 Meta-Data and segmentation reuse

With duplicate detection a link can be found between the low-quality audio and the high-quality audio. The detailed meta-data from the low-quality audio can then be attached to the high-quality audio. The segmentation timestamps for the unsegmented high-quality audio are derived from the recognized parts, as depicted in Figure 3.

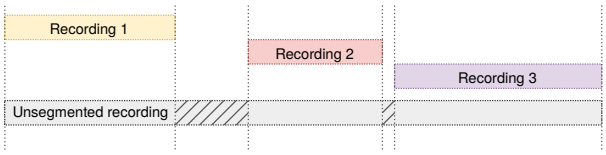

Fig. 3: Three works are found in a long, unsegmented recording. The meta-data of the works can be copied or compared. Segmentation timestamps can be derived from the recognized parts

For most works meta-dating and segmentation is straightforward after a link is found between the low- and high-quality audio archives. There are, however,

issues which complicate matters in some cases. The first is the unclear definition of work. Many tapes have been recorded during live concerts. These recordings contain spoken introductions and have an applause at the end. An intro and applause give meaningful context and might be relevant but are not consequently included. A work consisting of many parts might be divided up into its constituent parts or segmented as a single entity. See for example IPEM tape number 1076 in the supplementary material[2].

Another problem is that identical *audio is reused within the archive in different contexts*. Some tapes contain educational material: interviews, talks, lectures, radio shows often contain parts of works but they need not to be segmented as such (see IPEM tape number 1100). Some works also have multiple versions: they share much audio material but have slightly different meta-data. Composers also sampled material created for earlier works in new works: this reuse of material is revealing links between works. However the meta-data and segmentation information should not be simply copied. For example the works *'Difonium (cadenza, mix)'* and *'Difonium (C, mix)'* by Lucien Goethals share most audio. Many more can be found in the supplementary material[2].

A third problem is specific to the IPEM archive: different selection criteria have been used for the two digitization campaigns. The material in both collections overlap for a very large part but both also contain material not present in the other collection. So it is unclear when a link should be found but is lacking due to a flaw in the duplicate detection system. However, the synthetic evaluation below shows a very high reliability of the system used.

To evaluate the automatic segmentation approach we need two sets of recordings. Both sets need meta-data and segmentation info to check whether the matches are correct. Counter-intuitively, we need to manually segment a part of the unsegmented collection simply to be able to evaluate the segmentation timestamps found by duplicate detection. This manual segmentation was done using a custom build web interface as seen in Figure 4: the interface allows users to quickly modify segmentation boundaries.

For the 970 tapes in the archive, 2,790 segments were annotated. 1,158 segmented works were matched with the low-quality archive originating from CD's. Of these 1,158, the start and stop location is correct (within ten seconds) for 348 segments or 30%. The score is mainly due to the consistently higher granularity of segmentation (applause, introduction, parts) for the low quality audio compared with the annotated segments (works) of the high quality audio. With the audio correctly linked, a human expert is needed to evaluate whether the higher granularity matches are of interest and may be copied over to generate a definitive segmentation and meta-data set.

3.4 Key observations: meta-data reuse

The IPEM archive case effectively attaches meta-data and segmentation data from low-quality audio to high-quality audio. For most works this is a straightforward matter of copying meta-data to the matched high-quality audio. There

