Lab 1: Sampling and Quantization (Audio)

Introduction

The purpose of this lab is for you to understand the principles of sampling and quantization with an audio signal. The effect of sampling rate conversion and quantization will be investigated. By comparing the sound quality obtained with different sampling rates, quantizers, and pre-filters, you should have a better appreciation of the differences in performances by using pre-filters and quantizers.

Since OpenCV does not contain any functionality to read audio files, it is recommended you use MATLAB for this lab.

Audio file

The provided "ELE725_lab1.wav" file should be used as the reference audio file.

Audio File Properties

1. Read the file using audioread().
2. Find properties of the audio file such as sampling rate, bit rate etc. (hint: there is a built-in MATLAB function).
3. Calculate the file size from the properties.
4. Compare the file size with the file size that you get from the operating system. Show calculations in your report.

Sampling

In this part of the lab, you will simulate sampling rate conversion with/without the use of pre-filters and/or interpolation.

1 Construct a function `DownSample(inFile,outFile, N,pf)` that loads an audio sequence from inFile, then downsamples and plays it back, and saves the downsampled file as output.
   - N is the downsampling factor (e.g. keep every N^{th} sample).
   - pf is a boolean flag that indicates whether or not a pre-filter should be used (e.g. using `decimate()` to downsample. Read MATLAB help for filter options)

2 Create a reconstructed version of the original waveform using `interp()`.

3 Compute and display the spectra of the original, downsampled and reconstructed signals for different values of N (e.g. N=2,4,8). [Hint: you may use functions such as fft(), specgram() and spectrum() here - see MATLAB help]. Add spectrums in your report and comment on them.

4 Playback and listen to original, down-sampled and reconstructed versions (using sound()) and comment on the differences in your report.

Quantization

You are asked to build two MATLAB programs to implement uniform quantization and mu-law companding respectively:

```
UniformQuant (inFile, outFile, N)
MulawQuant (inFile, outFile, N, Mu)
```

Do NOT use MATLAB built-in functions such as compand() to implement these functions. However you are free to use built-in mathematical functions such as log(), sign(), etc.

`UniformQuant()` quantizes an input sequence using a uniform quantizer with a user-specified number of levels, where N = number of bits. You can use either mid-rise or mid-tread quantization (the only difference is in the quantization function, floor() vs round(), see lecture). The input should be reconstructed back before being saved/played.
`MulawQuant()` performs the mu-law scaling operation prior to applying `UniformQuant()`, and then reverses the process back prior to being saved/played.

1. Apply `UniformQuant`() to the input sound, with quantization level set to 8 bits (N=8), 4 bits (N=4), and 2 bits (N=2) respectively.

2. Compare the original sequence and quantized sequences with different levels, in terms of perceptual sound quality and the waveform. Also record and compare the quantization (MSE) error between the original and quantized samples.
3. Apply `MulawQuant`() to the input, with quantization level set to 8 bits (N=8), 4 bits (N=4), and 2 bits (N=2), respectively, with Mu=100. Compare the original sequence and quantized sequences with different levels.
4. Plot a part (200 samples) of the original sequence and the quantized sequences obtained with the uniform quantizer and that with the mu-law quantizer with the same number of quantization levels in the same graph. Which quantizer yields smaller (MSE) error for signal values that are small, and larger error for signal values that are large? Comment in your report.

Report:
Include as much information possible in your report. But do not exceed the 4 page limit (excluding cover page and references). Be concise and creative!

Submission:

You must demo your lab to your TA during the lab session. All group members MUST be present. Submit report (PDF in IEEE format as per D2L), and source files (MATLAB *.m) on your D2L submission folder. There will be one submission folder per group. **You don't need to submit the audio files you saved during the lab**, just keep them handy for the demo.

DEMO DUE: **week of February 4, lab session**
REPORT DUE: **February 8, 11.59 PM**

RUBRIC	5 (pts)	5 (pts)
	Code and Demo	Report