

Programación de sistemas

TAREA 2 – AUDIO STREAM SILENCE REMOVER

Presentation

The program “assr” (Audio Stream Silence Remover) was created in the aim of removing the pseudo-silent parts of an audio stream (parts without voice or music: only noise), replacing them by a real silence. In other words, assr sets the bytes of these stream parts at the value 0.

To detect silences in the audio stream, this program split it in packets of size given by the user, and calculates the FFT (Fast Fourier Transform) of each packet. It then computes a voice harmonics percentage, obtained by dividing the energy sum of the harmonics located in the range [125Hz ; 1kHz] by the total energy of the spectrum. If this percentage is greater than the threshold fixed by the user, the packet is kept, otherwise, it is replaced by zero values.

At the end of the processing, the program displays a 400ms part of the two audio streams: the original one and the modified one. To choose the part to display, the user has to define an offset in milliseconds when calling the application.

Finally, “assr” allows the user to listen the modified audio stream, just by specifying the “p” argument during the program call.

NB : As this application only works with 8kHz sampled audio streams, it is recommended for the user to choose a value of packet size in the rank [160; 320], which represents a time segment of 20 to 40 milliseconds.

NB2 : To know exactly the calling format of the program, please read the README.txt file.

Example

For an audio recorded stream of 4 seconds, choosing packets of size 320 samples and a threshold set at 42%, the obtained result, before and after the processing, is presented on the below figure.

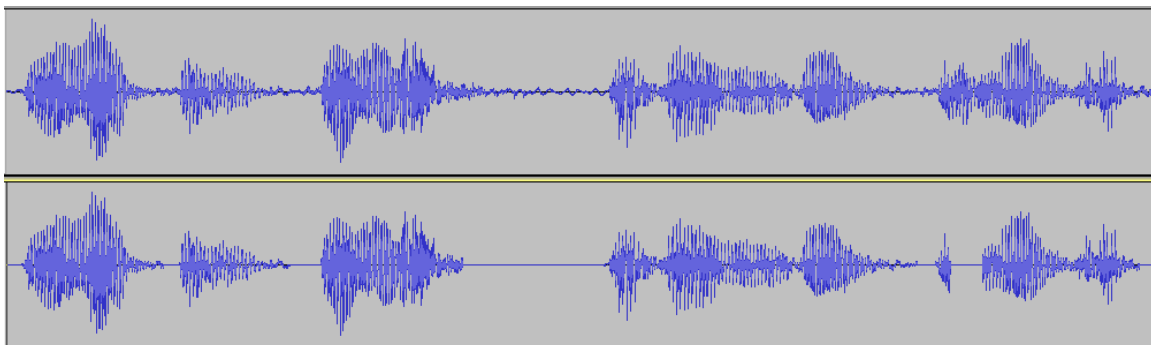


Figure 1 : Comparing before and after audio processing

Requirements

To process the song and play the result, the “assr” program uses other applications which perform calculations. The first of these applications is Octave, a GNU mathematics program available on Windows, Linux and MAC OS. Octave is used to compute the FFT of each packet and to graph a part of the result at the end of the process.

Finally, if the “p” option is specified, the program plays the processed song. To be able to do that, the “aplay” program, available for Unix systems, is called.

Please be sure to have these applications installed and added in the PATH environment variable of your Operating System before running this program.

How it works

To achieve the treatment process, “assr” creates child processes which run the third applications presented below. It communicates with them through routes called “pipes”. By this way, “assr” slaves these applications and give them all the necessary orders to perform needed calculations.

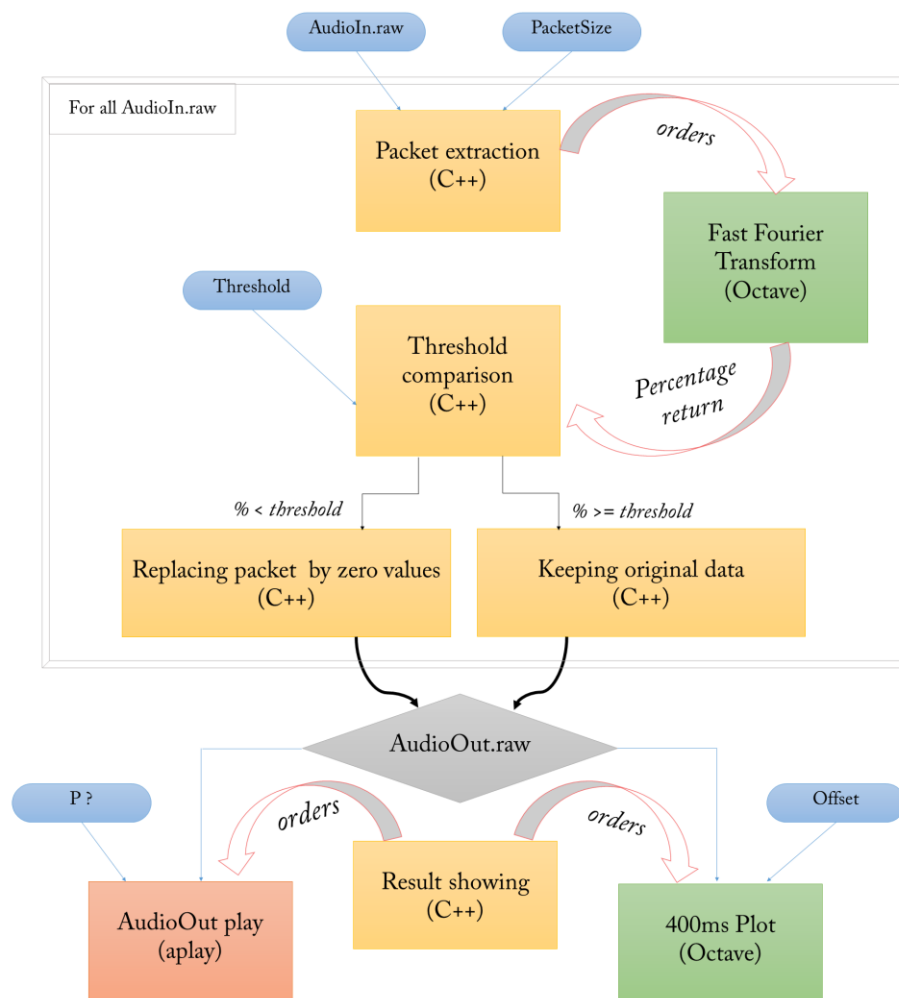


Figure 2 : Processes, treatments and communications of the “assr” application