

ENG5027

Assignment 1

Digital Signal Processing

Fourier Transform

University of Glasgow, James Watt School of Engineering

Rami Ghannam & Scott Watson

This assignment is about the Fourier Transform, how to enhance your voice and reduce artefacts.

Form groups of 4-6, work together and submit one report. Enter your team into the Wiki on moodle listing your names and matriculation numbers. There is a forum on the moodle you may use the moodle forum to find team-mates.

Record the voice with a handheld mic of one member of your group in uncompressed WAV format. Record a sentence at a Mic / speaker distance of 1cm.

Make sure that you record at least at 44.1kHz or at a higher sampling rate and that the audio is not clipping. The full audio spectrum up to 20kHz needs to be available. Low quality MP3 downloads from websites converted to WAV are not allowed and won't be marked. Reports based on recordings at sampling rates below 44kHz or downloads will not be marked.

1. Load the audio samples into python.
 1. plot the audio signals in the time domain (linear axis: normalised amplitudes -1..+1 vs time) and
 2. in the frequency domain (logarithmic axis for both frequency and amplitude in dB) with proper axis labels. Check out professional diagrams how to plot frequency spectra.

In both cases make sure you save your graphs a vector format (svg, pdf, ...)

2. Use a *vector*-drawing program (Inkscape, Illustrator, drawio, ...) and mark
 1. the *peaks* in the spectrum which correspond to the *fundamental* frequencies of the vowels spoken.
 2. Mark up the frequency *range* which mainly contains the *harmonics*.
 3. Mark up the frequency *bands* which most likely just contains noise and explain that in the context how humans produce sound and how they perceive it.

Provide brief explanations. **30%**

3. Using the speech audio you recorded, improve the quality of the voice, in particular by manipulating the frequency bands above 3kHz with the Fourier Transform. This is not about removing noise but to make the voice sounding perceptually more pleasant and interesting. Provide an explanation. **30%**

4. The so-called aural exciter sends a small amount of the sound through a non-linearity, for example \tanh and then adds this to the original signal. This makes the voice perceptually louder. Your task is to experiment with this by sending different frequency ranges through the non-linearity and then perhaps limiting the frequency band after the non-linearity and finally adding to the original signal. Here is an article about the original exciter: <https://www.muzines.co.uk/articles/aphex-aural-exciter-type-b/2850> Nowadays it's done of course digitally and your task is to create one in python. **40%**.

The report should be brief, concentrating on the technical aspects and why you have done the different steps. Do not add generic theory about voice or Fourier Transform. Just describe the method and the result. Complete PYTHON code *must* be included in the appendix and submitted via moodle alongside the report. All figures inline in the report must be high quality graphics in vector format. Blurry jpeg figures or screenshots will not be marked. Figure out early on how to add high quality graphics to your document such as EPS or SVG. Submission must be PDF. Generally Word loves SVG and LaTeX takes PDF or EPS.

Upload your code, data/WAV files to moodle in form as a single zip file. Follow *exactly* the naming conventions for all files as specified on moodle. The scripts will be tested under Linux from the command line (so not rely on Spyder, Pycharm or Vscod: your program must launch and run in the terminal). Make sure your code is platform-independent and does not contain absolute paths. Code that crashes will result in low marks. The same applies for code which won't display any plots, for example forgetting "plt.show()". Your audio files must be original ones and WAV 16 bit. No high level python signal processing / filtering commands are allowed except of the numpy FFT and IFFT commands.

We recommend that you test your submission by passing it to someone else in your group who should unpack it and run it in a new directory from the command line. It is very easy to lose marks by forgetting to include in your submitted package files you have written and that you require for you code to run.

Deadline: 21st Oct, 3pm on moodle.