Chroma and tonality

Juan Pablo Bello
MPATE-GE 2623 Music Information Retrieval
New York University
Tonality

• Most western music is based on the tonality system.

• Tonality: arranges sounds according to pitch relationships into inter-dependent spatial and temporal structures.

• Characterizing chords, keys, melody, motifs and even form, largely depends on understanding these structures.

• Harmony: vertical (synchronous) pitch structures

• Melody: horizontal (sequential) pitch structures
Pitch perception

• The pitch helix is a representation of pitch relationships that places tones in the surface of a cylinder (Shepard, 2001)

• Models the special relationship that exists between octave intervals.

• The model is a function of 2-dimensions:

  • Height: naturally organizes pitches from low to high

  • Chroma: represents the inherent circularity of pitch organization
Chroma

- Chroma describes the angle of pitch rotation as it traverses the helix
- Two octave-related pitches will share the same angle in the chroma circle: a relation that is not captured by a linear pitch scale (or even Mel).

- For the analysis of western tonal music we quantize this angle into 12 positions or pitch classes.
Independence of chroma from height

• Shepard tones: mix of sinusoids with octave-separated frequencies, and a bell-shaped spectral shape

• Scales of these tones create the illusion of constantly rising/falling
Chroma features

• aka Pitch Class Profiles (PCP): distribution of the signal’s energy across a predefined set of pitch classes (chroma).

• Popular feature in music DSP: introduced by Fujishima (ICMC, 99) and Wakefield (SPIE, 99). Extensively used for chord, key recognition, segmentation, synchronization, fingerprinting, etc.

• Many strategies for its computation: Log-frequency filterbanks in the time and frequency-domain, CQ-transform, SMS, phase vocoder.

![Diagram](audio -> DFT -> Log-freq filterbank -> Folding -> PCP)
Chroma features

- Center frequencies linear in log$_2$ scale

\[ f_c(k_{lf}) = f_{\text{min}} \times 2^{\frac{k_{lf}}{\beta}} \]

- $f_{\text{min}} =$ minimum frequency of the analysis (Hz)
- $k_{lf} =$ integer filter index $\in [0, (\beta \times Z) - 1]$
- $\beta =$ bins per octave
- $Z =$ number of octaves
Chroma features

- Filterbank of overlapping windows

- Center frequency of one window: starting point of next window and end point of previous window.

- All windows are normalized to unity sum.
Chroma features

![Chroma features diagram]

- Chroma features
- Spectrogram (dB)
- CQ filterbank
- CQ Spec. (dB)
- Chroma (dB)

Time (s)
Chroma features

• The chroma is computed by summing the log-frequency magnitude spectrum across octaves

\[ C_f(b) = \sum_{z=0}^{Z-1} |X_{lf}(b + z\beta)| \]

- \( X_{lf} = \) log-frequency spectrum
- \( z = \) integer octave index \( \in [0, Z - 1] \)
- \( Z = \) number of octaves
- \( b = \) integer pitch class (chroma) index \( \in [0, \beta - 1] \)
- \( \beta = \) bins per octave

• The resulting sequence of chroma vectors is known as chromagram
Chroma limitations
Chroma limitations
Chroma limitations
Improving Chroma

- Filterbank: each harmonic contributes to $f_0$ with a weight $s^{i-1}$, $s < 1$

Gomez (2006)
Improving Chroma

- Chroma Energy Normalized Statistics (CENS, Müller 2007)
Improving Chroma

- Chroma DCT-Reduced log Pitch (CRP, Müller and Ewert 2010)
Improving Chroma

- Beat synchronous (Bartsch and Wakefield, 2001)
Key detection

• Subjective ratings of fit for tones within a key context (Krumhansl and Kessler, 1982)
Key detection

- Gomez’s key finding algorithm (2006)

chroma features

Key Templates → Similarity → Max → key

Average
Key detection

- Templates: combine tonic, sub-dominant and dominant triads per key + harmonic info

- Similarity: Correlation between average chroma and template
Key detection

- Templates: combine tonic, sub-dominant and dominant triads per key + harmonic info

- Similarity: Correlation between average chroma and template
Chord recognition

- Template matching approach pioneered by Fujishima (ICMC, 1999), and used by, e.g., Harte and Sandler (AES, 05), Oudre et al (TSALP, 2011).
Chord recognition

- Simple binary templates
Chord recognition: distance/fitness

- Measures how well the templates fit each frame of the chromagram (we select, e.g. the template that maximizes the dot product at each frame)
Chord recognition

No filtering

Chroma

Templates

Chords

Time (s)
Chord recognition: pre-filtering

Original Features

Moving Average

Moving Median

✓ fast frame rate
★ noisy

✓ cleaner
★ blurred bound.

