Essentia in the browser

Luis Joglar-Ongay
SonoSuite
Music Technology Group
Universitat Pompeu Fabra
luis@sonosuite.com

Albin Andrew Correya
Music Technology Group
Universitat Pompeu Fabra
albin.correya@upf.edu

Pablo Alonso-Jiménez
Music Technology Group
Universitat Pompeu Fabra
pablo.alonso@upf.edu

Xavier Serra
Music Technology Group
Universitat Pompeu Fabra
xavier.serra@upf.edu

Dmitry Bogdanov
Music Technology Group
Universitat Pompeu Fabra
dmitry.bogdanov@upf.edu

ABSTRACT

Web technologies are evolving every day providing higher capabilities and enabling all kinds of software. We believe that audio signal processing and other MIR related tasks should not be an exception, and there is a clear interest and need of such web tools in this community. Ideally, these tools should be developed to be as powerful as the ones already available on other languages such as C/C++ and Python.

1. ESSENTIA AS JS LIBRARY

This motivation drove our exploration on how to use Essentia [2], an open source C++ library for music and audio analysis, description and synthesis developed by the Music Technology Group of the Universitat Pompeu Fabra, in the web client as a JavaScript library using Emscripten [6] and WebAssembly [3].

The biggest strength we see in using the same code both in the browser and in native applications is the robustness of our development process. This way all our data and efforts can be put into improving the algorithm available in Essentia instead of having to maintain two implementations in two different programming languages.

1.1 Compilation

In this talk, we will present the steps to follow to use Essentia in the client, as well as some use cases. We will also discuss difficulties we encountered during the implementation, such as problems compiling Essentia for Emscripten. Furthermore, we will analyze some decisions we took along the way and other questions that are still open.

1.2 Examples

As our main example, we will show the use of Essentia in the context of SonoSuite's digital music distribution platform, where users submit their audio tracks online to be immediately analysed in the client to detect possible audio quality problems. The system gives feedback directly back to the users, allowing them to make decisions on the quality of their tracks.

The employed algorithms, now used in the browser, have been previously developed and integrated into the core C++ Essentia library in a collaboration R&D project [1]. This project aimed to develop algorithms to automatically detect the most common audio problems SonoSuite quality control team encounters on client's audio files.

2. AVALUATION AND RESULTS

Although this is still a work in process a preliminar avaluation and comparison to other JavaScript libraries, such as Meyda [5] and JS-Xtract [4] will be presented. Similar tasks will be implemented and executed using all three libraries in the browser to compare their results and efficiency.

3. FURTHER WORK

At SonoSuite, we plan to keep using Essentia to improve quality control, as well as to keep innovating to help our users for a better experience. For instance, this will include implementing automatic metadata annotations by genre and language. Finally, in this talk we will expose our ideas for further work.

4. REFERENCES

[1] P. Alonso-Jiménez, L. Joglar-Ongay, X. Serra, and D. Bogdanov. Automatic detection of audio problems for quality control in digital music distribution. In *AES 146th Convention*, Dublín, 20/03/2019 2019.

[2] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. R. Zapata, and X. Serra. Essentia: an audio analysis library for music information retrieval. In *International Society for Music Information Retrieval Conference (ISMIR'13)*, pages 493–498, Curitiba, Brazil, November 2013.

[3] A. Haas, A. Rossberg, D. L. Schuff, B. L. Titzer, M. Holman, D. Gohman, L. Wagner, A. Zakai, and J. Bastien. Bringing the web up to speed with webassembly. *SIGPLAN Not.*, 52(6):185–200, June 2017.

[4] N. Jillings, J. Bullock, and R. Stables. JS-XTRACT: A realtime audio feature extraction library for the web, 8

2016. Online. Available from http://dmtlab.bcu.ac.uk/ryanstables/JSXtractISMIR2016.pdf.

[5] H. Rawlinson, J. Fiala, and N. Segal. Meyda: an audio feature extraction library for the web audio api. In *1st Web Audio Conference*, Paris, France, January 2015.

[6] A. Zakai. Emscripten: An llvm-to-javascript compiler. In *Proceedings of the ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion*, OOPSLA '11, pages 301–312, New York, NY, USA, 2011. ACM.