ELEG 467/667 - Image and Audio Signal Processing

Spring 2012
Nyquist sampling requires that the sampling rate (samples per second) must be at least twice the bandwidth (cycles per second) of the analog input signal.

If the analog input signal is not already guaranteed to be bandlimited to half the sampling rate, a suitable analog bandlimiting filter must be used prior to the analog-to-digital converter.

Digital sampling rate conversion must also obey the sampling theorem requirements even if no analog signal is involved.
For example, a digital data stream corresponding to a 48 kHz sampling rate must be carefully bandlimited to no more than 22.05 kHz before it is downsampled to a 44.1 kHz rate.

Aliasing produces a noticeable high frequency inharmonic whistling.

Examples:
1. Original piano
2. Piano with 20KHz bandwidth sampled at 8KHz without antialiasing filter
3. Piano with 20KHz bandwidth sampled at 10KHz without antialiasing filter
Spectral Effects of Sampling

Consider an analog signal $x(t)$ with Fourier magnitude spectrum $|X(f)|$ depicted here:

- This example spectrum contains energy from zero to above some specified frequency, $f_a$ Hz.
- When an analog signal is sampled, the effect on the spectrum is to replicate scaled versions of the original baseband spectrum at all integer multiples of the sampling rate.
The spectral images located at each integer multiple of the sampling rate overlap in the frequency regions near $\pm f_a, \pm 2f_a, \pm 3f_a, \ldots$, which correspond to half of the sampling rate (i.e., $f_s = 2f_a$, $f_s/2 = f_a$) and integer multiples.

There is no convenient way, in general, to separate the overlapping frequency regions once the sampling process is complete.
Spectral Effects of Sampling

- The solution to the spectral overlap problem in digital sampling is to ensure that the input signal is bandlimited so that the spectral images do not overlap.
Quantization

- The continuous-to-discrete time sampling is ideally a lossless process, but the digitization inherently requires quantization.
- The input signal is quantized into a finite range of amplitude steps, where each step is assigned an integer index value.
Hard Limiting (clipping), Analog and Digital

- When digital audio samples are allocated a fixed word size, all sample values are constrained to lie within the range.
- If a signal processing calculation creates a result that lies outside of the valid range, numerical overflow occurs and audible distortion is likely to occur.
- The overflow value must be replaced by a value within the allowable range.
- Most audio signal processing systems perform hard clipping when overflow occurs: all out-of-range values are replaced by the nearest in-range value.
Conventional arithmetic hardware, may not provide automatic clipping, in which case it is possible that the numerical representation will wrap around to the opposite polarity.

Wrap around causes severe signal distortion due to full-scale signal swing, and should therefore be detected and corrected.
Material has been amplified by 20 dB for the following examples. Turn the volume knob down before playing any of the examples.

- Hard digital clipping
- Wrap around
- Hard ’analog’ clipping
Audible Effects of Frequency Filtering

Various digital audio processing techniques involve frequency selective filters. Filtering is used for spectral equalization, bandlimiting prior to sample rate conversion (upsampling or downsampling), and for a variety of creative audio purposes.

- Original piano
- Piano with 100 Hz Highpass filter
- Piano with 500 Hz Highpass filter
- Piano with 1000 Hz Highpass filter
- Piano with 1000 Hz Lowpass filter
- Piano with 500 Hz Lowpass filter
- Piano with 100 Hz Lowpass filter
- Piano with 440Hz -880Hz Bandpass filter
Basic Psychoacoustics of Digital Audio

- The ear is a frequency analysis instrument.
- The ear filters signals into 100Hz (or a bit smaller) bands at low frequencies, and $1/3$ to $1/4$ octave bands at higher frequencies, with a crossover between the two happening smoothly in the 500 to 1000Hz range.
The human ear responds differently when two simple signals are presented separately or together.

Under some circumstances two signals that are audible when presented individually can mask each other when presented simultaneously, rendering one of the signals inaudible.

