# AuditoryProc

<table>
<thead>
<tr>
<th>Reference info for type</th>
<th>AuditoryProc:</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://grey.colorado.edu/gendoc/emergent/AuditoryProc.html">Wiki</a></td>
<td></td>
</tr>
</tbody>
</table>

See the demo project, demos/sound_object for an example of this code at work.

Also see the VocalTractObj for a speech generation system based on the engine used in the gnuspeech toolkit.

The AuditoryProc object processes raw sound inputs in a manner consistent with the peripheral auditory pathways, and widely used in speech perception research. There are several stages of processing:

- First, a Discrete Fourier Transform (DFT) implemented with the Fast Fourier Transform (FFT) algorithm, that transforms the time-varying sound signal into a spectrogram where different frequencies are represented using different input features. Typically the sound is sampled every 10 msec or so, with a window of 25 msec, over a period of e.g., 100 msec, producing a 2D time-frequency transform of the sound.

- Then a Mel-frequency feature bank aggregation of different frequencies using triangle-shaped filters with spacing that reflects the perceptual discriminability of different frequencies in humans.

- The last stage can be either:
  - A set of time-frequency tuned gabor filters based on those observed in the auditory pathways, that respond selectively to different trajectories of frequency power over time (e.g., rising vs. falling tones).
  - Or the Mel-cepstrum discrete cosine transform that is widely used in speech processing.
Article Sources and Contributors

AuditoryProc  Source: https://grey.colorado.edu/emergent/index.php?id=12287  Contributors: Orilly