A framework to provide fine-grained time-dependent context for active listening experiences

Joren Six and Marc Leman - joren.six@ugent.be - IPEM, University Ghent - Belgium

Abstract

This work presents a system that is able to provide fine-grained time-dependent context while listening to recorded music. By utilizing acoustic fingerprinting techniques the system recognizes which music is playing in the environment and also determines an exact playback position. This makes it possible to provide context at exactly the right time.

The design of the system can be used to augment listening experiences with lyrics, scores, tablature or even music videos.

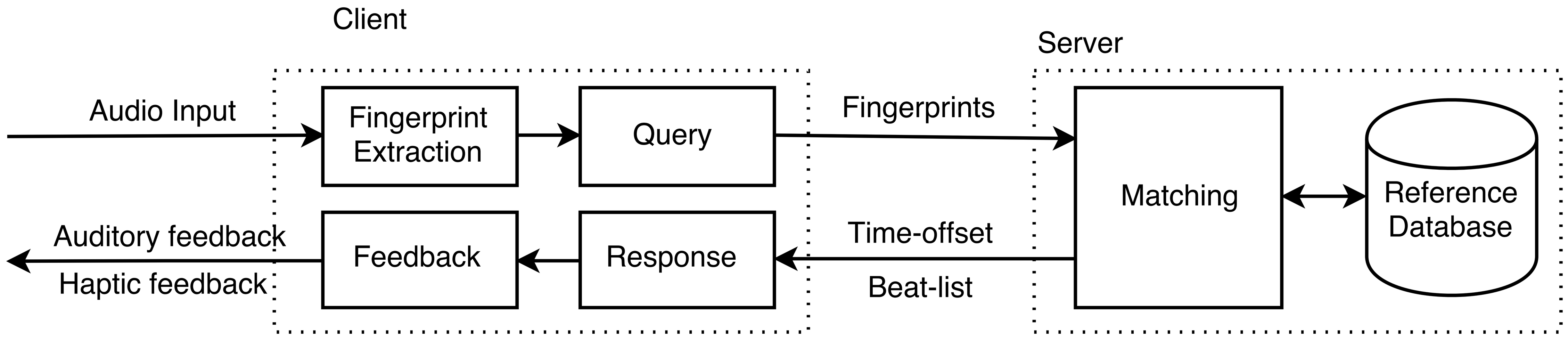

Fig 1: Schema of a client/server architecture to match an audio query and receive synchronized feedback on the beat.

Introduction

The ability to identify which music is playing in the environment of a user has several use cases. After a successful recognition meta-data about the music is immediately available: artist, title, album. Such systems have been in use for more than a decade now. Some even incorporate a real-time synchronized display of lyrics. However there are none that offer fine-grained synchronized feedback of meta-data.

In this work a design of a system is proposed that i) is able to find the playback time of the music in the environment precisely and ii) provide feedback synchronized to the environment.

The paper focuses providing feedback on the beat. A prototype is developed that provides feedback exactly on the beat.

Feedback on the beat could be a helpful tool for cochlear implant users that were implanted later in life. To measure how accurately the system is able to provide context to music (lyrics, music videos, ...) discrete beat events are ideal. Beat information on much music is readily available [4] for millions of tracks.

