

Spec-Recorder の Read me

ライセンス:MIT ライセンス

動作環境:今のところ、Ubuntu13.04 と Windows7 で動作を確認しています。きっと MacOSX や他の Linux ディストリビューションでも動きます。HTTP 経由の画像閲覧、SR HTTP Config maker のクライアントは、Google Chrome のみ動作を確認済みです。が、きっと他のブラウザでも動きます。

必要なモジュールなど:Python2.7 と Python Imaging Library(PIL)1.1.7 と SoundExchange(SoX)14.4.1 と numpy 1.7.1、設定により Opencv 2.4(デフォルトでは不要)

概要

このソフトウェアは主に流星電波観測のための、コンピュータのデフォルトのオーディオデバイスの入力信号を録音し、スペクトログラムとして PNG ファイルに書き出し、CSV ファイルに変換し、エコーを自動でカウントするものです。

Ubuntu でのインストール方法

sudo apt-get install sox libsox-fmt-all python-imaging python-numpy
を実行します。

Opencv を使う場合は、

spec-recorder の Zip ファイルをダウンロードして、展開します。

Windows でのインストール方法

spec-recorder の Zip ファイルをダウンロードして、展開します。

<http://www.python.jp/download/> から Python2.7.5 のインストーラをダウンロードし、インストールしてください。32bit の Windows を使っている場合は x86 を、64bit の場合は x86-64 を選んでください。

<http://www.pythonware.com/products/pil/> から Python Imaging Library 1.17 for Python2.7 をダウンロードし、インストールしてください。

<https://pypi.python.org/pypi/numpy> から、numpy-1.7.1.win32-py2.7.exe をダウンロードし、インストールしてください。

<http://sourceforge.net/projects/sox/files/sox/14.4.1/> から sox-14.4.1a-win32.zip をダウンロードし、展開してください。展開したフォルダの中のすべてのファイルを spec-recorder のフォルダにコピーしてください。

ubuntu での使い方

無線機などのオーディオ出力をシステムのデフォルトのサウンドデバイスの入力につないでください

コマンドラインを開いて main.py があるフォルダに移動してください

python main.py を端末から実行してください

適宜、ボリュームを調節してください

……待つ

5分毎に一つずつ PNG ファイルと OGG ファイルと CSV ファイルが生成されます

終了させるときは、Ctrl と C を同時に押してください。

Windows での使い方

main.py をダブルクリックします

(.py のファイルが Python に関連付けられていれば、これで起動するはずですが)

起動できないときは、IDLE(Python と一緒にインストールできます)で main.py を開いて F5 を押して実行してください。Server 機能を on にしている場合は、アクセスを許可してください。

Config ファイルの書き方

server = off を server = on に書き換えると Web サーバーの機能が有効になります。他も同様です。特に変更しなくても、デフォルトのままでもとりあえず動きます。

[sox]

項目なし

[server]

server:Python 内蔵の HTTP サーバーを使う。

ポートは 8000 番、localhost:8000/latest_image.html に WEB ブラウザでアクセスすると、最新の PNG ファイルが表示されます SR HTTP Config maker を使う時も on にしてください

[file]

del_ogg: 音声ファイルを自動で削除する。(PNG ファイルは残る)

write_filename: PNG ファイルの中に時刻を書き込む

write_csv: CSV ファイルとしてデータを書き出す

[analyze]

analyze: 自動エコーカウントを有効にする

echo_threshold: エコー検知の閾値です

unite_echo_threshold: 近いエコーを同一のものとし閾値です

(echo_threshold と unite_echo_threshold はソースコードを読んで、意味を理解するまで変更しないことをおすすめします)

compared_with_pixels: エコーを検出する際に各ピクセルと比較する対象です。median もしくは average を指定してください。

get_pixel: 各ピクセルを読み込む際に使用するライブラリです。pil もしくは cv を指定してください。cv を指定した際は、OpenCV のインストールが必要となります。

エコーの自動カウント機能について

この機能は、CSV ファイルを読み込み、その中からエコーを探し、エコーの始まりの PNG ファイルの X 軸のピクセルを、echo の CSV 形式のファイルに書き出すものです。その列がエコーの数となります
この機能をより良い物にするために、実際にお使いいただき、フィードバックをいただけると幸いです。

SR HTTP Config maker について

SR HTTP Config maker は Spec-Recorder の設定ファイルを Web ブラウザ経由で編集するものです。一部、編集できない項目があります。Spec-Recorder を動かしている時でなければ、SR HTTP Config maker は使用できません。

SR HTTP Config maker に Spec-Recorder を動かしているコンピュータからアクセスするには、

<http://localhost:8000/cgi-bin/configmaker.py> を Web ブラウザで開いてください。localhost を適宜 IP アドレスに置き換えれば LAN 内のコンピュータからの編集が可能ですが、インターネット経由での編集はセキュリティ面での危険があるため、しないほうが良いと思われます。

各項目の設定を適宜選択して、送信をクリックしてください。「選択してください」のままの項目は変更されません。変更はすぐに反映されます。

もし、SR HTTP Config maker にアクセスできない場合は、cgi-bin 中の configmaker.py に実行属性を与えてください。

更新履歴

beta1	基本的な機能の実装
beta2	HTTP サーバー機能などの実装 SoX の呼び出しに用いる関数の一部変更
Ver.1.0	wave.py のみの実行で freq_exec.py も実行されるように変更 画像の自動解析の開発中
Ver.1.01	wave.py を main.py に変更 (Python の WAVE モジュールと名前が被るため) エコーの自動カウントの開発進行 (まだ使える状態ではない) CSV 形式でのデータの書き出しの実装
Ver.1.1	流星の自動カウント機能のマージ 時々 PNG ファイルが生成されないという重大な欠陥の修正 Windows 版の開発 ファイルの保存先を各フォルダに分けるように変更
Ver.1.11	Windows 対応!!! (自動カウントは未対応 PNG ファイルの生成はできる) 時刻管理の改善 音声ファイルを Ogg Vorbis に変更
Ver.1.12	Windows での自動カウントに対応

	Echo ファイルの書き出し先を echo フォルダに変更 エラーが起こってもとりあえずメインループを実行し続けるように変更
Ver.1.13	自動カウントの精度向上 モノクロの画像を monoPNG に、カラーの画像を PNG フォルダに保存するように変更し、 config ファイルの白黒、ハイカラーの設定項目を削除 config ファイルの変更を再起動不要で反映するように変更
Ver.1.14	soxi.exe を作らなくても動くように変更 自動エコーカウントの各パラメータを設定項目に追加 ネットワーク経由での Config ファイル編集機能の開発!
Ver.1.20beta	実験的な要素の多いリリース(多くのバグが含まれているはずですので、実際の観測には Ver.1.20 を待つか、Ver.1.14 をお使いください) エコー判定の際のピクセルの比較対象を、すべてのピクセルの中央値に変更(これまでは平均) エコーの周波数を自動で認識し、その周波数から離れているエコーを除外するように変更(それ に伴い、Numpy が必要になる) RMOB 形式の書き出し機能の開発中 ソースコードの整理 モジュールのインポート時のエラー処理の実装 その他、多くの変更
Ver.1.20	バグの修正 画像のピクセルの読み出しに OpenCV を用いることもできるように変更(デフォルトでは不使 用)

謝辞:SoX の開発者の方に感謝いたします