MATLAB Exercise – Waveform Similarity and Overlap Add (WSOLA) for Speech and Audio

Program Directory: `matlab_gui\WSOLA`
Program Name: `WSOLA_GUI25.m`
GUI data file: `WSOLA.mat`
Callbacks file: `Callbacks_WSOLA_GUI25.m`
TADSP: Problem 7.29

This MATLAB exercise implements the Waveform Similarity and Overlap Add (WSOLA) method of Verhelst and Roelands (An Overlap-Add Technique Based on Waveform Similarity (WSOLA) for High Quality Time-Scale Modification of Speech, Werner Verhelst and Marc Roelands, ICASSP 1993, pp. II-554-II-557).

The MATLAB exercise speeds-up or slows-down a speech or audio file by a factor of α, where α is nominally any value in the range of 0.33 (slow-down by a factor of 3-to-1) to 4.0 (speed-up by a factor of 4-to-1).

The WSOLA method is used to best align each signal block of the rate changed signal to the ideal signal block (no rate change) at each frame (overlapping window) in order to minimize the distortion due to phase differences at the frame boundaries. Overlapping windows are weighted by a suitable window to reduce the remaining effects of discontinuities at the boundaries between frames. The frame generation and overlap parameters, along with the duration of the alignment offset parameter, are algorithm variables that must be specified.

WSOLA – Theory of Operation

The signal processing parameters of the WSOLA algorithm graphical user interface include the following:

- filename: speech/audio file for processing (e.g., 'we were away a year ago_lrr.wav')
- L_m: analysis frame length (msec) (5-80); (default is L_m=40 msec)
- R_m: analysis frame shift (msec) (1.25-40); (default is R_m=10 msec)
- `wtype`: window type (0-rectangular, 1-Hamming, 2-triangular) (default is Hamming)
- `deltamax`: maximum frame offset for matching (msec) (0-10); (default is `deltamax=5` msec)

The parameters `Lm, Rm`, and `deltamax` need to be converted from the msec range to the sample range at the sampling rate of f_s samples per second, using the relations:

1. `L` = round($L_m * f_s/1000$) (samples),
2. `R` = round($R_m * f_s/1000$) (samples),
3. `deltas` = round(`deltamax` * $f_s/1000$) (samples).

Thus for a sampling rate of f_s=16000 Hz and frame parameters of L_m=40, R_m=10, `deltamax=5`, all in msec, the WSOLA signal processing parameters become `L=640, R=160, deltas=80` samples.

The basic signal processing for WSOLA is straightforward. Frames of the signal at both the original speed and the changed speed are created. The changed speed signal generates a frame of speech every `alpha*R` samples. Each such frame is of length `L` samples. This frame of speech (which we call `xreal` for processing purposes) is expanded in length by `deltas` samples at both the beginning of the frame and at the end of the frame. The goal of WSOLA processing is to find the sub-frame of length `L` samples, within the speech array `xreal`, that best aligns with an ideal frame of speech of length `L` samples that begins `R` samples after the previous overlap added frame. The best sub-frame is found by cross correlating the expanded length speech frame (`xreal`) with the ideal frame (`xideal`) that would be used in a direct overlap add synthesis. Once the index of best alignment between `xreal` and `xideal` has been found, a frame of length `L` samples, beginning at the index of maximum correlation is saved in the array `xadd`, windowed by the designated window, and overlap added into the output array `yout`. The process of generating new frames for

overlap addition is iterated throughout the speech duration using a shift of `alpha*R` samples for each successive frame.

Thus to effect a speed-up or slow-down by a factor of `alpha`, we need to form and keep track of (on a frame-by-frame basis) four arrays which we label as:

1. `xreal`, an array of nominal length `L` samples. This array accounts for the speed change by shifting the starting frame value by `alpha*R` samples (from that of the previous frame), as would be needed for the rate changed signal. In order to account for mis-alignment of the speech samples at the frame boundaries, due to the rate change, the `xreal` array is augmented by `deltas` samples at both the beginning of the array and at the end of the array. The expanded size of the `xreal` array is `L+2*deltas` samples.

2. `xideal`, an array of length L samples that ideally overlap adds at the current frame with no speed change, i.e., for each frame the starting sample for the `xideal` array is shifted by R samples from the best match of the previous frame.

