DISCSTITCH: TOWARDS AUDIO-TO-AUDIO ALIGNMENT WITH ROBUSTNESS TO PLAYBACK SPEED VARIABILITIES

Joren Six
joren.six@ugent.be
IPEM, Ghent University, Belgium

Late-Breaking / Demo Session Extended Abstract, ISMIR 2022 Conference

ABSTRACT

Before magnetic tape recording was common, acetate discs were the main audio storage medium for radio broadcasters. Acetate discs only had a capacity to record about ten minutes. Longer material was recorded on overlapping discs using (at least) two recorders. Unfortunately, the recorders used were not reliable in terms of recording speed, resulting in audio of variable speed.

To make digitized audio originating from acetate discs fit for reuse, (1) overlapping parts need to be identified, (2) a precise alignment needs to be found and (3) a mixing point suggested. All three steps are challenging due to the audio speed variabilities.

This paper introduces the ideas behind DiscStitch: which aims to reassemble audio from overlapping parts, even if variable speed is present. The main contribution is a fast and precise audio alignment strategy based on spectral peaks. The method is evaluated on a synthetic data set.

1. INTRODUCTION

The archives of the national radio broadcaster of Belgium contains many **acetate discs**, most were recorded between 1930 and 1960. The main advantage was that the discs were relatively cheap and allowed long term storage. A disadvantage was that the sound carriers were quite brittle: the needle used for playback damages the record much more than the contemporary shellac or vinyl discs.

Another disadvantage of these sound carriers was that they had a relatively **short recording duration**: about 5 to 10 minutes, depending on the RPM and disc diameter. To record longer events, two machines were used and alternated. When a first disc was almost full, a second recorder was started to record on a subsequent disc. To continue recording, the machines were alternated eventually resulting in an ‘album’ of discs, most with some overlap to the next disc. Note that most - but not all - subsequent discs overlap: natural pauses sometimes did coincide with the end of a disc. E.g. when a full song fits on disc, a new song was started on the next disc, without any overlap.

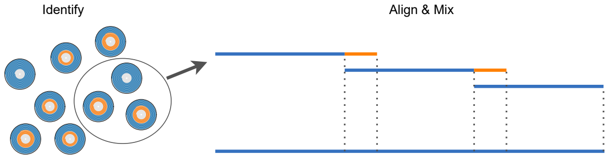

Figure 1. Overlapping discs need to be identified from a large set of disks. The overlapping parts (in orange) need to be precisely aligned. A mixing step reassembles the original recorded event. Identification, alignment and mixing is problematic due to speed differences and speed variabilities in overlapping discs.

The digitized archive of such discs contains digital audio files with overlapping audio. The archive of the Belgian national radio contains over 10 000 digitized acetate disc recordings. To make the material fit for reuse three steps are needed. First, the albums need to be recombined based on meta-data or the overlapping audio from the separate files. This is the **identification** step. Second, the overlapping audio needs to be **aligned** precisely. The final and third step is **mixing**. This uses the alignment and determines mixing points to finally arrive at a recombined long sound file fit for reuse. A visual representation of this process can be found in Figure 1.

Since the identification step is a solved problem with modern acoustic fingerprinting systems [1–3]. We focus on the alignment step.

The main problem with these discs is that the recording speed differed slightly between machines. Moreover, there are the instabilities in recording speeds which were caused by mechanical imperfections. For the digital files this means that the **speed of overlapping audio can differ** by a few percent and that this difference is not constant. This interferes with straightforward solutions for alignment and mixing.

In this paper we present a solution for alignment and mixing of audio with variable speed. The main contribution lies in a new algorithm to align audio with speed differences precisely and quickly. An additional contribution is a generally applicable method to gauge relative recording and/or playback speed differences in audio.

2. AUDIO-TO-AUDIO ALIGNMENT

The most straightforward way to do audio-to-audio alignment is sample-wise cross-correlation. Cross-correlation

© J. Six. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** J. Six, “DiscStitch: towards audio-to-audio alignment with robustness to playback speed variabilities”, in *Extended Abstracts for the Late-Breaking Demo Session of the 23rd Int. Society for Music Information Retrieval Conf.*, Bengaluru, India, 2022.

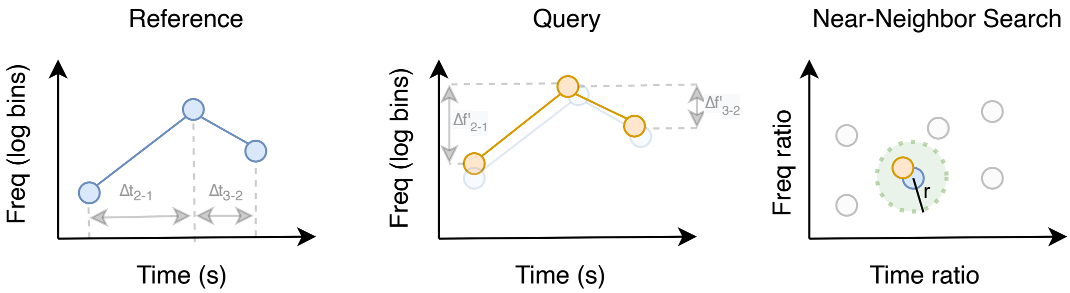

Figure 2. Spectral peaks are extracted from reference audio and combined in lists of three peaks (blue). A slightly sped up query (orange) has a slightly higher frequency and a slightly faster succession of peaks. The ratios between the time $\Delta t_{2-1}/\Delta t_{2-1} = \Delta t'_{2-1}/\Delta t'_{2-1}$ and frequency deltas $\Delta f_{2-1}/\Delta f_{2-1} = \Delta f'_{2-1}/\Delta f'_{2-1}$ stay nearly identical for query and reference. If each list of three peaks is mapped to a 2D plane a near-neighbour search with a small circular radius r yields spectral peak lists originating from duplicate audio.

quickly becomes computationally unfeasible with unknown speed differences. Typically dynamic time warping (DTW) based algorithms [4–8] are used to align audio with speed differences. However, most DTW algorithms have the assumption that the start and end points of audio to align is known beforehand. The DiscStitch algorithm presented here does not have this assumption and offers different trade-offs.