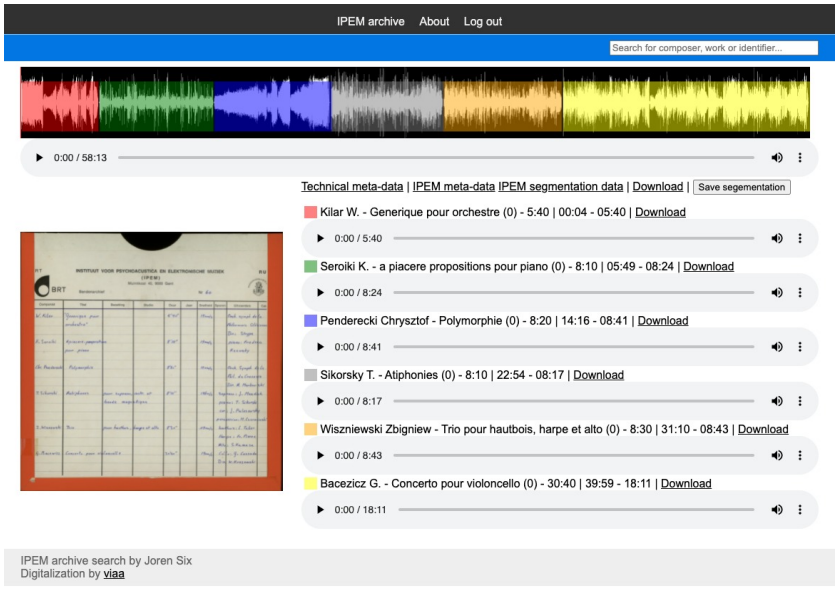

Fig. 4: A web interface to segment tapes manually. The segmentation boundaries (colored sections above) can be modified on the waveform.

are, however, caveats which make some cases more difficult: reuse of audio (sampling other works, several similar versions of works) and semantics around segmentation behaviour (including or excluding applause?). It is good practice to keep an expert listener in the loop to verify, confirm or modify meta-data.

4 Duplicate detection deep dive

In order to better grasp the *strengths and limitations* of the technology used in the case studies, the underlying algorithm is explained in this section. An efficient duplicate detection system is able to sort through millions of seconds of audio and come up with a relevant match, almost instantaneously, for a query containing only a handful of seconds of audio. This efficiency and level of accuracy made the previously discussed case studies possible.

The umbrella-term for duplicate detection in large archives is known as *acoustic fingerprinting*. The general idea is depicted in Figure 5. Some feature is extracted from audio and combined into a fingerprint. These fingerprints are then matched other fingerprints stored in a reference database. If a match is found, it is reported.

An audio feature with attractive properties for acoustic fingerprinting purposes are *local peaks in a spectral representation*. They have been used in many systems [29,26,23]. Alternative features are, for example, energy change in spec-

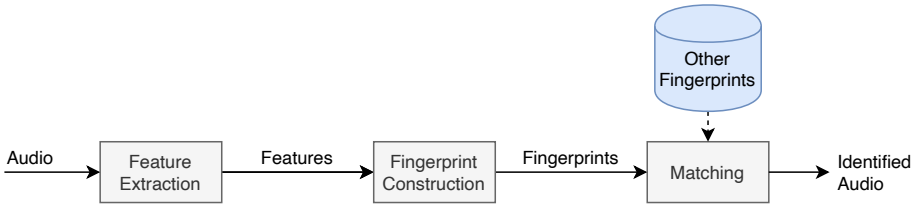

Fig. 5: General acoustic fingerprinting system.

tral bands [7,11]. However, there are not many systems which are both freely available and easy to use on larger datasets. One of the few open source systems is Panako[23], the system used here.

4.1 Panako - An acoustic fingerprinting system

Panako[3] [23] is an acoustic fingerprinting system. It is available under an AGPL license. Panako is based on TarsosDSP [22], a popular Java DSP library. Panako implements a baseline algorithm [29] and the Panako algorithm [23]. The Panako algorithm is able to match queries which are time-stretched, pitch-shifted or sped up with respect to the indexed audio. This is required for monitoring DJ mixes [25,13,18] and to match music from analogue media digitized at different speeds. The original Panako paper [23] describes the system in detail.

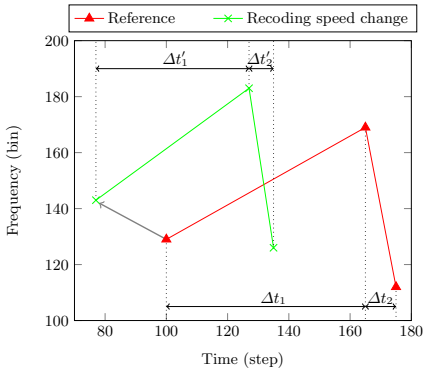

Fig. 6: The effect of speed modification on a fingerprint. The figure shows a single fingerprint extracted from reference audio (—▲—) and the same fingerprint extracted from audio after recording speed modification (—×—).

