MATLAB Exercise – LPC Vocoder

Program Directory: `matlab_gui\lpc_vocoder`
Program Name: `lpc_vocoder_GUI25.m`
GUI data file: `lpc_vocoder.mat`
Callbacks file: `Callbacks_lpc_vocoder_GUI25.m`
TADSP: Problem 9.36

This MATLAB exercise builds an LPC vocoder, i.e., performs LPC analysis and synthesis on a speech file, resulting in a synthetic speech approximation to the original speech. The LPC analysis uses a standard autocorrelation analysis to determine the sets of LPC coefficients, on a frame-by-frame basis, along with the frame-based gain, `G`. An independent analysis method (a cepstral pitch period detector) classifies each frame of speech as being either voiced speech (with period determined by the location of the cepstral peak in a designated range of pitch periods) or unvoiced speech (simulated by a random noise frame) designated as a frame pitch period of 0 samples. The independent analysis provides a two-state excitation function for the LPC synthesis part of the processing, consisting of a series of pitch pulses (during voiced frames) and/or noise sequences (during unvoiced frames).

LPC Vocoder – Theory of Operation

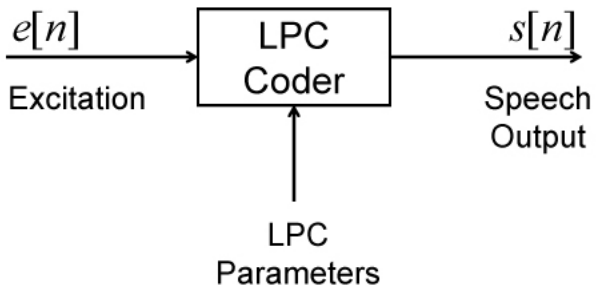

Figure 1: LPC Model of Speech Synthesis

This MATLAB program analyzes and synthesizes an utterance using the LPC synthesis model of Figure 1. In order to synthesize the utterance, the speech signal is analyzed by an LPC autocorrelation analysis method in order to obtain the LPC parameters at an appropriate update rate. The speech signal is also analyzed by a cepstral pitch detector in order to create the appropriate excitation signal, again at the same rate as the LPC parameters.

A zero value of the pitch period estimate indicates that the speech frame was classified as unvoiced (or silence/background signal), and a non-zero value is the pitch period estimate (in samples at an 8 kHz sampling rate) when the frame is voiced. Using the excitation signal created from the cepstral pitch detector analysis, and the predictor data that was computed periodically throughout the speech, and, using a conventional autocorrelation LPC analysis method, the task is to synthesize a synthetic version of the original sentence that preserves as much of the intelligibility and naturalness of the original speech as possible.

One of the challenges of this problem is that you will have to think about how to properly create the excitation signal of Figure 1 so that it provides a close approximation to the real speech excitation signal, $e[n]$, from the LPC analysis, and is periodic at the appropriate period for voiced speech and noise-like with the correct gain for unvoiced/silence/background signals.

In the course of working with this LPC Vocoder you should think about the following issues and how they affect the quality of your synthesis:

1. Note that what goes on in a given frame is *not* independent of what happened in previous frames. For example, as the pitch period changes, you will need to know where the last pitch impulse occurred in the previous frame so as to determine the location of the next impulse in the current frame.

2. You can change the vocal tract filter only once per frame, or you can interpolate between frames and change it more often (e.g., at each new pitch period). What parameters can you interpolate and still maintain stability of the resulting synthesis?

3. You don't have to quantize the vocal tract filter and gain parameters, but you should consider doing this if you are satisfied with the quality of your synthesized speech utterance.

4. Listen to your synthetic speech and see if you can isolate the main sources of distortion.

5. Implement a "debug" model where you show the "true excitation signal"; i.e., the residual LPC analysis error, $e[n]$, and the synthetic excitation signal that you created as well as the resulting synthetic speech signal and the original speech signal, for any frame or group of frames of speech. Using this debug mode, see if you can refine your estimates of the key sources of distortion in the LPC Vocoder.

6. How could you take advantage of the overlap-add method of short-time analysis and synthesis to simplify your synthesis procedure and make it less sensitive to frame phasing and alignment errors.

