Cepstral Liftering

Cepstral Features

A cepstrogram is similar in many ways to a spectrogram. The only difference is that each 'column' in the display is a cepstral slice and not a spectral slice as in a spectrogram.

The cepstrum is obtained by inverse DFT of the log spectral magnitude. Thus, the units in which a cepstrum is displayed is the same as time. In order to avoid confusion and stress the derivation from the spectrum, we call it the *Quefrency* domain, where we will use our usual **msec** as units.

The cepstrum has something magic about it wrt. speech. It provides an almost natural separation of *source* and *filter* information. The lowest order cepstral coefficients contain pretty much all information wrt. the spectral envelope (the filter in the source-filter model) and during segments of voiced speech the cepstrogram we may get a clean observation of the pitch in a different range of cepstral coefficients.

This pitch peak in the cepstrogram corresponds to the pitch period, thus will be in the range [2.5 , 16] msec roughly. Spectral envelope can be limited to the range below 2msec.

Cepstral Liftering

The above understanding lets us split the cepstrum in an *envelope* part and in a *pitch* part, by selecting cepstral coefficients below respectively above a certain cutoff. Retaining the low order quefrency components only and then reconstructing a spectrum on the basis of this is often called **cepstral liftering** (cepstral equivalent of low-pass filtering). A complementary operation exists in selecting he high order components and reconstruct them to a *pitch spectrogram*; with common liftering it will mainly contain pitch information (alternative this could be called the *residue spectrogram*, i.e. what is remaining after the envelope has been removed).

Waveform → Fourier Spectrum → Cepstrum

Cepstrum → selected the coefficients below liftering cutoff → Spectral Envelope

Cepstrum → selected the coefficients above liftering cutoff → Spectral Residue

Time domain feature extraction

1. Energy

Energy (E^2) is defined as the average per sample energy in a short window. It is computed as the total energy in a frame (no windowing is applied) and which is then normalized for the number of samples in the frame, thus:

$E^2 = \frac{1}{N} \sum_i x^2[i]$

2. Pitch

For Pitch estimation we use the YIN algorithm as implemented in librosa. It estimates the pitch period as the minimum of the difference function:

$d(\tau) = \sum_t (x(t) - x(t - \tau))$

$T = argmin_\tau d(\tau)$

The estimated pitch period is expressed in *(m)sec* and the pitch frequency (in *Hz*) is obtained by inversion of the pitch period.

$f_0 = \frac{1}{T}$

The above algorithm is simple and naive at the same time as human speech is never perfectly periodic. Moreover non-periodic segments (unvoiced speech, silence, ...) should be recognized as such. Therefore a few additional heuristics are required to turn this baseline algorithm into an excellent state-of-the-art pitch estimator.