Segmenting sounds: simpler version

LSS 31 3 2019.

This work is based on work I did a long time ago, but updated a bit since then. Original references are [Sound Segmentation Using Onsets and Offsets](http://www.cs.stir.ac.uk/~lss/Papers/JNMR1994.pdf), Smith L.S.

Journal of New Music Research, 23, 1, 11-23, March 1994.

Interestingly, re-reading my old papers (yes, one forgets what one write after more than 20 years) I see that in a later paper, for NIPS 1995 (*Onset based sound segmentation*, pp 729--735, in Touretzky D.S., Mozer M.C., Hasselmo M.E. (eds) Advances in Neural Information Processing Systems 8 (Proceedings of the 1995 Conference), MIT Press, 1996), I added a layer of integrate-and-fire neurons. I haven’t taken that approach here, though it might be a good idea, because it would integrate the signals across adjacent channels (where “adjacent” means whatever the topology of the network implies). But I digress: the aim of this document is to explain what this system does.

The overall concept here is:

1. Bandpass the incoming sound signal into a number of bands, using the gammatone filter-bank. One can adjust the number of bands, the lowest and highest frequencies of the centres of each band, and the bandwidth of the bands. These are set in *findsegments\_all*, with default values, over-ridable using the varargin mechanism. See 'mincochfreq', 'maxcochfreq', 'n\_erbs'.
2. Apply the onset-offset filter to each band. This is achieved by
   1. Rectifying the signal
   2. Smoothing the rectified signal (convolution)
   3. Applying the onset-offset filter (convolution)

See lines 101-103 of findsegments\_1.m. Note that the rectified signals are summed prior to smoothing or the application of the onset/offset filter. If one was going to use onset neurons, one would not do this, but would smooth and then apply the onset offset filter to each band. This would give a set of onset/offset signals, one for each band. (I must have done this in the 1990’s but I’ve no idea where the code is).

1. Use the onset/offset signal generated to do the segmentation. Currently, this is done by
   1. Finding the onset peaks (peaks of the positive signal) and the offset peaks (peaks of -1 \* onsetoffset signal).
   2. Use these to segment the signal in a slightly ad-hoc way.

This all works reasonably well. I enclose an extra function, run\_multiplesegmentations.m which runs this software for a variable number of filter bands. See the comments inside this file as well.

In addition, in the directory stimuli\_2019, there are a few files with speech and noise. The system works well for narrowband noise (here, a 440Hz sine), but less well for wideband noise (here, white noise). You should try out the system with these.

2 April 2019

New function findsegments\_2.m started.

This

1. Calculates the onset/offset signal for each band
2. Computes the onset signal and the offset signal from this (max(0, abs()), and max(0, abs(-))
3. Applies a convergence to these multiple channel signals
4. Uses the converged input to an LIF neuron one in each channel (one set for onsets, and one set for offsets).

These still need to be processed to compute segments. (And then assessed and compared with the earlier function).

New parameters and default values (alterable using varargin)

% new parameters for LIF based onset and offset

onset\_diss = 50 ; % dissipation for onset neurons

onset\_rp = 0.05 ; % refractory period for onset cells

onset\_wt = 10.0 ; % onset weight

offset\_diss = 50 ; % dissipation for offset neurons

offset\_rp = 0.05 ; % refractory period for onset cells

offset\_wt = 10.0 ; % offset weight

convergence = 2 ; % convegence (no of inputs to each neuron = 2\*convergence + 1)