   To best align the pair of arrays, `xideal` and `xreal`, the two arrays are cross correlated. The index of maximum correlation (`maxind`) is used as the starting index of the expanded `xreal` array at the next iteration, and the speech samples from `xreal(maxind:maxind+L-1)` are stored in the array (`xadd`), windowed by the designated window, and overlap added to the output array, `yout`.

   The MATLAB function, `xcorr(xreal, xideal)`, is used to cross correlate the arrays `xideal` and `xreal` and is very efficient for this task.

3. `xadd` is the array of `L` samples that correspond to the best alignment between `xideal` and the extended array `xreal` (i.e., the frame of `L` samples within `xreal` that gives the highest cross correlation with `xideal`. This resulting array is windowed using the designated window, and overlap added to the speech output array

4. `yout` is the speech output array that stores the overlap added frames and ultimately represents the rate changed speech signal.

WSOLA Steps (The user can skip the next two sections as they contain programming details that are not essential to the running of the WSOLA exercise code.)

There are four arrays that are used in the WSOLA implementation, as discussed above. We use the following notation to describe which speech samples are used at each frame of the WSOLA processing:

- `xreal_start(n)` is the starting sample for array `xreal` at frame `n`; the duration of array `xreal` is `L+2*deltas` samples
- `xideal_start(n)` is the starting sample for array `xideal` at frame `n`; the duration of array `xideal` is `L` samples
- `maxind(n)` is the index into array `xreal` which corresponds to the point of maximum correlation between the arrays `xideal` and the expanded array `xreal` at frame `n`
- `xadd_start(n)` is the starting sample for array `xadd` as determined by `maxind`; the duration of array `xadd` is `L` samples
- `yout_start(n)` is the starting sample for array `yout`; the duration of array `yout` is `L` samples

Thus the WSOLA algorithm runs as follows. For frame 0 (initialization):

- `xreal_start(0)=1;`
- `xideal_start(0)=1;`
- `maxind=1;`

- `xadd_start(0)=1;`
- `yout_start(0)=1;`

For frame 1 we have:

- `xreal_start(1)=1;`
- `xideal_start(1)=1+R;`
- `maxind(1)=R+1;`
- `xadd_start(1)=R+1;`
- `yout_start(1)=R+1;`

For frames 2 to `nfrm` (where `nfrm` is the number of frames at the changed speech rate), we have:

- `xreal_start(n)=xreal_start(n-1)+alpha*R;`
- `xideal_start(n)=xadd_start(n-1)+R;`
- `xadd_start(n)=xreal_start(n)+maxind-1;`
- `yout_start(n)=yout_start(n-1)+R;`

Consider the following example that illustrates the operation of WSOLA.

1. Assume the signal processing parameters are set to `fs=16000, L=640, R=160, deltas=80, alpha=0.5` and we use a Hamming window for the overlap addition process.
2. We block the speech into overlapping frames of length L samples, with frame shift of R samples.
3. For frame 0 we initialize all the four arrays to the first L samples of the signal. The initial output array, `yout`, uses the window weighted first frame as its initialization. Thus the zeroth frame of the four arrays is defined as:
   - `xideal(1:L)=s(1:L), L=640`
   - `xreal(1:L)=s(1:L)`
   - `xadd(1:L)=s(1:L)`
   - `yout(1:L)+s(1:L)*window(1:L)`

   We also define the best index of the array `xadd`, as obtained from the correlation routine, as `maxind=1` for the first frame of signal.
4. For frame 1 the ideal array starting sample is shifted by R samples from the starting sample of the best match array, `xadd`, in order to be consistent with the overlap addition process at the speeded-up or slowed-down rate. The real array starting sample is nominally shifted by `alpha*R` samples; however for this first real frame, the starting sample is extended backwards by `deltas` samples, so the extended array begins at sample `alpha*R-deltas` and is of length `L+2*deltas` samples.
5. The next step in the WSOLA method is to cross-correlate the arrays `xideal` and `xreal` and to find the index of maximum correlation, `maxind`. Using the index of maximum correlation, the frame of length L samples, beginning at the index of maximum correlation and of length L samples is saved in the array `xadd`. Thus the array `xadd` stores the best match between the ideal and the real speech arrays.
6. Finally we complete the computation at frame 1 by windowing the signal in array `xadd` and overlap adding it into output array `yout` for the range of indices of the best matching part of the `xadd` array.
7. We iterate this process for each frame in the signal.

