Analisis del canto de pájaros

Generated by Doxygen 1.8.6

Sat Mar 11 2017 00:14:15

Contents

Chapter 1

Canto5

Análisis de señales de canto de pajaros

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Namespace Documentation

3.1 envolvente Namespace Reference

Functions

- def envolvente

3.1.1 Detailed Description

```
Finds a wrapping-signal of a raw-audio wave, dividing the set of pints in packages of a
given number of points (bin_size), and computing the max or mean value of the signal in
each one of them.
```

3.1.2 Function Documentation

3.1.2.1 def envolvente.envolvente (*raw_audio*, *times*, *sample_rate*, *data_points* = None, *bin_size* = None, *mean* = 0, *der* = 0)

```
Input: raw_audio, timebase, sample rate and total number of data points of the audio file,
        (output of read_wav.py executed on a single file). (Controls): specify bin_size
        (=100 by default); specify if you want to consider the  maximum (mean=0, default)
       or mean (mean=1) value of the data points in bin_size; specify if you want to
       derivate the signal setting der=1 (0 by default).

Output: resultant wrapping-signal
```

3.2 find_syllables Namespace Reference

Functions

- def find_syllables

3.2.1 Detailed Description

```
Finds the time windows where the syllables are, inside the raw-audio file.
```

3.2.2 Function Documentation

3.2.2.1 def find_syllables.find_syllables (*raw_audio, times, env, t_env, npoints_umbral, fbird*)

```
Input: raw_audio, times - Amplitud and time vector of the original audio file
                          (only to plot a explanary graf).
       env, t_env       - Amplitude and time vector of the wrapping wave of the
                          audio signal.
       npoints_umbral   - Number of points counted from the start, considered
                          to estimate a threshold value "umbral" that the
                          amplitude of the syllable usually surpasses.
Output: loc_silaba      - Boolean signal identifying the time windows where the
                          syllables are. (mask)
```

3.3 get_files_paths Namespace Reference

Functions

- def get_files_paths

3.3.1 Detailed Description

```
Extract all paths of files with a certain extension inside all subfolders
of "root" folder. Also read the information contained in the file "info_file".
```

3.3.2 Function Documentation

3.3.2.1 def get_files_paths.get_files_paths (*root, extension, info_file*)

```
input:
    root: (string) root path within which the files will be searched
    extension: (string) file extension of the desired files
    info_file: (string) file name of the information file

output:
    paths_to_files: (list) each element in this list contains a tuple. The
        1st position of each tuple conteins the info (string) associated
        to the file and the 2nd position, the file path (string).

Ex: paths_to_files = get_files_paths('./datos', 'wav', 'bird_name.txt')
```

3.4 main Namespace Reference

Variables

- string **directorio** = '.'
- int **bin_size** = 1000
- float **fbird** = 1.8
- int **npoints_umbral** = 100
- tuple **loc_silabas** = find_syllables(raw_audio, times, env, t_env, npoints_umbral, fbird)
- int **margen** = 50
- tuple **silabas** = split_syllables(raw_audio, loc_silabas, margen)
- tuple **time_windows** = split_syllables(times, loc_silabas, margen)
- tuple **nro_sil** = len(silabas)

3.4.1 Detailed Description

```
Función principal. Obtiene silabas separadas y caracterizadas desde un archivo wav.
```

3.5 read_wav Namespace Reference

Functions

- def read_wav

3.5.1 Detailed Description

```
Read wav files.
```

3.5.2 Function Documentation

3.5.2.1 def read_wav.read_wav (*audio_file*)

```
Input: file path.
Output: raw audio amplitudes, corresponding times, sample rate, number of
samples.
```

3.6 spectrum Namespace Reference

Functions

- def spectrum

3.6.1 Detailed Description

```
This function calcuates the spectrogram of a signal. It also can be used
for getting the coordinates of points with a frequency-amplitude greater
than certain treshold (uncommenting the last lines).
```

3.6.2 Function Documentation

3.6.2.1 def spectrum.spectrum (*audio*, *sample_rate*)

```
input:
    audio (1D array). Signal to transform.
    sample_rate (float). Sampling frequency of the audio file corresponding
    to 'audio'.
output:
    t (1d array). Times corresponding to the second component of 'Sxx'
    f (1d array). Frequencies corresponding to the first component of 'Sxx'
    Sxx (array). Spectrogram. Frequency-amplitudes as functions of times
    and frequencies.
```

3.7 split_syllables Namespace Reference

Functions

- def split_syllables

3.7.1 Detailed Description

```
Use a boolean mask signal to split the signal.
```

3.7.2 Function Documentation

3.7.2.1 def split_syllables.split_syllables (*signal*, *boo*, *margin*)

```
input: signal (numpy array), boo (boolean array), margin (int)
output: syllables(list of numpy arrays)
```

Index