[3] `http://panako.be`: the Panako website (last visited January 19, 2022)

The key insight used in Panako is that a information of three local peaks in a spectral representation can be combined in a single hash robust against time/pitch modifications. In Figure 6, each peak has a time t, frequency f and magnitude m component and, for example, $\Delta t_1/\Delta t_2$ of the reference fingerprint equals $\Delta t'_1/\Delta t'_2$ after speed-up. Combining only such relative information in hashes allows to match the reference audio with modified audio from the same recorded event even after speed-up, time-stretching or pitch-shifting. More details can be found in the original Panako paper [23].

Panako received updates in 2021 after close inspection of an efficient implementation of a similar algorithm [20] for embedded systems. The underlying concepts from the original paper still stand but two changes improve the system considerably[4].

The first change replaces the frequency transform from a classical constant-Q transform [2,1] to a *constant-Q non-stationary Gabor transform* [12]. The latter has is more efficient and allows a finer frequency resolution at equal computational cost. See [28] for a detailed comparison. The spectral peak detection in Panako improves by the use of this finer frequency resolution. In Panako, JGaborator[5] is used: a wrapper around the Gaborator[6] library which implements a constant-Q non-stationary Gabor transform in C++11.

The second change replaces an *exact hash matching* technique by a *near-exact hash matching* approach. Some background helps understand this change. As mentioned before, the first step in the algorithm is a transform a one dimensional time series into a two dimensional time/frequency grid. Each bin in this grid has a very short duration and a small frequency dimension. The exact dimensions of these bins are determined by the spectral transform parameters and can be small but they remain discrete. This means that when a query and a match differ by about half the duration of a bin, energy is spread over neighbouring bins. Since time-frequency coordinates of peak magnitude bins are used in fingerprints, off-by-one errors are to be expected, both in time and frequency. Off-by-one errors are handled by the hash in the 2021 version of Panako.

For indexing and matching a hash is constructed from the components mentioned below. A hash combines fingerprint information into a single integer. The additional information contains an audio identifier used to tally matches. Below, the components are ordered from most to least significant. By only including approximate frequency information and having the time ratio in the least significant bytes, range queries become possible. The last couple of bits can be ignored during a search in ordered hashes: effectively dealing with off-by-one errors.

$$|f_3 - f_2|/4 \ ; \ |f_2 - f_1|/4 \ ; \ \tilde{f}_1$$
$$|t_3 - t_2| > |t_2 - t_1|$$
$$m_3 > m_2 \ ; \ m_3 > m_1 \ ; \ m_1 > m_2$$

[4] Note that this text describes and evaluates Panako as is in the following commit found on GitHub: 6cf936730131d71c94c562a06a1a791e09b4c520.

[5] `https://github.com/JorenSix/JGaborator` The JGaborator Github repository

[6] `http://gaborator.com/` The Gaborator website.

$$f_3 > f_1 \ ; \ f_3 > f_2 \ ; \ f_1 > f_2$$
$$(t_2 - t_1)/(t_3 - t_1)$$

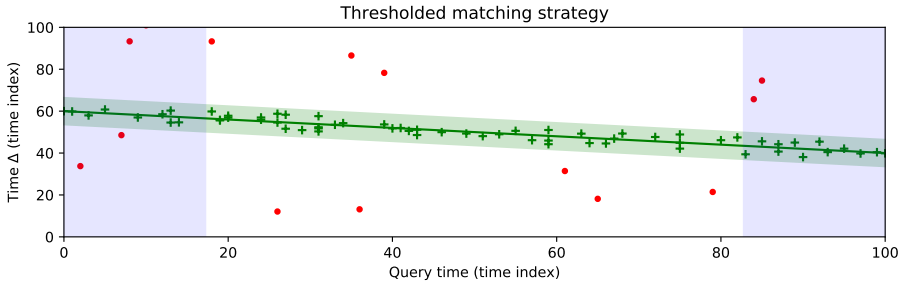

Fig. 7: To discriminate true from false positives a thresholded matching strategy is used. The first few and last few matches (blue background) are used to calculate a median Δt. Accepted matches (green background) fall in a small range around a linear regression from the first to last median Δt. Some random matches (red dots) are dismissed.