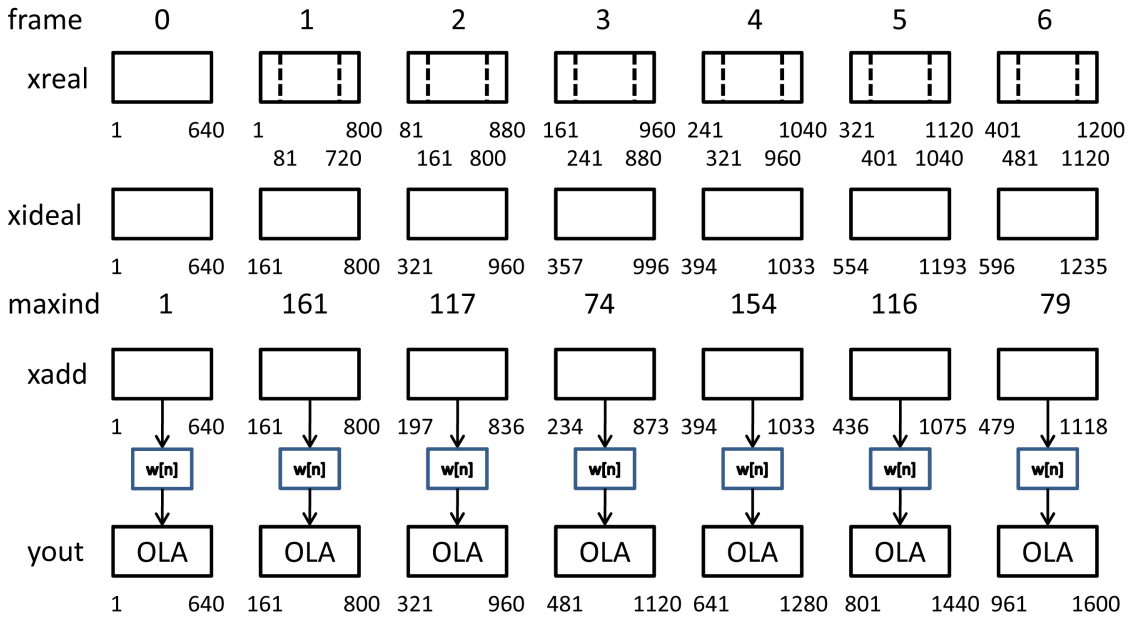

Figure 1: Frame-by-frame WSOLA processing showing the starting and ending indices of each of the four arrays used in WSOLA along with the maximum correlation index.

fno	xideal	xreal	xadd	yout	maxind
1	(1:640)	(1:640)	(1:640)	(1:640)	1
2	(161:800)	(1:800)	(161:800)	(161:800)	161
3	(321:690)	(81:880)	(197:836)	(321:960)	117
4	(357:996)	(161:960)	(234:873)	(481:1120)	74
5	(394:1033)	(241:1040)	(394:1033)	(641:1280)	154
6	(554:1193)	(321:1120)	(436:1075)	(801:1440)	116
7	(596:1235)	(401:1200)	(479:1118)	(961:1600)	79

Table 1: Array values of example for first 7 frames.

WSOLA Example

The following real example illustrates the use of the WSOLA algorithm on the sentence 'we were away a year ago_lrr.wav' with:

```
alpha=0.5,  fs=16000, L=640, R=160, deltas=80, window=Hamming
```

The results of the computation are shown in Figure 1 and Table 1.

- Frame 0 initializes all four arrays to the speech signal samples from 1 to 640, and initializes the parameter `maxind` to the value 1, as discussed above.
- Frames 1 through the last frame of the signal assigns speech samples to the four arrays by following the starting sample array rules, as given above. (Note that for frame 1, the starting sample for array `xreal` is offset by `deltas`.)
- The results for the first 7 frames are shown in Table 1 and Figure 1.
- The resulting speeded-up or slowed-down speech is played to verify the performance of the WSOLA algorithm.

WSOLA – GUI Design

The GUI for this exercise consists of two panels, 2 graphics panels, 1 title box and 14 buttons. The functionality of the two panels is:

1. one panel for the graphics display,

2. one panel for parameters related to the WSOLA signal processing, and for running the program.