This auditory phenomenon is exploited in perceptual audio coders such as MPEG-1 Layer 3 (MP3) so that quantization noise is masked by tonal components in the audio signal itself.
In this example, the signal is a sequence demonstrating the audibility of noise in a narrow bandwidth (somewhat narrower than a critical band) around a sine wave. The sine wave is at 4kHz. The noise is $\pm 400$ Hz, with uniform amplitude and random phase. The level of the noise energy (total) relative to the sine wave is (in dB) as follows: -40 -35 -30 -25 -20 -15 -10 -5 0 -5 -10 -15 -20 -25 -30 -35 -40.
Amplitude envelope of tone masking noise sequence
Noise masking tone

The next example demonstrates the audibility of a tone masked by a sine wave. The tone is again at 4000Hz, and the noise uniformly distributed within a +-400Hz bandwidth around it. In this case, the level of the noise is constant, and the level of the tone is varied (in dB) relative to the noise as follows: -15 -10 -5 -2.5 0 2.5 5 10 5 2.5 0 -2.5 -5 -10 -15.
Amplitude envelope of noise masking tone sequence
Fourier Transform

\[ f \]

\[ |\hat{f}| \]
Fourier Transform

Signal \( f : \mathbb{R} \to \mathbb{R} \)

Fourier representation \( f(t) = \int_{\omega \in \mathbb{R}} c_\omega e^{2\pi i \omega t} d\omega \), \( c_\omega = \hat{f}(\omega) \)

Fourier transform \( \hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t)e^{-2\pi i \omega t} dt \)
Fourier Transform

Signal \[ f : \mathbb{R} \rightarrow \mathbb{R} \]

Fourier representation \[ f(t) = \int_{\omega \in \mathbb{R}} c_\omega e^{2\pi i \omega t} d\omega, \quad c_\omega = \hat{f}(\omega) \]

Fourier transform \[ \hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) e^{-2\pi i \omega t} dt \]

- Tells \textbf{which} notes (frequencies) are played, but does not \textbf{tell when} the notes are played
- Frequency information is averaged over the entire time interval
- Time information is hidden in the phase
Fourier Transform

\[ f \]

Time (seconds)

\[ |\hat{f}| \]

Frequency (Hz)
Short Time Fourier Transform

Idea (Dennis Gabor, 1946):

- Consider only a small section of the signal for the spectral analysis
  -> recovery of time information

- Short Time Fourier Transform (STFT)

- Section is determined by pointwise multiplication of the signal with a localizing window function
Short Time Fourier Transform

\[ f(t) \]

Time (seconds)

\[ |\hat{f}| \]

Frequency (Hz)
Short Time Fourier Transform

\[ f \]

\[ |\hat{f}| \]

Time (seconds)

Frequency (Hz)
Short Time Fourier Transform

\[ f \]

Time (seconds)

\[ |\hat{f}| \]
Frequency (Hz)
Short Time Fourier Transform

\[ f(t) \]

\[ |\hat{f}(\omega)| \]

\[ 0 \quad 10 \quad 20 \quad 30 \quad 40 \quad 50 \quad 60 \]

Frequency (Hz)

\[ 0 \quad 0.05 \quad 0.1 \quad 0.15 \]

\[ 0 \quad 0.5 \quad 1 \quad 1.5 \]

Time (seconds)
Short Time Fourier Transform

$f$

Time (seconds)

$|\hat{f}|$

Frequency (Hz)
Short Time Fourier Transform

\[ f \]

\[ |\hat{f}| \]
Short Time Fourier Transform

\[ f \]

Time (seconds)

\[ |\hat{f}| \]

Frequency (Hz)
Short Time Fourier Transform

Definition

- **Signal** \( f : \mathbb{R} \to \mathbb{R} \)

- **Window function** \( g : \mathbb{R} \to \mathbb{R} \quad (g \in L^2(\mathbb{R}), \|g\| = 1) \)

- **STFT** \( \tilde{f}(\omega, t) := \int_{\mathbb{R}} f(u) \overline{g}(u - t) e^{-2\pi i \omega u} \, du = \langle f | g_{\omega, t} \rangle \)

with \( g_{\omega, t}(u) := e^{2\pi i \omega u} g(u - t), \quad u \in \mathbb{R} \)
Short Time Fourier Transform

Intuition:

- $g_{\omega,t}$ is "musical note" of frequency $\omega$, which oscillates within the translated window $u \rightarrow g(u - t)$