This idea of gracefully handling off-by-one errors needs to be reflected in the matching scheme as well. The fingerprints extracted from a query are matched with with the index. A list of matching prints is returned and needs to be filtered: hashes might randomly collide or short fragments might match for a very short duration. To filter true positive matches from false positives the difference in time (Δt) between each reference and query hash. For a true match Δt is either a constant or changes linearly over time. In the original paper[29] a true positive is only accepted if Δt is a fixed constant. Here, we calculate a linear regression from the first matches (blue in Figure 7) to the last and allow some small margin in which matches are accepted. In this manner off-by-one matches and time-stretching/speed-up are supported.

When larger archives are indexed, the characteristics of the key-value store become more and more important. The key-value store stores a hashes together with some additional information. A hash combines fingerprint information into a single number. The additional information contains an audio identifier used to tally matches. The 2021 Panako verion stores ordered fingerprints using a persistent, compact, high performance, B-Tree[7] [4]. The speed, small storage overhead and performance allow more beneficial trade-offs between query performance: it facilitates storing more fingerprints per second of audio and larger data-sets for equal or better query performance.

[7] LMDB: Lightning Memory-Mapped Database Manager, `http://lmdb.tech` (last visited January 19, 2022)

4.2 Panako Evaluation

To show the strengths and weaknesses of the Panako system, an evaluation is done on the free music archive [6] medium dataset[8]. In total 25,000 music fragments of 30 seconds were used in the evaluation. One fifth of the data set was not indexed to check for true negatives. Readers are encouraged to repeat the evaluation as it is completely reproducible with the evaluation script provided as part of the Panako software distribution. Details on the exact parameters for the modifications can be found there as well.

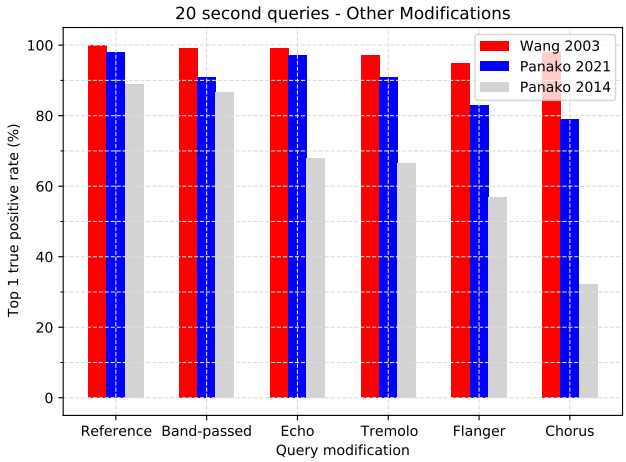

Fig. 8: Comparison of the top one true positive rate after several modifications for 20 second query fragments for Wang 2003 [29], Panako 2014 [23] and Panako 2021.

Panako is evaluated for various modifications (Figure 8), playback speed modification (Figure 11), time-stretching 10 and pitch-shifting 9. The evaluation follows a straightforward method: a random fragment is selected, a modification is applied and the modified fragment is used to queried the index for matches. Only the best match is considered and counted as a true positive, false positive, true negative or false negative. The sensitivity or true positive rate is reported ($TP/(TP+FN)$).

For all modifications in Figure 8 the query performance from the 2014 to 2021 version of Panako is clear. The chorus effect impacts the spectrogram the most: the performance increases from about 25% to 80%. The baseline algorithm (which is also implemented in Panako) is, however, always better but does not

[8] The data can be downloaded at `https://github.com/mdeff/fma` (last visited January 19, 2022)

support pitch-shift or time-stretch. It shows that there is still headroom for further refinement.