The graphics panels are used in two modes, namely:

1. a frame-by-frame mode that cycles through the speech or audio file on a frame-by-frame basis in order to illustrate the best alignment between the ideal overlap-added frame and the real frame that is separated in time by the speed-up or slow-down factor,

2. the normal WSOLA speed-up or slow-down mode which performs all the WSOLA computations

In frame-by-frame mode the set of two graphics panels is used to display the following:

1. the ideal speech frame and overlaid with it the best aligned speech frame in the upper graphics panel,

2. the modified correlation between the real and ideal frames showing the location of the offset of highest correlation (dashed blue line) in the bottom graphics panel

In normal operation the set of two graphics panels is used to display the following:

1. the upper graphics panel shows the original speech or audio signal,

2. the lower graphics panel shows the speeded-up or slowed-down speech or audio signal.

The title box displays the information about the selected file used for WSOLA analysis and the associated signal processing parameters. The functionality of the 14 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 duration (in seconds) of a new recording; (default is 3 seconds),

4. an editable button that specifies the sampling rate, `fsr`, of a new recording; (default is 16,000 Hz),

5. a pushbutton to begin recording a new speech or audio file,

6. an editable button that specifies the frame duration, L_m, (in msec) for short-time analysis; (the default value is L_m=40 msec),

7. an editable button that specifies the frame shirt, R_m, (in msec) for short-time analysis; (the default value is R_m=10 msec),

8. an editable button that specifies the maximum deviation in time between ideal and real speech frames used in the WSOLA signal processing; (the default value is `deltamax`=5 msec),

9. a popupmenu button that lets the user choose either a Hamming or Triangular or Rectangular window for short-time analysis; (the default value is a Hamming window),

10. an editable button that selects the WSOLA running mode; there are three choices for running mode, namely a full run of WSOLA on a standard speech or audio file, or from a new recording; a frame-by-frame mode, with no pause between frames, which lets the user see the degree of overlap between the ideal overlap-add frame, and the real overlap-add frame based on the speed-up/slow-down factor, alpha; and finally a frame-by-frame mode, with a pause between frames, where the user hits a carriage return to advance the frames, one-at-a-time; the user can leave the frame-by-frame mode by changing this button to the terminate frame mode option, being careful to hit one last carriage return, thereby finishing the frames with the no-pause feature; the default run mode is a full run; it is instructional to set the scaling factor `alpha` to 1 and set the run mode to frame-by-frame run with no delay between frames, since, in this case, the alignment between ideal and real overlap-add frames is essentially perfect and the output is identical to the input,

11. an editable button that specifies `alpha`, the speed-up or slow-down factor for the WSOLA analysis; values of `alpha < 1` are for speech/audio slowdown, and values of `alpha > 1` are for speech/audio speedup,

12. a pushbutton to run the code and display the resulting waveforms on the (original and modified signals) graphics panel displays,

13. a pushbutton to play out in sequence the original speech or audio signal, followed by the speeded-up or slowed-down signal, followed by the original signal,

14. a pushbutton to close the GUI.

WSOLA – Scripted Run

A scripted run of the program 'WSOLA_GUI25.m' is as follows:

1. run the program 'WSOLA_GUI25.m' from the directory 'matlab_gui\WSOLA',

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, choose an initial value of 40 msec for the frame length (L_m), 10 msec for the frame shift, R_m, and 5 msec for the maximum frame offset,

5. select the 'full WSOLA' run mode as the initial run,

6. set the speed-up or slow-down factor, `alpha`, to 0.5 for a slow-down (by a factor of 2),

7. hit the 'Run WSOLA' button to initiate the full run of the WSOLA algorithm with the user-selected parameters; the program processes the input and plays out the time-adjusted speech/audio file,

8. hit the 'Play Orig | WSOLA | Orig' button to play out, in sequence, the original speech or audio file, followed by the speeded-up or slowed-down signal, followed by the original speech or audio file,

9. experiment with different choices of speech file, and with different values for L_m, R_m, window type, `deltamax`, and `alpha`,

10. experiment with the frame-by-frame modes to see exactly how the ideal and real overlap-add frames provide the basis for speeding up and slowing down speech or audio files,

11. 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' and running in debug mode is shown in Figure 2. The graphics panels show the waveforms of the current frame for both the best-aligned and ideal frames (top graphics panel), and the modified correlation between the real and ideal frames, showing the offset that provides the best alignment between frames as indicated by the dashed blue vertical line (bottom graphics panel).
An example of the waveforms resulting from normal WSOLA operation with a slow-down factor of `alpha=0.5` is shown in Figure 3. The upper graphics panel shows the waveform of the original signal and the lower graphics panel shows the waveform of the signal slowed-down by a factor of 2 using a value of `alpha=0.5`.
It is interesting and entertaining to use WSOLA on an audio selection. The file 'Maple_short.wav' (included in the speech files directory), provides a good example (order of 3 seconds of Maple Leaf rag), for WSOLA analysis and experimentation.