![Illustration of oscillating waveforms](image)

- Inner product $\langle f | g_{\omega,t} \rangle$ measures the correlation between the musical note $g_{\omega,t}$ and the signal $f$. 
Window Function

Box window

\[ g \]

\[ |\hat{g}| \]
Window Function

Triangle window

\[ g \]

\[ |\hat{g}| \]
Window Function

Hann window

\[ g \]

\[ |\hat{g}| \]
Window Function

Trade off between smoothing and „ringing“
Time-Frequency Representation

Frequency \( \omega \) (Hertz)

Intensity (dB)

Time \( t \) (seconds)
Time-Frequency Representation

Chirp signal and STFT with box window of length 0.05
Time-Frequency Representation

Chirp signal and STFT with Hann window of length 0.05
Time-Frequency Localization

- Size of window constitutes a trade-off between time resolution and frequency resolution:
  - Large window: poor time resolution, good frequency resolution
  - Small window: good time resolution, poor frequency resolution

- Heisenberg Uncertainty Principle: there is no window function that localizes in time and frequency with arbitrary position.
Short Time Fourier Transform

Signal and STFT with Hann window of length 0.02
Short Time Fourier Transform

Signal and STFT with Hann window of length 0.1
Heisenberg Uncertainty Principle

Window function \( g \in L^2(\mathbb{R}) \) with \( \|g\| = 1 \)

Center

\[
t_0 = t_0(g) := \int_{-\infty}^{\infty} t |g(t)|^2 dt
\]

\[
\omega_0 = \omega_0(g) := \int_{-\infty}^{\infty} \omega |\hat{g}(\omega)|^2 d\omega
\]

Width

\[
T(g) := \left( \int_{-\infty}^{\infty} (t - t_0)^2 |g(t)|^2 dt \right)^{\frac{1}{2}}
\]

\[
\Omega(g) := \left( \int_{-\infty}^{\infty} (\omega - \omega_0)^2 |\hat{g}(\omega)|^2 d\omega \right)^{\frac{1}{2}}
\]

\[
T(g) \cdot \Omega(g) \geq \frac{1}{4\pi}
\]
Information Cells

\[ g_{\omega,t}(u) := e^{2\pi i \omega u} g(u - t) \]  

"musical note"
MATLAB

- MATLAB function SPECTROGRAM
- \( N \) = window length (in samples)
- \( M \) = overlap (usually \( N/2 \))
- Compute DFT\(_N\) for every windowed section
- Keep lower \( N/2 \) Fourier coefficients

→ Sequence of spectral vectors
  (for each window a vector of dimension \( N/2 \))
Example

Let \( x \) be a discrete time signal \( x(n) = f(Tn) \)

Sampling rate: \( 1/T = 22050 \text{ Hz} \)
Window length: \( N = 4096 \)
Overlap: \( N/2 = 2048 \)
Hopsize: window length – overlap

Let \( v_0 := (x(0), x(1), \ldots, x(4095)) \)
\( v_1 := (x(2048), \ldots, x(6143)) \)
\( v_2 := (x(4096), \ldots, x(8191)) \)
\[ \vdots \]
\( v_m \) corresponds to window \([m \cdot 2048 : m \cdot 2048 + 4095]\)
Example

Time resolution:

\[
\frac{\text{hopsize}}{\text{sampling rate}} = \frac{4096 - 2048}{22050} = 0.093 = 93 \text{ ms}
\]

Frequency resolution:

\[v = v_0, \quad \hat{v} := \text{DFT}_N(v)\]

\[
\hat{v}(k) \approx \frac{1}{T} \cdot \hat{f} \left( \frac{k}{N} \cdot \frac{1}{T} \right)
\]

\[
\omega = \frac{k}{N} \cdot \frac{1}{T} = k \cdot \frac{22050}{4096} = k \cdot 5.38 \text{ Hz}
\]
Pitch Features

Model assumption: Equal-tempered scale
- MIDI pitches: $p \in [1 : 128]
- Piano notes: $p = 21$ (A0) to $p = 108$ (C8)
- Concert pitch: $p = 69$ (A4)
- Center frequency: $f_{\text{MIDI}}(p) = 2^{\frac{p-69}{12}} \cdot 440$ Hz