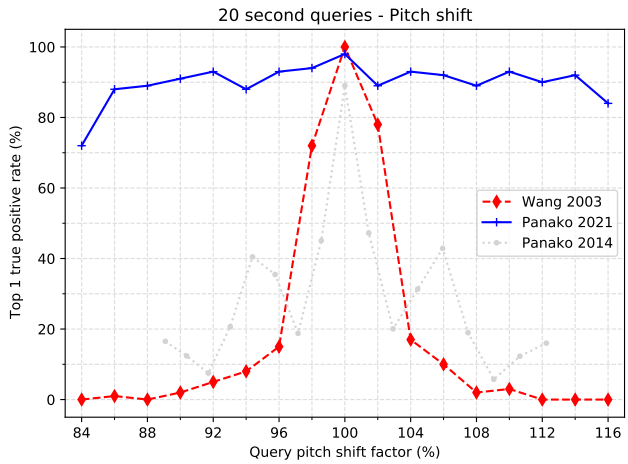

Fig. 9: Comparison of the top one true positive rate after pitch-shifting for 20s query fragments for Wang 2003 [29], Panako 2014 [23] and Panako 2021.

The pitch-shift and time-stretch modifications (Figures 10 and 9) are calculated from -16% to +16% to reflect a common maximum modification on DJ equipment. Again it is clear that the upgrades drastically improve the performance especially in higher modification factors. The baseline algorithm [29] normally does not support time-stretch modification. The Panako implementation of [29] uses the matching strategy described above which allows time-stretch: Δt is not a fixed constant but is allowed to change linearly.

The speed-up modification (Figures 11 can be seen as a combination of both time-stretching and pitch-shifting with the same factors. Query performance is, in other words, limited by the time-stretch and pitch-shift performance. For the larger modification factors the performance drops below 80% but is still much above Panako 2014[9]. More extreme playback speed modification is supported by [26] but their reported query speed is much slower and the system is not freely available.

The query speed of Panako varies with the size of the database and properties of the audio: acoustically dense audio generates more fingerprints. On a Early 2015 Macbook Pro with a 2,9 GHz Dual-Core Intel Core i5 and storage and query is 38 times faster than real-time per processor core. This means that 38 seconds of audio is handled in one second for each processor core.

[9] For Panako, time-stretch and pitch-shift factors do not need to be the same: a fragment pitch-shifted 104% followed by a 92% time-stretch will match the original.

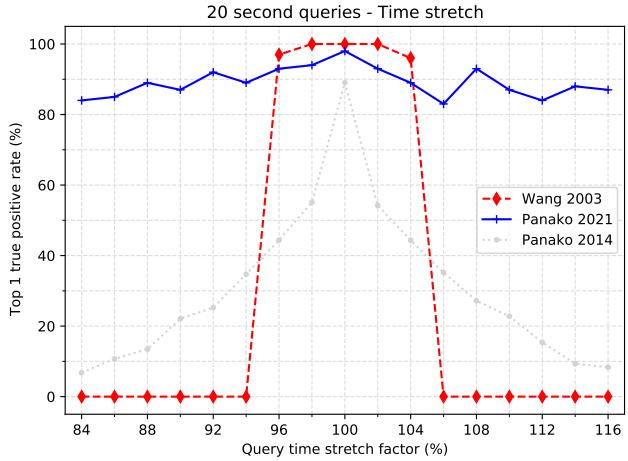

Fig. 10: Comparison of the top one true positive rate after time-stretching for 20s query fragments for Wang 2003 [29], Panako 2014 [23] and Panako 2021.

