OLAF: Overly Lightweight Acoustic Fingerprinting

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

Abstract

Olaf is a portable, landmark-based, acoustic fingerprinting system released as open source software. Olaf runs on embedded platforms, traditional computers and in the browser. Olaf is able to extract fingerprints from an audio stream, and either store those fingerprints in a database, or find a match between extracted fingerprints and stored fingerprints. It implements an algorithm similar to the one described in a classic ISMIR paper [7] and has similar retrieval performance. It facilitates the many use cases acoustic fingerprinting has to offer such as duplicate detection, meta-data coupling, and synchronization [1,4,5]. Olaf stands out for three reasons.

Olaf runs on embedded systems. On embedded platforms memory and computational resources are severely limited. Olaf is written in ANSI C with these restrictions in mind. Olaf targets 32-bit ARM microcontrollers with at least 256kB memory such as the Arduino Nano 33 BLE, the ESP32 or the Teensy 4.0. As far as I know this is unique for an acoustic fingerprinting system and allows innovative IoT music recognition and synchronization applications. The original motivation behind Olaf was to give my daughter a modified "Elsa-dress" for her birthday. The modification added a LED-strip which lights up when, and only when, '*Let It Go*' from the Frozen soundtrack is playing.

Figure 1. An embedded system running Olaf. Music in the environment is recognized and LEDs are lit in sync.

Thanks to its low computational and memory footprint **Olaf is fast on traditional computers**. On such devices, fingerprints are stored in a high-performance key-value-store: LMDB. LMDB offers a B+-tree [2] based persistent storage which is ideal for small keys and values with low storage overhead. On modest computing hardware [1] extracting and storing fingerprints takes around 1429 ±205 times real-time. One hour of audio takes about 2.5 seconds to analyze and store. Query performance is slightly slower and depends on the number of fingerprints in the database. With 10k 4min songs in the reference database queries are handled at around 891 ±99 times real time. The scale of performance increases relative to other fingerprinting systems[2] is such that, for example, duplicate detection in large music archives, becomes practical.

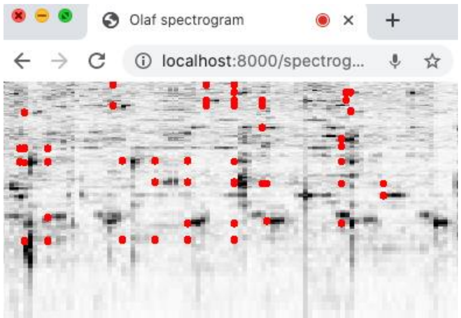

Figure 2. A visualization web app with Olaf extracting fingerprints (red dots) in a browser. The Web Audio API makes the microphone accessible, WASM the Olaf functionality.

Olaf works in the browser. Via Emscripten Olaf can be compiled to WASM, a type of machine language that browsers are able to run in a sandboxed environment. This makes it relatively straightforward to combine the capabilities of the Web Audio API and Olaf to create browser-based audio fingerprinting applications. This is especially powerful when combined with WebSockets. WebSockets allow a full-duplex connection to webservers and, in this case, allow interactive, browser-based acoustic fingerprinting applications.

Next to the previous three unique features the system may also be of interest to the ISMIR community as an acoustic fingerprinting system that can serve as a baseline to compare current acoustic fingerprinting systems to. Of course, Olaf itself offers a starting point to experiment with acoustic fingerprinting systems. Olaf is available on http://github.com/JorenSix/Olaf

[1] The test-system system contained a 2017, intel i5-7260U CPU @ 2.20GHz and an SSD.

[2] Olaf is more than 30 times faster than a system described in [6] and about 1000 times faster than [3].

1. REFERENCES

[1] P. Cano, E. Batlle, E. Gómez, L. de CT Gomes & M. Bonnet. "*Audio fingerprinting: concepts and applications*". In Computational intelligence for modelling and prediction (pp. 233-245). Springer, Berlin, Heidelberg. 2005.

[2] D. Comer, "*Ubiquitous B-tree*". ACM Computing Surveys (CSUR), 11(2), 1979 121-137.

[3] D. Ellis, "*The 2014 labrosa audio fingerprint system.*" ISMIR. 2014. MIREX abstract.

[4] J. Six, F. Bressan & M. Leman, *"Applications of duplicate detection in music archives: from metadata comparison to storage optimisation."* In Italian Research Conference on Digital Libraries (pp. 101-113). Springer, Cham. 2018

[5] J. Six and M. Leman, *"Synchronizing multimodal recordings using audio-to-audio alignment"*. Journal on Multimodal User Interfaces, 9(3), 2015, 223-229.

[6] 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.

[7] A. L. Wang, "*An Industrial-Strength Audio Search Algorithm*". In Proc. of the 4th Int. Symposium on Music Information Retrieval Conf. 2003, pp. 7–13