→ Logarithmic frequency distribution
   Octave: doubling of frequency
Pitch Features

Idea: Binning of Fourier coefficients

Divide up the frequency axis into logarithmically spaced "pitch regions" and combine spectral coefficients of each region to a single pitch coefficient.
Pitch Features

Time-frequency representation

Windowing in the frequency domain

Windowing in the time domain
Pitch Features

Details:

- Let \( \hat{v} \) be a spectral vector obtained from a spectrogram w.r.t. a sampling rate \( 1/T \) and a window length \( N \). The spectral coefficient \( \hat{v}(k) \) corresponds to the frequency

\[
f_{\text{coeff}}(k) := \frac{k}{N} \cdot \frac{1}{T}
\]

- Let

\[
S(p) := \{k : f_{\text{MIDI}}(p - 0.5) \leq f_{\text{coeff}}(k) < f_{\text{MIDI}}(p + 0.5)\}
\]

be the set of coefficients assigned to a pitch \( p \in [1:128] \).

Then the pitch coefficient \( P(p) \) is defined as

\[
P(p) := \sum_{k \in S(p)} |\hat{v}(k)|^2
\]
Pitch Features

Example: A4, $p = 69$

- Center frequency: $f(p = 69) = 2^{\frac{0}{12}} \cdot 440 = 440\, Hz$
- Lower bound: $f(p = 68.5) = 2^{\frac{-0.5}{12}} \cdot 440 = 427.5\, Hz$
- Upper bound: $f(p = 69.5) = 2^{\frac{0.5}{12}} \cdot 440 = 452.9\, Hz$
- STFT with $N = 4096, 1/T = 22050$

\[
\begin{align*}
  f(k = 79) &= 425.3\, Hz \\
  f(k = 80) &= 430.7\, Hz \\
  f(k = 81) &= 436.0\, Hz \\
  f(k = 82) &= 441.4\, Hz \\
  f(k = 83) &= 446.8\, Hz \\
  f(k = 84) &= 452.2\, Hz \\
  f(k = 85) &= 457.6\, Hz \\
\end{align*}
\]
Pitch Features

Example: A4, \( p = 69 \)

- Center frequency: \( f(p = 69) = 2^{\frac{0}{12}} \cdot 440 = 440 \text{ Hz} \)
- Lower bound: \( f(p = 68.5) = 2^{\frac{-0.5}{12}} \cdot 440 = 427.5 \text{ Hz} \)
- Upper bound: \( f(p = 69.5) = 2^{\frac{0.5}{12}} \cdot 440 = 452.9 \text{ Hz} \)
- STFT with \( N = 4096, 1/T = 22050 \)

\[
\begin{array}{l}
  f(k = 79) = 425.3 \text{ Hz} \\
  f(k = 80) = 430.7 \text{ Hz} \\
  f(k = 81) = 436.0 \text{ Hz} \\
  f(k = 82) = 441.4 \text{ Hz} \\
  f(k = 83) = 446.8 \text{ Hz} \\
  f(k = 84) = 452.2 \text{ Hz} \\
  f(k = 85) = 457.6 \text{ Hz} \\
\end{array}
\]

\[
\left\{ \begin{array}{l}
  S(p = 69) \\
  P(p = 69) = \sum_{k=80}^{84} |\hat{v}(k)|^2
\end{array} \right.
\]
## Pitch Features