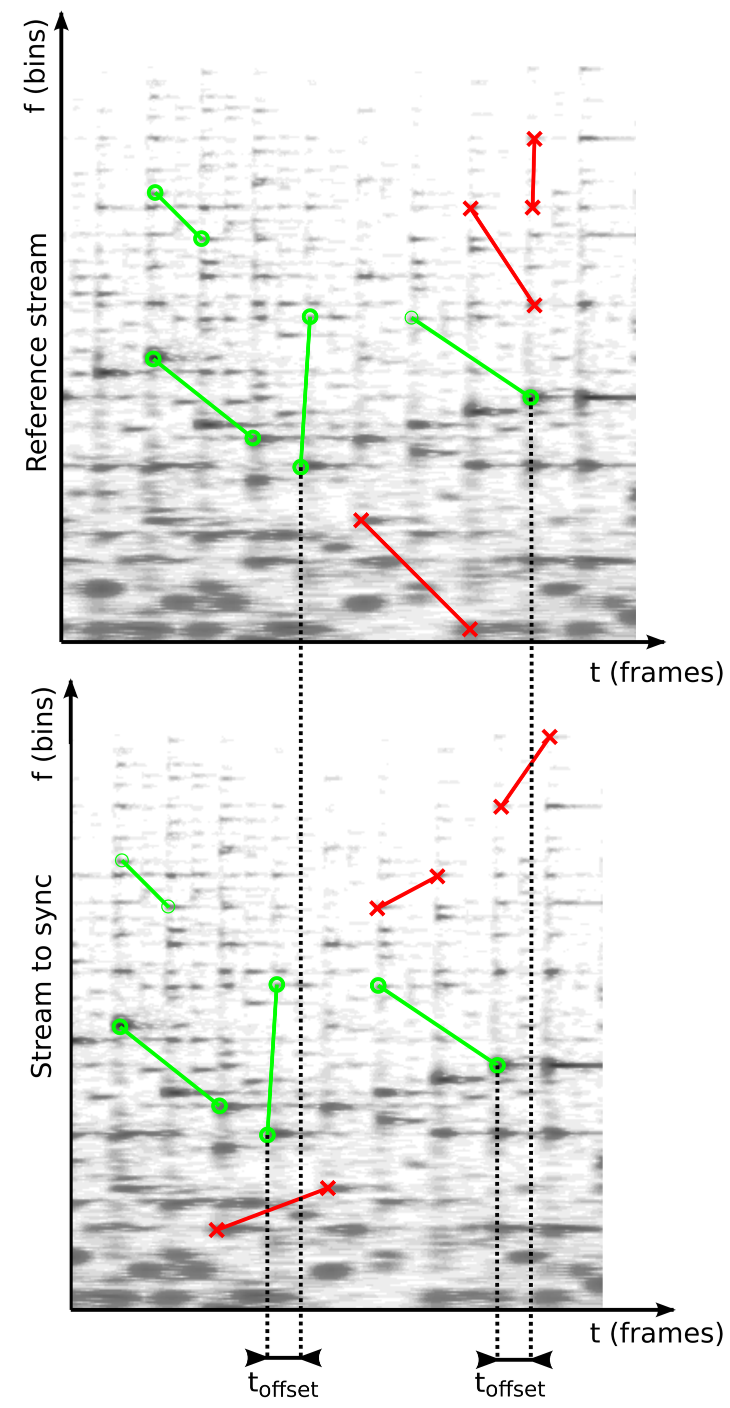

Fig 2: Pairs of spectral peaks are used to synchronize the query with the reference audio.

Evaluation & Results

An implementation was done with Panako[1] an open source acoustic fingerprinting framework and the underlying TarsosDSP [2] audio processing framework. Panako was exteded to allow extraction of meta-data and with a HTTP REST API to provide the client-server architecture depicted in Figure 1. The acoustic fingerprinting algorithm from [3] (Figure 2) was already available in the framework

The evaluation makes clear how synchronized context can be delivered to ambient audio or music. The evaluation quantifies the time-offset between the beats - annotated beforehand - and the time of the feedback event that should correspond with a beat. The evaluation uses a micro-controller with a real time operating system and low audio output latency (Fig 3) to ensure correct measurements.

The results are depicted in Figure 4. The system responds on average 16 ms before the beat. This allows feedback events to be perceived together with the beat. Depending on the tempo (BPM) of the music and type of feedback it might be needed to schedule events later or even sooner. This can be done by adapting the latency parameter which modifies the timing of the scheduled feedback events. However, there is a large standard deviation of 42 ms. The current system is limited to an accuracy of 32 ms: the size of a block of 256 audio samples, sampled at 8000 Hz. The block size used in the fingerprinter.

In Figure 4 each histogram bin is 32 ms wide and centered around -16 ms. The results show that the system is able to recognize the correct audio block but is sometimes one block off. The main issue here is the unpredictable nature of scheduling in Java. The results, however, do show that the concept of the system is very promising and can deliver timing dependent context.

The accuracy of the system could be improved by running a real-time onset detector and correlating the detected onsets list with the beat list returned by the server, this would however make the system more computationally expensive.

System overview

A schematic overview of the system can be found in figure 1.

A client uses its microphone to register sounds in the environment. Next, fingerprints are extracted from the audio stream [3]. The fingerprints are sent to a server. The server matches the fingerprints with a reference database. If a match is found, a detailed time-offset between the query and the reference audio is calculated.

Subsequently, the server returns this time-offset together with a list of beat timestamps. Using this information the client is able to generate feedback events that coincide with the beat of the music playing in the users environment. This process is repeated to make sure that the feedback events remain in sync with the music in the room. If the server fails to find a match, the feedback events stop.