- in between
Chord recognition

Pre filtering only

Chroma

Templates

Chords

Time (s)
Chord recognition: pre-filtering

~20% increase

* from Cho et al (SMC, 2010)
Chord recognition: post-filtering

• The fitness matrix is also filtered before peak picking (e.g. via moving mean or median filters or, preferably, using the Viterbi algorithm)
Chord recognition

Pre + post filtering

Chroma

Templates

Chords

Time (s)
Viterbi algorithm

- Chords: hidden states of a process ($s_i$); Chroma features: observable result of that process ($y_t$). One state per observation.

- States and observations are connected by the emission probability: observing a chroma vector at frame $t$ given chord $i$

- Next chord occurrence depends only on the current chord (Markov process)

- Goal: to find the most likely sequence of chords that results on the current chromagram - > Viterbi algorithm
Viterbi algorithm

- **States** ($s_j$): a finite set of $J$ chords (e.g. 24 major/minor triads)
- **Observations** ($y_t$): chromagram
- **Initial (prior) probability** ($\pi_j$): same value for all chords
- **$P(y_t \mid s_j)$**: positive fitness/matching values, normalized to sum to unity
- **Matrix of transition probabilities between states** ($a_{ij}$)
Viterbi algorithm

Initialization:

\[ V_{1,j} = \log[(P(y_1|s_j)] + \log[\pi_j] \]

then for \( t = 2 : T \),

\[ V_{t,j} = \max_i \{ V_{t-1,i} + \log[a_{ij}] + \log[(P(y_t|s_j)] \}, \ i \in [1, J] \]

\( path(t - 1, j) = \hat{i} \), the \( i \) that maximizes the sum.

Finally,

\[ \hat{path} = path(t, \arg\max_i (V_{T,i})), \ \forall t \in [1, T] \]
Chord recognition

Pre filtering + Viterbi

Chroma

Templates

Chords

Time (s)
Chord ID: Post-filtering

- Computation of transition probabilities \( (a_{ij}) \) from: music knowledge, annotated data, random.

![Circle of fifths](image1)

![Chord bi-grams](image2)

![Uniform](image3)

- Separately adjusting the self-transition probability via a transition penalty \( P \):

\[
\log(\hat{a}) = \begin{cases} 
\log(a_{ij}) - \log(P) & \text{for } i \neq j \\
\log(a_{ij}) & \text{for } i = j 
\end{cases}
\]
Chord recognition: post-filtering

- Enforcing strong self-transitions (regardless of the rest!):

\[ \sim 25\% \text{ increase} \]

* from Cho et al (SMC, 2010)
Chord recognition: Pattern Matching

Binary Template
(Fujishima 99, Harte and Sandler 05)

Single Gaussian
(Sheh and Ellis 03, Bello and Pickens 05)

Mixtures of Gaussians
(Burgoyne et al 05, Reed et al 09)

Networks of HMMs
(Khadkevich and Omologo 09)
Chord recognition: Pattern Matching

Binary Template
(Fujishima 99, Harte and Sandler 05)

Single Gaussian
(Sheh and Ellis 03, Bello and Pickens 05)

Mixtures of Gaussians
(Burgoyne et al 05, Reed et al 09)

Networks of HMMs
(Khadkevich and Omologo 09)

Accuracy increase of no more than 5% between the simplest and the most complex model!
Tonnetz

- The Tonnetz is a pitch space defined by the network of relationships between musical pitches in just intonation (Euler, 1739)

- Close harmonic relations are modeled as short distances on an infinite Euclidian plane
Tonnetz

- Chords become geometric structures on the plane, keys are defined by regions in the harmonic network.

- Major triad
- Minor triad
- Augmented
- Diminished

- Major 3rds
- Perfect 5ths
- Minor 3rds
- Major 7th chord
Tonnetz

- Introducing Enharmonic and Octave Equivalence reduces the set of all notes to 12 pitch classes and wraps the plane into a hypertorus.

- The 6D interior space of the hypertorus can be seen as three 2D circles: of fifths, major thirds and minor thirds. Chords can be described by their 6D centroids in this space (Harte and Gasser, 2006).
Harte and Gasser’s tonal centroid of a chroma vector can be computed as:

$$\text{TC}(d) = \frac{1}{\sum_b |C_f(b)|} \sum_{b=0}^{\beta-1} \Phi(d, b) C_f(b)$$

$$\Phi = [\phi_0, \phi_1 \cdots \phi_{\beta-1}]$$

$$\phi_b = \begin{bmatrix} r_1 \sin(b \frac{7\pi}{6}) \\ r_1 \cos(b \frac{7\pi}{6}) \\ r_2 \sin(b \frac{3\pi}{2}) \\ r_2 \cos(b \frac{3\pi}{2}) \\ r_3 \sin(b \frac{2\pi}{3}) \\ r_3 \cos(b \frac{2\pi}{3}) \end{bmatrix}$$
References


References