<table>
<thead>
<tr>
<th>Note</th>
<th>MIDI pitch</th>
<th>Center [Hz] frequency</th>
<th>Left [Hz] boundary</th>
<th>Right [Hz] boundary</th>
<th>Width [Hz]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A3</td>
<td>57</td>
<td>220.0</td>
<td>213.7</td>
<td>226.4</td>
<td>12.7</td>
</tr>
<tr>
<td>A#3</td>
<td>58</td>
<td>233.1</td>
<td>226.4</td>
<td>239.9</td>
<td>13.5</td>
</tr>
<tr>
<td>B3</td>
<td>59</td>
<td>246.9</td>
<td>239.9</td>
<td>254.2</td>
<td>14.3</td>
</tr>
<tr>
<td>C4</td>
<td>60</td>
<td>261.6</td>
<td>254.2</td>
<td>269.3</td>
<td>15.1</td>
</tr>
<tr>
<td>C#4</td>
<td>61</td>
<td>277.2</td>
<td>269.3</td>
<td>285.3</td>
<td>16.0</td>
</tr>
<tr>
<td>D4</td>
<td>62</td>
<td>293.7</td>
<td>285.3</td>
<td>302.3</td>
<td>17.0</td>
</tr>
<tr>
<td>D#4</td>
<td>63</td>
<td>311.1</td>
<td>302.3</td>
<td>320.2</td>
<td>18.0</td>
</tr>
<tr>
<td>E4</td>
<td>64</td>
<td>329.6</td>
<td>320.2</td>
<td>339.3</td>
<td>19.0</td>
</tr>
<tr>
<td>F4</td>
<td>65</td>
<td>349.2</td>
<td>339.3</td>
<td>359.5</td>
<td>20.2</td>
</tr>
<tr>
<td>F#4</td>
<td>66</td>
<td>370.0</td>
<td>359.5</td>
<td>380.8</td>
<td>21.4</td>
</tr>
<tr>
<td>G4</td>
<td>67</td>
<td>392.0</td>
<td>380.8</td>
<td>403.5</td>
<td>22.6</td>
</tr>
<tr>
<td>G#4</td>
<td>68</td>
<td>415.3</td>
<td>403.5</td>
<td>427.5</td>
<td>24.0</td>
</tr>
<tr>
<td>A4</td>
<td>69</td>
<td>440.0</td>
<td>427.5</td>
<td>452.9</td>
<td>25.4</td>
</tr>
</tbody>
</table>
Pitch Features

Note:

- \( P \in \mathbb{R}^{128} \)
- For some pitches, \( S(p) \) may be empty. This particularly holds for low notes corresponding to narrow frequency bands.

→ Linear frequency sampling is problematic!

Solution:
Multi-resolution spectrograms or multirate filterbanks
Pitch Features

Example: Friedrich Burgmüller, Op. 100, No. 2
Pitch Features

Spectrogram
Pitch Features

Spectrogram

C8: 4186 Hz
C7: 2093 Hz
C6: 1046 Hz
C5: 523 Hz
C4: 261 Hz
Pitch Features

Pitch representation

C8
C7
C6
C5
C4
Pitch Features

Example: Chromatic Scale

Spectrogram
Pitch Features

Example: Chromatic Scale

Pitch representation
Chroma Features

- Human perception of pitch is periodic in the sense that two pitches are perceived as similar in color if they differ by an octave.

- Separation of pitch into two components: tone height (octave number) and chroma.

- Chroma: 12 traditional pitch classes of the equal-tempered scale. For example:
  Chroma C \( \cong \{ \ldots, C0, C1, C2, C3, \ldots \} \)

- Computation: pitch features \( \rightarrow \) chroma features
  Add up all pitches belonging to the same class

- Result: 12-dimensional chroma vector.
Chroma Features

Chromatic circle

Shepard's helix of pitch perception

http://en.wikipedia.org/wiki/Pitch_class_space

Bartsch/Wakefield, IEEE Trans. Multimedia, 2005
Chroma Features

- Sequence of chroma vectors correlates to the harmonic progression

- Normalization $v \rightarrow \frac{v}{\|v\|}$ makes features invariant to changes in dynamics

- Further quantization and smoothing: CENS features

- Taking logarithm before adding up pitch coefficients accounts for logarithmic sensation of intensity
Chroma Features

Example: C-Major Scale
Chroma Features

Pitch representation

C8
C7
C6
C5
C4
Chroma Features

Chroma representation
Chroma Features

Chroma representation (normalized)
Chroma Features

Example: Bach Toccata

Feature resolution: 1 Hz
Chroma Features

Example: Bach Toccata

Koopman  Ruebsam

Feature resolution: 0.33 Hz
Chroma Features

Example: Zager & Evans “In The Year 2525”

How to deal with transpositions?
Chroma Features

Example: Zager & Evans “In The Year 2525”

Original: \((v^1, \ldots, v^N)\)

Shifted: \((\sigma(v^1), \ldots, \sigma(v^N))\)