

Essentia installation guide for macOS

PROJECT STRUCTURE

\$HOME/essentia_deps_static # location of statically compiled dependencies for Essentia
\$HOME/Projects/music_analysis # location of project

REQUIREMENTS (dependencies)

Core Library

essentia==2.1-beta5

Python

python==3.10

Note: Essentia C++ is built using python waf. **You MUST use Python 3.10.**

Math and DSP

eigen3==3.4.0

fftw3==3.3.10

fftw3f==3.3.10

libsamplerate==0.2.2

FFmpeg (**version 4.4 REQUIRED**)

Essentia 2.1-beta5 requires FFmpeg 4.4 and is incompatible with FFmpeg >= 5.0.

The following FFmpeg components were explicitly enabled:

- Protocols: file

- Demuxers: wav

- Decoders: pcm_s16le

- Libraries: avcodec, avformat, avutil, avresample, avdevice, avfilter, swresample, swscale, postproc

libavcodec==58.134.100

libavformat==58.76.100

libavutil==56.70.100

libavresample==4.0.0

libavdevice==58.13.100

libavfilter==7.110.100

libswresample==3.9.100

libswscale==5.9.100

libpostproc==57.1.100

Audio Fingerprinting

libchromaprint==1.5.1

Audio Metadata

taglib==2.1.1

taglib_c==2.1.1

YAML Support

yaml-0.1==0.2.5

Compression

zlib==1.3.1

STATICALLY COMPILING ESSENTIA

Inside music_analysis:

- python3.10 -m venv my_venv # **MUST use python3.10**
- source my_venv/bin/activate # activate virtual environment
- git clone <https://github.com/MTG/essentia.git>
- cd essentia
- git checkout v2.1_beta5 # latest stable version
- export PREFIX=\$HOME/essentia_deps_static # location of dependencies

- export PKG_CONFIG_PATH="\$PREFIX/lib/pkgconfig" # implicitly used by waf
- python waf clean # remove previous build
- python waf configure --mode=release --build-static --prefix=\$PREFIX --no-msse --pkg-config-path="\$PREFIX/lib/pkgconfig" # make sure that all the needed dependencies are found
- python waf # compile
- python waf install # Essentia is also installed to essentia_deps_static

EXTRACTING UTILITY FROM ESSENTIA

MUST use C++ 14

Inside GitHub repo essentia_util

- mkdir build && cd build
- cmake ..
- make

Then move to essentia_util/test and try to compile a test executable

- mkdir build && cd build
- cmake ..
- make