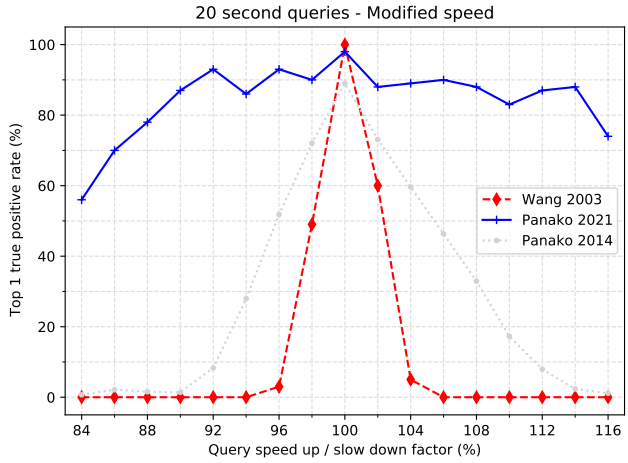

Fig. 11: Comparison of the top one true positive rate after speed up/slow down for 20s query fragments for Wang 2003 [29], Panako 2014 [23] and Panako 2021. The audio playback speed is modified from 84 to 116% with respect to the indexed reference audio. If the query is slowed down by 10%, the duration ends up being 22s. For the 2021 Panako algorithm, audio recognition performance suffers (below 80%) when playback speed is changed more than 10%.

5 Conclusion

In this chapter a mature MIR technology of duplicate detection was described and put to the test in two case studies. In the first case study, duplicates were detected for a part of the music archive of a public broadcaster. This allowed to identify the unique material in the archive and had additional benefits to check meta-data quality and redirect listeners to higher quality duplicates. In the second case study meta-data and segmentation information of low-quality audio was attached to higher-quality duplicates.

The main takeaway from both case studies is that duplicates can be found reliably and easily even in larger archives. For the most part duplicates are straightforward but some surprising and interesting cases of audio-reuse (sampling, translations, versions) might warrant the need for a human expert in the loop.

The chapter concluded with a technical description and evaluation of Panako - an acoustic fingerprinting system. This was done in order to better understand strengths and weaknesses of the technology. The evaluation of the 2021 version of Panako shows much improved performance over previous versions.

With the case studies, we aimed to directly improve the quality of two collections by showing how duplicate detection technology helps to offer better services to end users. Evidently, we hope to accelerate the adoption of duplicate detection technology by the community of audio archivists and indirectly improve many digital audio archives.

References

1. Judith C Brown. Calculation of a constant q spectral transform. *The Journal of the Acoustical Society of America*, 89(1):425–434, 1991.
2. Judith C Brown and Miller S Puckette. An efficient algorithm for the calculation of a constant q transform. *The Journal of the Acoustical Society of America*, 92(5):2698–2701, 1992.
3. Pedro Cano, Eloi Batlle, Ton Kalker, and Jaap Haitsma. A review of audio fingerprinting. *The Journal of VLSI Signal Processing*, 41:271–284, 2005.
4. Douglas Comer. Ubiquitous b-tree. *ACM Computing Surveys (CSUR)*, 11(2):121–137, 1979.
5. Reinier de Valk, Anja Volk, Andre Holzapfel, Aggelos Pikrakis, Nadine Kroher, and Joren Six. Mirchiving: Challenges and opportunities of connecting mir research and digital music archives. In *Proceedings of the 4th International Workshop on Digital Libraries for Musicology*, DLfM '17, pages 25–28, New York, NY, USA, 2017. ACM.
6. Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. FMA: A dataset for music analysis. In *18th International Society for Music Information Retrieval Conference (ISMIR)*, 2017.
7. Dan Ellis, Brian Whitman, and Alastair Porter. Echoprint - an open music identification service. In *Proceedings of the 12th International Symposium on Music Information Retrieval (ISMIR 2011)*, 2011.
8. Sébastien Fenet, Gaël Richard, and Yves Grenier. A Scalable Audio Fingerprint Method with Robustness to Pitch-Shifting. In *Proceedings of the 12th International Symposium on Music Information Retrieval (ISMIR 2011)*, pages 121–126, 2011.

