STFT Exercise

Digital Processing of Single and Multi-Dimensional Signals 0510-6201
Tel-Aviv University

Submission deadline: 23/05/2019

You may submit in pairs.

In this exercise you will implement the Short-Time-Fourier-Transform (STFT) and use it to decode a phone dialer. Please read the entire exercise before diving into MATLAB, writing a clear and generic code will help you to reduce the time required to answer all questions. In addition you will be requested to submit parts of your code.

What to submit:

1. PDF file with your answers to all questions.
2. `STFT.m` - see question (5) for details.
3. `Decode.m` - see question (9) for details.
4. `Main.m` - a script that runs the entire exercise solution and recreates all the results you presented in your PDF file.

Code will be automatically tested so make sure it runs without any errors. If it fails to run the grade on it will be 0. Document your code, failing to do so will reduce your grade.

Description:

The famous “touch-tone” keypad is based on a coding method called Dual-tone multi-frequency signaling, or DTMF. It is based on a simple coding method, using combinations of two frequencies to represent unique codes for each of the keys in the keypad (See table).

Assumptions:

1) In each code both frequencies have equal amplitude.

2) Dial tones come in equal durations, and so does the intervals between them.

	1209 Hz	**1336 Hz**	**1477 Hz**
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9

You are requested to build a touch-tone decoder for a communication line with additive Gaussian noise. This means a person on the one end of the line hits a number pad, while the signal travels to the other end Gaussian noise is added to it. Your decoder should be able to decode the signal for an adequate SNR.

STFT

1) Write the definition of the STFT for a signal x, using a general window w of size M, a time-hop of size R.
2) For M $\quad R$ derive the equivalent representation and equations, and draw a diagram using a filter followed by down sampling.
3) For the frequency vector$\omega_k \quad \frac{2\pi}{M}k \; \{k \quad 0, \dots, M-1\}$, convert your diagram from (2) to a poly-phase diagram and explain.
4) Compare the computational complexity of the diagram in (3) with the one from (2).
5) Implement in MATLAB the function:

```
function S = STFT(signal, win, hopSize, F,Fs )
```

Input:

signal -input audio signal.

win - the window to be used for STFT, if **win** is a scalar a Hamming widow of length **win should be used** .

hopSize - time-hop between consecutive STFT time frames.

F - if **F** is a scalar it should be the number of frequencies to be used by the FFT $\omega_k \quad \frac{2\pi}{F}k \; \{k \quad 0, \dots, F-1\}$, in this case **S** should have **F** rows. If F is a vector it should contain the frequencies for which the STFT should be calculated at each time-step, in this case **S** should have the same number of rows as the number of elements in **F**.

Fs - the signals sampling frequency.

Output:

S - the STFT of the input signal. S should always have a number of columns equal to the number of time-steps in the signal.

Automatic testing against MATLAB's spectrogram will be applied.

Design a decoder:

6) Load the audio file **touchtone_1.wav** to MATLAB and listen to it. You should hear the touch tones of a sequence of digits.
7) Plot the Fourier transform of the entire signal and explain what you see. (The recording was done at 4096Hz.)
8) Calculate the STFT of the signal using **your implementation of the STFT** function for a variety of parameters:
   a. `win = hopSize = 512;        F = 64;`
   b. `win = hopSize = 256;        F = 256;`
   c. `win = hopSize = 1024;       F = 256;`
   d. `win = hopSize = 2048;       F = 256;`

   Discuss the result and explain the usage of `nfft` and `window` and how they affect the results.
9) Write a MATLAB function that decodes an audio signal using STFT.

```
[digits] = function decode(signal)

Input:
      signal - input, the input audio signal.
Output:
      digits - output, 1xM vector containing the M digits.
```

   Describe your considerations in selecting the different parameters: window type and length, overlap, etc.

Decode audio files:

In all audio files the noise free signal has a dynamic range of 1.

10) Decode the noise free signal **touchtone_1.wav**. What is the number sequence you decoded from it?
11) **touchtone_2.wav** contains a 1000 digit sequence, you can find the sequence in the file: **touchtone_2_sequence.mat**.

    Add Gaussian noise to the signal with increasing standard deviation σ_n.

    i. Plot the spectrogram for $\sigma_n. = 0.05, 0.1, 0.25, 0.5,1,2$.
    ii. Plot a the error rate as a function of σ_n.
12) Decode the noisy signal **touchtone_3.wav** ($\sigma_n = 0.5$). What is the number sequence you decoded from it?