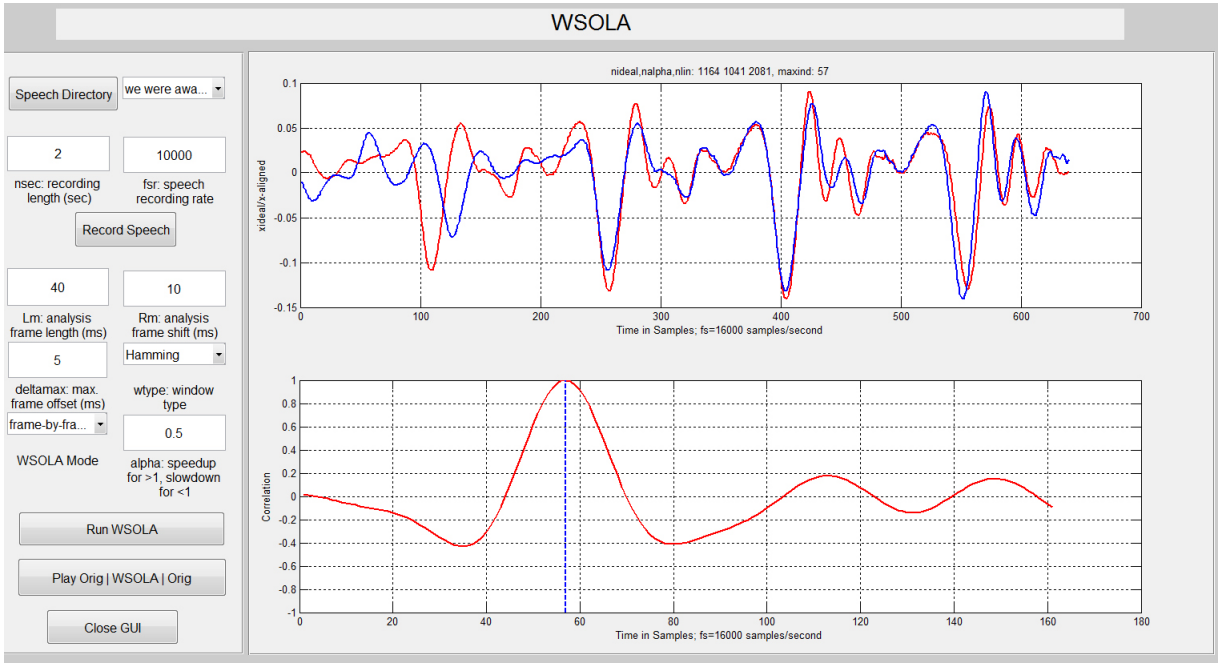

Figure 2: A single frame from the frame-by-frame analysis of WSOLA operations. The graphics panels show the waveforms of the current frame for both the best-aligned and ideal frames (top graphics panel), and the modified correlation between the real and ideal frames, showing the offset that provides the best alignment between frames as indicated by the dashed blue vertical line (bottom graphics panel).

WSOLA – Issues for Experimentation

1. run the scripted exercise above, and answer the following:

   - using the default parameters for the WSOLA signal processing parameters, access the quality of the speeded-up and slowed-down speech for the following values of `alpha`:
     - `alpha=0.5` (slow-down to half rate)
     - `alpha=0.75` (slow-down to three-quarters rate)
     - `alpha=1` (no change in speech rate)
     - `alpha=1.5` (speed-up by 50%)
     - `alpha=2.0` (speed-up by 100%)