9. Beat Gfeller, Dominik Roblek, Marco Tagliasacchi, and Pen Li. Learning to denoise historical music. In *ISMIR 2020 - 21st International Society for Music Information Retrieval Conference*, 2020.
10. Beat Gfeller, Dominik Roblek, Marco Tagliasacchi, and Pen Li. Learning to denoise historical music. In *ISMIR 2020 - 21st International Society for Music Information Retrieval Conference*, 2020.
11. Jaap Haitsma and Ton Kalker. A highly robust audio fingerprinting system. In *Proceedings of the 3th International Symposium on Music Information Retrieval (ISMIR 2002)*, 2002.
12. Holighaus, Nicki and Dörfler, Monika and Velasco, Gino Angelo and Grill, Thomas. A framework for invertible, real-time constant-q transforms. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(4):775–785, 2012.
13. Taejun Kim, Minsuk Choi, Evan Sacks, Yi-Hsuan Yang, and Juhan Nam. A computational analysis of real-world dj mixes using mix-to-track subsequence alignment. In *21st International Society for Music Information Retrieval Conference (ISMIR), 2020*, 2020.
14. Leman, Marc and Dierickx, Jelle and Martens, Gaëtan. The IPEM-archive conservation and digitalization project. *JOURNAL OF NEW MUSIC RESEARCH*, 30(4):389–393, 2001.
15. Lesaffre, Micheline. 50 years of the Institute for psychoacoustics and electronic music. In Jacobs, Greg, editor, *IPEM Institute for Psychoacoustics and Electronic Music : 50 years of electronic and electroacoustic music at the Ghent University*, volume 004 of *Metaphon*, pages 23–25. Metaphon, 2013.
16. Matija Marolt, Ciril Bohak, Alenka Kavčič, and Matevž Pesek. Automatic segmentation of ethnomusicological field recordings. *Applied Sciences*, 9(3):439, 2019.
17. Nicola Orio. Searching and classifying affinities in a web music collection. In *Italian Research Conference on Digital Libraries*, pages 59–70. Springer, 2016.
18. Diemo Schwarz and Dominique Fourer. Unmixdb: Een dataset voor het ophalen van dj-mixinformatie. In *19th International Symposium on Music Information Retrieval (ISMIR)*, 2018.
19. Joan Serra, Emilia Gómez, and Perfecto Herrera. Audio cover song identification and similarity: background, approaches, evaluation, and beyond. In *Advances in music information retrieval*, pages 307–332. Springer, 2010.
20. Joren Six. Olaf: Overly lightweight acoustic fingerprinting. 2020.
21. Joren Six, Federica Bressan, and Marc Leman. Applications of duplicate detection in music archives: from metadata comparison to storage optimisation. In *Italian Research Conference on Digital Libraries*, pages 101–113. Springer, 2018.
22. Joren Six, Olmo Cornelis, and Marc Leman. TarsosDSP, a real-time audio processing framework in Java. In *Proceedings of the 53rd AES Conference (AES 53rd)*. The Audio Engineering Society, 2014.
23. Joren Six and Marc Leman. Panako - A scalable acoustic fingerprinting system handling time-scale and pitch modification. In *Proceedings of the 15th ISMIR Conference (ISMIR 2014)*, pages 1–6, 2014.
24. Joren Six and Marc Leman. Synchronizing multimodal recordings using audio-to-audio alignment. *Journal on Multimodal User Interfaces*, 9(3):223–229, Sep 2015.
25. Reinhard Sonnleitner, Andreas Arzt, and Gerhard Widmer. Landmark-based audio fingerprinting for dj mix monitoring. In *ISMIR*, pages 185–191, 2016.
26. Reinhard Sonnleitner and Gerhard Widmer. Quad-based Audio Fingerprinting Robust To Time And Frequency Scaling. In *Proceedings of the 17th International Conference on Digital Audio Effects (DAFx-14)*, 2014.

27. Christoph F. Stallmann and Andries P. Engelbrecht. Gramophone noise reconstruction - a comparative study of interpolation algorithms for noise reduction. In *2015 12th International Joint Conference on e-Business and Telecommunications (ICETE)*, volume 05, pages 31–38, 2015.
28. Velasco, Gino Angelo and Holighaus, Nicki and Dörfler, Monika and Grill, Thomas. Constructing an invertible constant-q transform with non-stationary gabor frames. *Proceedings of DAFX11, Paris*, 33, 2011.
29. Avery Li-Chun Wang. An industrial-strength audio search algorithm. In *Proceedings of the 4th International Symposium on Music Information Retrieval (ISMIR 2003)*, pages 7–13, 2003.