LPC Vocoder – GUI Design

The GUI for this exercise consists of three panels, 4 graphics panels, 1 title box and 18 buttons. The functionality of the two panels is:

1. one panel for the graphics display,

2. one panel for parameters related to the linear prediction analysis and synthesis, and for running the program,

3. one panel for playing back the original speech file, the re-synthesized speech file, the excitation without gain, and the excitation with gain.

The set of four graphics panels is used to display the following:

1. the original speech waveform,

2. the excitation signal for synthesis,

3. the pitch period contour as extracted by the cepstral pitch detector,

4. the resulting synthetic signal.

The title box displays the information about the selected file along with the set of LPC analysis/synthesis parameters. The functionality of the 18 buttons is:

1. a pushbutton to select the directory with the speech file that is to be analyzed using short-time analysis methods; the default directory is 'speech_files',

2. a popupmenu button that allows the user to select the speech file for analysis,

3. an editable button that specifies the sampling rate, `fsr`, for new recordings; (the default value for `fsr` is 10000 samples per second),

4. an editable button that specifies the number of seconds of recording, `nsec`; (the default value for `nsec` is 2 seconds),

5. a pushbutton that starts the speech recording process, along with a user selection of the region of speech of interest for LPC vocoding,

6. a pushbutton that displays the speech signal (either retrieved from the speech directory or recorded in the current session) and allows the user to select the starting and ending samples for LPC vocoding, using the graphics cursor,

7. an editable button that specifies the frame duration, L_m, (in msec) for short-time analysis; (the default value is $L_m = 40$ msec),
8. an editable button that specifies the frame shift, R_m, (in msec) for short-time analysis; (the default value is $R_m = 10$ msec),
9. a popupmenu button that specifies the range for search for the pitch period of a voiced speech frame, `imf`; (the choices for `imf` are male pitch, female pitch, or combined range pitch and the default is male pitch range);
10. a popupmenu button that specifies the type of pitch detector for performing pitch analysis of the utterance; (the only valid choice currently is cepstral pitch detection),
11. a popupmenu button that specifies the synthesis method, `isyn`; (the choices for synthesis are 10 msec excitation frames or 40 msec overlap-added frames; (the default value is 10 ms excitation, but the preferred method is the 40 msec overlap-added frames),
12. a pushbutton to run the code and display the results of short-time LPC analysis on the four graphics panel displays,
13. a pushbutton to play the original speech file,
14. a pushbutton to play the synthesized signal,
15. a pushbutton to play the excitation signal without the excitation gain,
16. a pushbutton to play the excitation sign with the excitation gain,
17. a pushbutton to close the GUI.

LPC Vocoder – Scripted Run

A scripted run of the program 'lpc_vocoder_GUI25.m' is as follows:

1. run the program 'lpc_vocoder_GUI25.m' from the directory 'matlab_gui\lpc_vocoder',
2. hit the pushbutton 'Directory'; this will initiate a system call to locate and display the filesystem for the directory 'speech_files',
3. using the popupmenu button, select the speech file for short-time feature analysis; choose the file 'we were away a year ago_lrr.wav' for this example,
4. using the editable buttons, set the initial values for the recording conditions as `fsr=10000` Hz, and `nsec=3` seconds; this step can be skipped if no new recording is desired,
5. using the editable buttons, set the initial values for the analysis/synthesis frames as $L_m = 40$ msec and $R_m = 10$ msec,
6. using the popupmenu button, set the range of pitch period to male range,
7. using the popupmenu button, set the pitch detector method to cepstral,[1]
8. using the popupmenu button, set the synthesis frame overlap-addition method to 40 msec excitation frames,
9. hit the 'Run LPC Vocoder' button to perform the computations for LPC analysis, cepstral pitch detection, and LPC synthesis; the graphics panels display the signal waveforms of the LPC synthesizer, as well as the pitch period contour of the utterance',
10. hit the 'Play Original Speech' button to play the original speech file,

[1] currently the only working method is the cepstral pitch detector method

11. hit the 'Play Synthesized Speech' button to play the LPC synthesis,

12. hit the 'Play Excitation Without Gain' button to play the excitation as constant amplitude pitch pulses or unit variance noise,

13. hit the 'Play Excitation With Gain' button to play the excitation as variable amplitude pitch pulses or variable amplitude noise,

14. experiment with different choices of speech file, and with different values for the LPC analysis/synthesis parameters and with different pitch period ranges,

15. hit the 'Close GUI' button to terminate the run.

