ECE 417: Multimedia Signal Processing
Fall 2018

Machine Problem #1

Due: Tuesday, September 25, 2018

---

1 Overview

In this machine problem, you will synthesize speech signals using a method based on the paper "Multiband Excitation Vocoder" by Griffin and Lim [1]. You will apply short time analysis of the speech signal, estimate the pitch, spectral envelope parameters and make voiced/unvoiced decisions for each frequency band. Then, you will synthesize the voiced and unvoiced parts of the speech signal using the parameters that you estimated.

2 Procedure

1. **Analysis**: Read your input audio file. Extract frames of duration T_{frame} seconds at every T_{skip} seconds. Use Hamming window to scale the frames. Compute FFT of windowed frames. Get a rough integer estimate for the pitch period of each frame based on the autocorrelation method described by Eqs. 12 and 13 in the paper.

   Write a function that computes the estimate of spectral envelope parameters A_m using Eq. 8 of the paper for each frequency band for a given pitch period P. The frequency bands are described by the intervals $[(m-1/2)\omega_0, (m+1/2)\omega_0)$ where $\omega_0 = 2\pi/P$ and $m = 1, 2, \ldots, P-1$. For each frequency band, try to fit a voiced and unvoiced excitation and pick the one leading to the smallest error which is defined in Eq. 5 of the paper. You can compute the error in a given band by $\varepsilon_m = \frac{1}{2\pi}\int_{a_m}^{b_m} |S_w(\omega) - A_m E_w(\omega)|^2 d\omega$. In the voiced case, excitation E_w will be the FFT of the Hamming window centered around the harmonic $m\omega_0$. In the unvoiced case, take E_w to be constant and 1 over the frequency band (idealized white noise). At the end, your function should return A_m and voiced/unvoiced decisions for each frequency band and also the total error given in Eq. 9.

   Then refine your pitch estimate: For each frame, use your function described above to compute the error for different values of P which are centered around your rough estimate, e.g. $P \pm 2$ with a step size of 0.2 samples. Once you compute the final value of the pitch P, get your final estimates for A_m and voiced/unvoiced decisions.

2. **Synthesis**: In this part of the problem, you will use the estimated parameters from the analysis step to reconstruct a speech signal.

   To synthesize the voiced part of the speech, you will use Eq. 15 in the paper. You will linearly interpolate A_m's between frames and compute the phase of the sinusoidal term using the interpolation formulas presented in Lecture 6. Note that the A_m should be taken as zero for the unvoiced frequency bands. You can initialize the phase with zeros for the first frame.

   To synthesize the unvoiced part of the speech, for each of the unvoiced frequency bands compute the energy in the band and get the variance estimate of the transform coefficients for Gaussian noise. Generate a complex Gaussian random variable using this variance. These random numbers will correspond to the FFT of the noise. Then take inverse FFT to get the time domain noise samples and then use linear interpolation to achieve overlap-add.

   To get the final synthetic speech add the voiced and unvoiced parts. Your signal will be complex, take its real part.

3 Experiments

1. Implement the analysis and synthesis algorithms described above.
2. Run your functions with the input file 's5.wav' for $T_{frame} = 25$ms and T_{skip}=10ms, use 1024-point FFT, use the interval [20, 90] for the initial estimate of the pitch. Plot the final pitch estimates versus the frame index. Also plot the error per frame. Comment on the quality of the synthetic speech, e.g. comment on intelligibility and buzziness.
3. Spectrogram is a time frequency representation of a signal which shows the amplitude or the energy of the frequency components at each time frame. Plot the spectrograms of the original and the synthesized signal and compare them. You can use '*spectrogram*' (MATLAB) or '*scipy.signal.spectrogram*' (Python) functions with the appropriate parameters to generate the plots.
4. Now consider the case where we make a single voiced/unvoiced decision for each frame. For each frame, if most of the harmonics are declared as voiced, assume that all harmonics in that frame are voiced, otherwise unvoiced. Synthesize the speech using this single bit voiced/unvoiced decision per frame. Compare the resulting signal with the one generated in Step 2.
5. Add Gaussian noise with variance σ^2 to the signal 's5.wav' and repeat steps 2-4. Try $\sigma = 0.01, 0.05, 0.1$. Compare the outputs for different noise levels.
6. Record your voice saying 'I am working on multiband excitation synthesis.' at 16kHz. Repeat steps 2-4.

4 Notes

- You can use MATLAB or Python for your experiments.
- You can record your voice using Wavesurfer or Praat.
- If the input speech file has two channels, use the the first channel as input to your algorithm.
- In the rough estimate of pitch, you should adjust the range of the pitch (in samples) based on the sampling frequency of the signal.
- You can make your input signal zero-mean before applying the algorithm and then downscale it by its total energy. Please do not forget to multiply the signal with the scaling factor and then add the mean back after synthesis.

5 Submission

You will submit (1) a report in PDF format, and (2) a zip file containing your code along with a Readme file to Compass. You must name your report as <Lastname>_<Firstname>_report.pdf and your zip file as <Lastname>_<Firstname>_code.zip

If you are working as a team, only one person should upload the report. Please make sure that the title page includes the names of all the team members.

References

[1] D. W. Griffin and J. S. Lim, "Multiband excitation vocoder," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 36, no. 8, pp. 1223–1235, 1988.