Audio- / Videosignalverarbeitung
Advanced Digital Signal Processing
Digital Signal Processing 2

Seminar 3

WS 2019/2020

Homework assignment

1. Write a framework to simulate real-time audio reading-processing-playing back:
   a) read audio file
   b) take piece by piece this audio file (in literature or Internet it is called CHUNCK)
   c) choose reasonable CHUNCK size, e.i 512 samples (explain your choice)
   d) process it
   e) play it back
   f) implement several control buttons

Homework assignment

In the framework has to be implemented following signal processing:

1. Filter the speech signal before downsampling with the given FIR and IIR filters (button switch)

   Hint: use `signal.lfilter(b, a, speech)`

2. Downsample the given speech signal (32kHz) to 8 kHz

   a) Use the .wav-file which is uploaded at moodle in unit „Seminar 3“

   b) Downsampling by the factor 4

   c) Plot real-time spectra of the original and processed signal

Hint: use signal.freqz(...) for creating the spectrum

See additional slide for animation

Homework assignment

In the framework has to be implemented following signal processing:

3. Upsample the signal to reconstract original sampling rate
4. Filter the Upsampled speech signal with the given filters (button switch)

   Hint: use `signal.lfilter(b, a, speechUp)`

5. Play back the filtered signal after downsampling and after upsampling

Homework assignment

FIR lowpass filtering

a) Implement the **FIR lowpass** filter with the following difference equation:

$$y(n) = 0.3235 * x(n) + 0.2665 * x(n-1) + 0.2940 * x(n-2) + \\ + 0.2655 * x(n-3) + 0.3235 * x(n-4)$$

b) Plot the impulse response (first 50 samples)

c) Plot the frequency response

Homework assignment

IIR lowpass filtering

a) Implement the **IIR lowpass** filter with the following difference equation:

$$y(n) = 0.256 * x(n) + 0.0512 * x(n-1) + 0.256 * x(n-2) + \\ + (1.3547 * y(n-1) - 0.6125 * y(n-2))$$

b) Plot the impulse response (first 50 samples)

c) Plot the frequency response

Homework assignment

Compare FIR and IIR lowpass filters:

- Transfer functions
- Make your own conclusions about comparison
- Which filter is better? Why?

Homework assignment

Real-time animation (Tutorial link)

```
import matplotlib.animation as animation

fig, (p1, p2) = plt.subplots(2, 1)
fig.set_size_inches(15, 8, forward=True)
data = np.zeros(len(global_variable_name_1))
original_spectrum, = p1.plot(data, animated=True)
processed_spectrum, = p2.plot(data, animated=True)

p1.set_xlim(x_min, x_max)
p1.set_ylim(y_min, y_max)
p1.grid(True)
p1.set_title('Title')
fig.subplots_adjust(hspace=.5)

p2.set_xlim(x_min, x_max)
p2.set_ylim(y_min, y_max)
p2.grid(True)
p2.set_title('Title')
fig.subplots_adjust(hspace=.5)

def animate(i):
    # updated global variables

    original_spectrum.set_ydata(global_variable_name_1)
    original_spectrum.set_xdata(range(len(global_variable_name_1)))

    processed_spectrum.set_ydata(global_variable_name_2)
    processed_spectrum.set_xdata(range(len(global_variable_name_2)))

    return [variable_name]

ani = animation.FuncAnimation(fig, animate, range(10000), interval=25, blit=True)
plt.show()
```