minispec Documentation

Release 0.1.1

The minispec development team

Feb 28, 2019

Contents

minispec is a minimal module for computing audio spectrograms.

CHAPTER 1

Getting started

1.1 Installation instructions

1.1.1 pypi

The simplest way to install *minispec* is through the Python Package Index (PyPI). This will ensure that all required dependencies are fulfilled. This can be achieved by executing the following command:

```
pip install minispec
```

or:

```
sudo pip install minispec
```

to install system-wide, or:

```
pip install -u minispec
```

to install just for your own user.

1.1.2 Source

If you've downloaded the archive manually from the releases page, you can install using the `setuptools` script:

```
tar xzf minispec-VERSION.tar.gz
cd minispec-VERSION/
python setup.py install
```

If you intend to develop minispec or make changes to the source code, you can install with *pip install -e* to link to your actively developed source tree:

```
tar xzf minispec-VERSION.tar.gz
cd minispec-VERSION/
pip install -e .
```

Alternately, the latest development version can be installed via pip:

```
pip install git+https://github.com/minispec/minispecbrew install ffmpeg` or get a␣
↪binary version from their website https://www.ffmpeg.org.
```

CHAPTER 2

Troubleshooting

If you have questions about how to use minispec, please consult the discussion forum. For bug reports and other, more technical issues, consult the github issues.

CHAPTER 3

API documentation

CHAPTER 4

Advanced topics

CHAPTER 5

Reference

5.1 Changelog

5.1.1 v0.1.1

2019-02-28

Merge minispec with librosa v0.6.3

5.1.2 v0.1.0

2018-11-29

Initial release.

5.2 Glossary

time series Typically an audio signal, denoted by `y`, and represented as a one-dimensional *numpy.ndarray* of floating-point values. `y[t]` corresponds to amplitude of the waveform at sample `t`.

sampling rate The (positive integer) number of samples per second of a time series. This is denoted by an integer variable `sr`.

frame A short slice of a *time series* used for analysis purposes. This usually corresponds to a single column of a spectrogram matrix.

window A vector or function used to weight samples within a frame when computing a spectrogram.

frame length The (positive integer) number of samples in an analysis window (or *frame*). This is denoted by an integer variable `n_fft`.

hop length The number of samples between successive frames, e.g., the columns of a spectrogram. This is denoted as a positive integer `hop_length`.

window length The length (width) of the window function (e.g., Hann window). Note that this can be smaller than the *frame length* used in a short-time Fourier transform. Typically denoted as a positive integer variable `win_length`.

spectrogram A matrix `S` where the rows index frequency bins, and the columns index frames (time). Spectrograms can be either real-valued or complex-valued. By convention, real-valued spectrograms are denoted as *numpy.ndarray*s `S`, while complex-valued STFT matrices are denoted as `D`.

onset (strength) envelope An onset envelope `onset_env[t]` measures the strength of note onsets at frame `t`. Typically stored as a one-dimensional *numpy.ndarray* of floating-point values `onset_envelope`.

chroma Also known as pitch class profile (PCP). Chroma representations measure the amount of relative energy in each pitch class (e.g., the 12 notes in the chromatic scale) at a given frame/time.

- genindex

Index