The DiscStitch audio-to-audio alignment algorithm works by extracting peaks in a spectral representation with a max filter. Each peak has a time and frequency component: (t, f). These lists of peaks reduces the information drastically. Now the audio alignment problem is reduced to aligning these lists of peaks.

Alignment of peaks is difficult due to variable speed differences. To cope with this we bundle neighbouring peaks in sets of three and calculate $t_{ratio} = (t_2 - t_1)/(t_3 - t_2)$ and $f_{ratio} = (f_2 - f_1)/(f_3 - f_2)$, an idea similar to [1,9]. As visualized in fig , these ratios stay equal even if the audio is sped-up with respect to the reference.

Next the ratio's are mapped to a 2D pane the with the time-ratio in the horizontal and frequency-ratio in the vertical axis. Points close to each other in this pane have nearly the same ratios and might originate from spectral triplets with the same shape which may mean that they originate from similar audio. For an efficient near-neighbor search, the ratios from the reference are added to a R*-Tree [10]. Then, the ratios from the query are used to search for near neighbors in a small circular radius. Resulting in a list of matches.

The matches are then filtered to weed out random hits. For an actual audio alignment there is a near linear relation between reference time component and query time component. Additionally frequency components for reference and query should be relatively close to each other. The filtered matches give a list of times where audio from the reference aligns with the query.

3. PRELIMINARY EVALUATION & DISCUSSION

The DiscStitch audio-to-audio alignment has been implemented twice. One is a browser-based JavaScript/WASM version [1] , the other is more capable Java based version [2] which is evaluated below.

For evaluation purposes a long reference file is chopped up into two parts with 20 to 40 second overlap. Next, these two files are aligned and mixed. If the alignment is correct and mixing works as expected the duration of the long reference file should equal the length of the mixed parts. The preliminary evaluation shows that the duration of the mixed file differs only slightly from the original ($\mu = 0.1ms, \sigma = 1.2ms, N = 80$) indicating a precise alignment and mix. Note that sound travels 1m in about 3ms. The evaluated implementation uses step sizes of 1.4ms (32 samples at 22050Hz) which limits alignment precision and explains the standard deviation.

A second test changes the speed of the second part ($\pm 3\%$) . The goal is then to estimate and undo the speed change applied to the second part and align an mix it with the first part. In that case the remixed duration matches ($\mu = 1.3ms, \sigma = 237.2ms, N = 80$). A small error in estimating the speed change can result in a large error in overall duration, explaining the higher standard deviation.

A third test was done with Sonic Lineup: a piece of software of the Sonic Visualiser family *'designed for rapid visualisation of multiple audio files containing versions of the same source material'* [3] . It allows to call external alignment programs. With DiscStitch doing the alignment. A listening test with digitized acetate disks showed precise alignment.

DiscStitch offers a system to potentially speed up the restoration of lacquer disc albums by identifying potential alignment points. However, an expert archivist needs to be in the loop to determine which alignment point to choose, which speed modifications to apply and to potentially apply other modifications.

[1] https://github.com/JorenSix/SyncSink.wasm
[2] https://github.com/JorenSix/DiscStitch
[3] https://www.sonicvisualiser.org/sonic-lineup/

4. REFERENCES

[1] J. Six and M. Leman, “Panako: a scalable acoustic fingerprinting system handling time-scale and pitch modification,” in *Proc. of the 15th Int. Society for Music Information Retrieval Conf.*, 2014.

[2] R. Sonnleitner and G. Widmer, “Robust quad-based audio fingerprinting,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 409–421, 2015.

[3] J. Six, “Panako: a scalable audio search system,” *Journal of Open Source Software*, vol. 7, no. 78, p. 4554, 2022. [Online]. Available: https://doi.org/10.21105/joss.04554

[4] S. Dixon and G. Widmer, “Match: A music alignment tool chest.” in *Proc. of the 6th Int. Society for Music Information Retrieval Conf.*, 2005.

[5] M. Müller, *Dynamic Time Warping*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 69–84. [Online]. Available: https://doi.org/10.1007/978-3-540-74048-3_4

[6] M. Müller, Y. Özer, M. Krause, T. Prätzlich, and J. Driedger, “Sync toolbox: A python package for efficient, robust, and accurate music synchronization,” *Journal of Open Source Software*, vol. 6, no. 64, p. 3434, 2021.

[7] S. Ewert, M. Muller, and P. Grosche, “High resolution audio synchronization using chroma onset features,” in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 1869–1872.

[8] C. Tralie and E. Dempsey, “Parallelizable dynamic time warping alignment with linear memory,” in *Proc. of the Int. Soc. for Music Information Retrieval Conf. (ISMIR), in print*, 2020.

[9] A. L.-c. Wang and D. Culbert, “Robust and invariant audio pattern matching,” US Patent US7 627 477 B, 2003.

[10] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, “The r*-tree: An efficient and robust access method for points and rectangles,” in *Proceedings of the 1990 ACM SIGMOD international conference on Management of data*, 1990, pp. 322–331.