An example of the graphical output obtained from this exercise using the speech file 'we were away a year ago_lrr.wav' is shown in Figure 2. The graphics panels show the speech waveform in the top graphics panel, the excitation signal in the second graphics panel, pitch period contour in the third graphics panel, and the the synthesized signal in the bottom graphics panel.

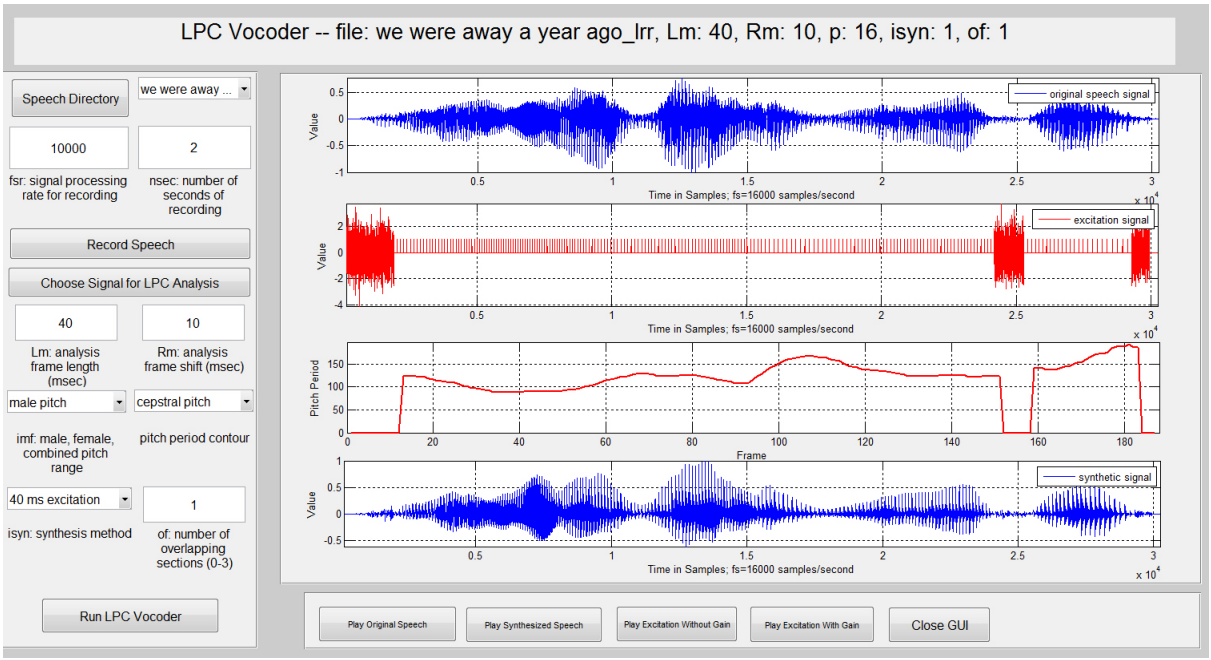

Figure 2: Graphical output from LPC Vocoder exercise. The graphics panels show the speech waveform in the top graphics panel, the excitation signal in the second graphics panel, the synthesized signal in the third graphics panel, and the pitch period contour in the bottom graphics panel.

LPC Vocoder – Issues for Experimentation

- choose a speech file for LPC vocoding using the default LPC analysis parameters. What is the quality of the synthetic speech from the LPC vocoder? Experiment with the LPC analysis parameters and determine which ones increase the quality of the synthesis, and which ones decrease the quality of the synthesis. Why do these changes in quality appear? Is there any logic to how the individual parameters affect synthesis quality that you discern from listening to the synthetic speech files?

- there are several variations on how the excitation excites the vocal tract system via the method of overlap-addition or just by summing filter outputs with overlapping tails. Experiment with the various synthesis parameters (isn and of) and see how the synthetic speech quality varies with changes in these synthesis parameters.