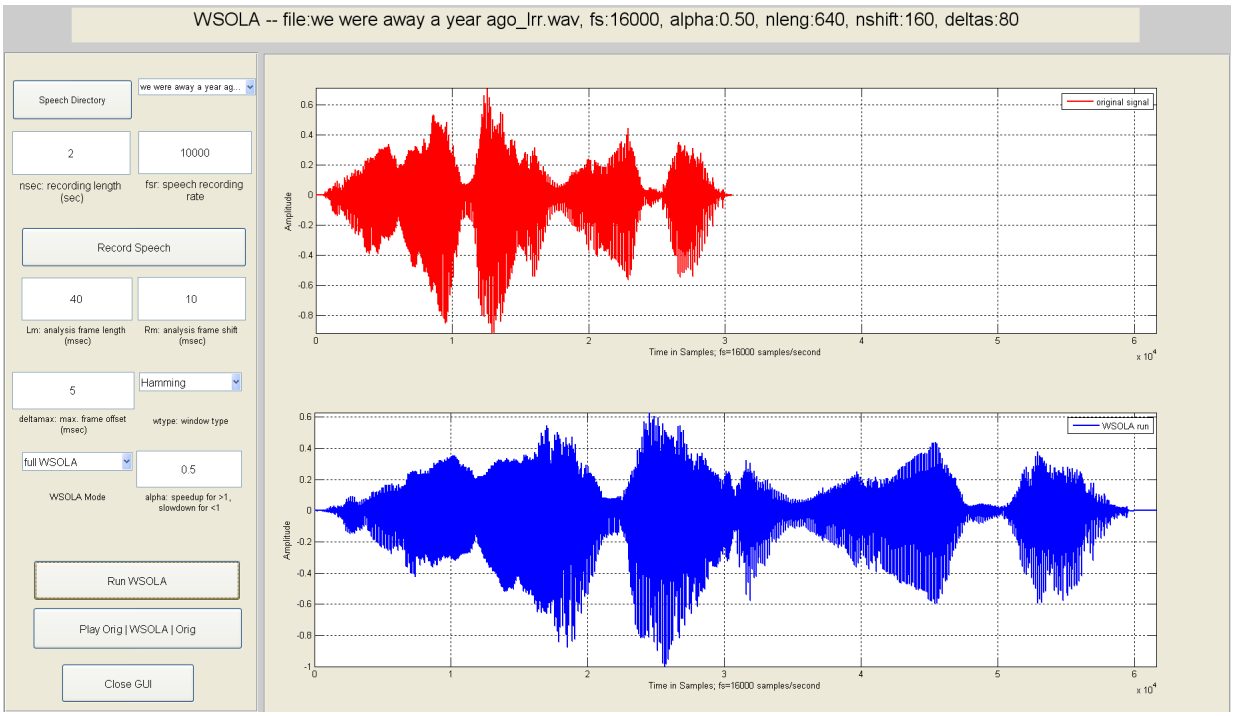

Figure 3: An example of the waveforms resulting from normal WSOLA operation with a slow-down factor of `alpha=0.5` . The upper graphics panel shows the waveform of the original signal and the lower graphics panel shows the waveform of the signal slowed-down by a factor of 2; i.e., `alpha=0.5`.

- at what slow-down and speed-up rates does the speech begin to distort enough that you can reliably tell the quality is diminished?

2. change the parameter `deltamax` from 5 to 2.5 and then to 7.5. Rerun the exercise with `alpha=0.5` with both new values of `deltamax`. Can you hear the difference in quality between runs with the different values of `deltamax`? What accounts for these changes in quality?

3. set the 'WSOLA Mode' button to "frame-by-frame without pause", and set `alpha` to the value 1 (no change in speed); hit the 'Run WSOLA' button to run the speed-up/slow-down algorithm in this frame display mode, and display the ideal and aligned frames for each frame of the utterance in the upper graphics panel. What can you say about the alignments for this degenerate value of `alpha`?

4. now change `alpha` to the value 0.5 and repeat the run; what can you say about the three waveforms that are displayed, namely `xideal`, overlayed with the optimally shifted `xreal`, as shown in the top panel, and the resulting correlation of `xideal` and `xreal` (as shown in the bottom panel)?

5. now change the 'WSOLA Mode' button to "frame-by-frame with pause" and hit the 'Run WSOLA' button; the waveforms are plotted in the two graphics panels, but instead of automatically proceeding to the next frame, the program waits for a carriage return to advance to the next frame; in this manner you can see how the ideal and real frames best align, within the region of $\pm$`deltamax` from the origin of each new frame.

6. load the audio file, 'Maple_short.wav' and repeat the speed-up, slow-down experiments with this short audio file; over what range of speed-up/slow-down does WSOLA do a good job on this audio file? why is the range any different from the speech file used in earlier experiments?