**Vocal Pitch Correction Tool (Autotune)**

Project made by Grama Andrei Razvan 1232A

**Overview**

This project involves a Python program designed to perform pitch correction (autotuning) on a vocal audio file. The program allows for two types of pitch correction methods: closest pitch correction and scale-based pitch correction. It can also generate a plot visualizing the original and corrected pitches over the spectrogram of the audio if requested.

**Generated plot**

A screen shot of a graph

Description automatically generated

**Audio Processing:**

* Loads the audio file using librosa. If the audio is stereo, only the first channel is used.
* Selects the appropriate pitch correction function based on the specified correction method.
* Calls the autotune function to perform pitch correction on the audio signal.
* Writes the pitch-corrected audio to a new file, appending \_pitch\_corrected to the original file name.

**Detailed Workflow**

Loading Audio:

The audio file is loaded with librosa.load, which returns the audio time series (y) and the sampling rate (sr). If the audio is in stereo, only the first channel is used for processing.

Pitch Tracking:

The PYIN algorithm is used to estimate the fundamental frequency (pitch) of the audio. This algorithm provides a robust way to track pitch over time, returning an array of pitch values (f0), voiced flags, and voiced probabilities.

Pitch Correction:

Depending on the correction method specified, the pitch values are adjusted:

Closest Pitch Correction: Rounds each pitch to the nearest MIDI note and converts it back to Hz.

Scale-Based Pitch Correction: Maps each pitch to the nearest pitch belonging to the specified musical scale.

Median filtering is applied to smooth the pitch trajectory, reducing artifacts and abrupt changes.

Plotting:

If the --plot flag is set, a spectrogram of the audio is generated, and the original and corrected pitch trajectories are overlaid. This plot is saved as pitch\_correction.png.

Pitch Shifting:

The PSOLA algorithm is used to adjust the pitch of the audio signal based on the corrected pitch values. This process involves modifying the audio signal to match the desired pitch trajectory while preserving the natural characteristics of the voice.

Saving the Output:

The pitch-corrected audio is saved to a new file, ensuring that the original audio is not overwritten.

**Libraries Used**

**argparse**: For parsing command-line arguments.

**librosa**: For audio analysis and processing.

**numpy**: For numerical operations.

**matplotlib**: For plotting spectrograms.

**soundfile**: For reading and writing audio files.

**scipy**: For signal processing (median filtering).

**psola**: For pitch shifting using the PSOLA algorithm.

**Usage Examples**

**Closest Pitch Correction**:

python autotune.py vocals.wav --correction-method closest

**Scale-Based Pitch Correction**:

python autotune.py vocals.wav --correction-method scale --scale 'C:maj'

**With Plotting**:

python autotune.py vocals.wav --correction-method scale --scale 'C:maj' --plot