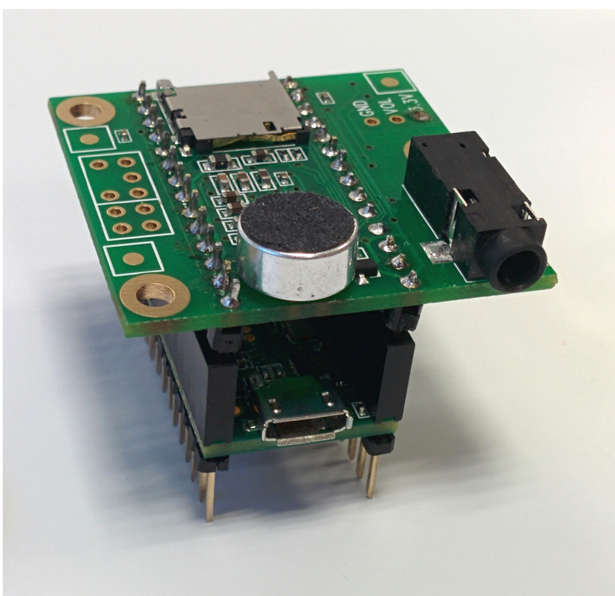

Fig 3: Teensy with Audio Board. The microcontroller is capable of high-quality low-latency audio playback from an SD-card and precise time measurement

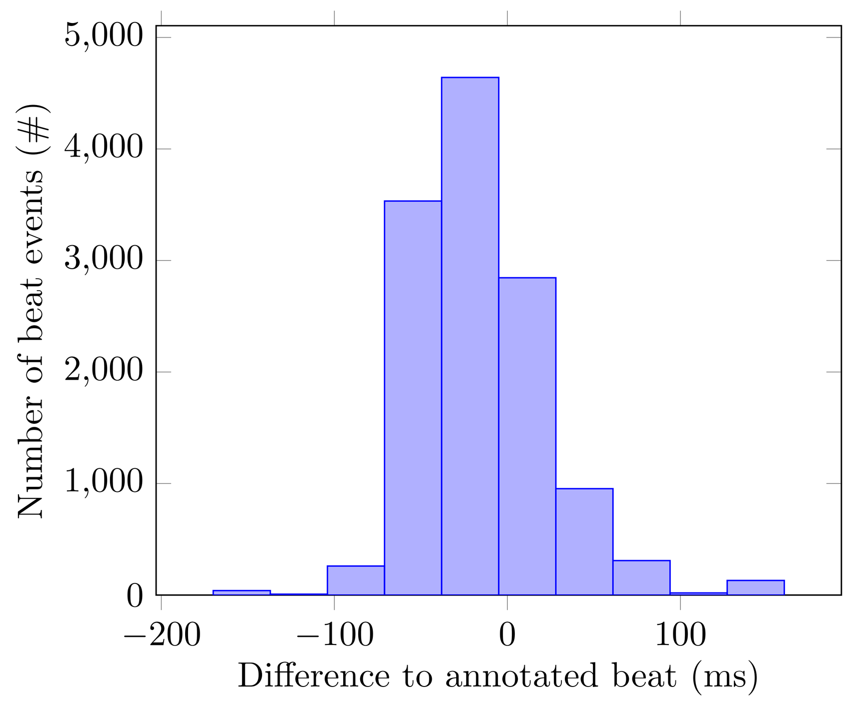

Fig 4: The difference (in ms) between feedback events and beats. The feedback events arrive 16ms before the beat to allow to perceive beats together with feedback. The data is centered around the average and bins of 32ms are used, the precision of the system..

Fig 5: Screenshot of the fingerprint visualizer, part of the Panako acoustic fingerprinting software.

Conclusion

A system was described that employs acoustic fingerprinting techniques to provide augmented music listening experience. A prototype was developed that provides feedback synchronized with music being played in the environment. The system needs a dataset with fingerprints from reference audio and pre-computed beat-lists.

Since it offers fine-grained context awareness the systems design can be used to also show lyrics, scores, visuals, aligned music videos or other meta-data that enrich the listening experience.

References

[1] Joren Six and Marc Leman, Panako - A Scalable Acoustic Fingerprinting System Handling Time-Scale and Pitch Modification in Proceedings f the 15th ISMIR Conference (ISMIR 2014)
[2] Joren Six, Olmo Cornelis, and Marc Leman. TarsosDSP, a Real-Time Audio Processing Framework in Java. In Proceedings of the 53rd AES Conference (AES53rd), 2014.
[3] Avery L. Wang. An Industrial-Strength Audio Search Algorithm. In Proceedings of the 4th International Symposium on Music Information Retrieval (ISMIR 2003), pages 7–13, 2003.
[4] Porter, A., Bogdanov, D., Kaye, R., Tsukanov, R., & Serra, X. Acousticbrainz: a community platform for gathering music information obtained from audio. ISMIR 2015.

Availability

Panako and TarsosDSP are open source software and available on under the restrictions of the GPL license on

```
http://0110.be
http://panako.be
https://github.com/JorenSix/Panako
```