Investigating Style Evolution of Western Classical Music: A Computational Approach

Christof Weiß[1], Matthias Mauch[2], Simon Dixon[2], and Meinard Müller[1]

Supplementary Information

This documents provides supplementary information to the paper "Investigating Style Evolution of Western Classical Music: A Computational Approach." In particular, we provide details on the feature extraction procedures including the underlying chroma features (Section 1) as well as derived features for describing interval categories (Section 2), tonal compexity (Section 3), and chord transitions (Section 4). In Section 5, we explain our feature aggregation method and the basic ideas of PCA. Beyond that, Section 6 provides additional results on measuring diversity from the piece-wise clustering results.

S1 Chroma Features

Chroma features have turned out to be suitable mid-level representations for approaching different music processing tasks (Fujishima, 1999; Bartsch & Wakefield, 2005; Ellis & Poliner, 2007; Müller et al., 2005). They have been shown to capture the tonal content of audio signals and to be invariant against timbre variations to a certain extent (Gómez & Herrera, 2004; Müller, 2015). In the following, let $\mathbf{c} = (c_0, c_1, \ldots, c_{11})^{\mathrm{T}} \in \mathbb{R}^N$ denote a chroma vector of dimension $N := 12$, with $c_n \geq 0$ indicating the energy of the n-th pitch class. The indices $n = 0, 1, 2, \ldots, 11$ correspond to the twelve chroma values C, C♯, D, …, B. Because of the octave invariance, the features show a cyclic nature so that a transposition in pitch leads to circular shift. We compute the chroma features in a temporal resolution of 0.1 ms (10 Hz). The features are normalized columnwise with respect to the ℓ^1-norm so that $\ell^1(\mathbf{c}) = 1$.

When extracting chroma features from music recordings, a number of acoustic effects such as noisy or percussive sounds play a role. In particular, partials of a played note have an influence on the chroma distribution. This may lead to a remarkable difference between the chroma vector

[1]International Audio Laboratories Erlangen, Germany
[2]Centre for Digital Music, Queen Mary University of London, UK

Corresponding author:
Christof Weiß, International Audio Laboratories Erlangen, Am Wolfsmantel 33, 91058 Erlangen, Germany.
Email: christof.weiss@audiolabs-erlangen.de

and the pitch classes as notated in a score. Several strategies have been proposed to reduce this influence of partials and to make the chroma features more robust to timbre variations (Lee, 2006; Gómez, 2006; Müller & Ewert, 2010; Mauch & Dixon, 2010). For our experiments, we use a strategy proposed by Mauch and Dixon (2010), which reduces the influence of partials using an approximate transcription method based on the *Non-Negative Least Squares* (NNLS) algorithm. These features considerably improved chord detection results for popular music. The code was published as "Vamp" plugin.[1] We use this plugin in combination with the command line tool Sonic Annotator.[2]

S2 Interval Category Features

To describe style-relevant tonal characteristics, the measurement of harmonic intervals can be useful. A systematic way of interval-based analysis is the pitch class set theory (Hanson, 1960; Forte, 1973). In the context of this theory, a system of six *interval categories* (ICs) was developed for style analysis (Quinn, 2001; Honingh et al., 2009). In previous work (Weiß et al., 2014), we proposed features to measure the occurrence of these interval categories from audio recordings. They are invariant to musical transposition and have shown good results as basis features for classifying musical styles (Weiß et al., 2014; Weiß, 2017).

We compute the features on the basis of NNLS chroma representations with a temporal resolution of 10 Hz (see Section 1). For a given chroma vector $\mathbf{c}$, we compute the likelihood for the joint appearance of two pitch classes that relate by the respective interval. To this end, we multiply the respective entries c_q of the chroma vector $\mathbf{c}$, with $q \in \{0, 1, \ldots, 11\} := [0 : 11]$. For the feature F_5 relating to perfect fifth and perfect fourth intervals, we multiply the chroma value c_0 for pitch class C ($q = 0$) with the value c_5 for F ($q = 5$). These pitch classes form an interval of 5 semitones distance. Since we are interested in the *type* of the interval and not in the specific pitches, we equally weight all transpositions of this interval by summing over all cyclic shifts (C-F, C♯-F♯, D-G, ..., B-E). We obtain the feature value

$$F_5 := \sum_{q=0}^{11} c_q \cdot c_{(q+5) \bmod 12}. \tag{1}$$

To generalize this expression, we use a binary template $\mathbf{T} := (T_0, \ldots, T_{11})^{\mathrm{T}} \in \mathbb{R}^{12}$:

$$F_{\mathbf{T}} = \sum_{q=0}^{11} \left(\prod_{k=0}^{11} \left(c_{(q+k) \bmod 12} \right)^{T_k} \right) \tag{2}$$

By suitably choosing $\mathbf{T}$, we can estimate the different interval categories:

$$\begin{aligned}
\mathbf{T}_1 &= (1,1,0,0,0,0,0,0,0,0,0,0)^{\mathrm{T}} \\
\mathbf{T}_2 &= (1,0,1,0,0,0,0,0,0,0,0,0)^{\mathrm{T}} \\
\mathbf{T}_3 &= (1,0,0,1,0,0,0,0,0,0,0,0)^{\mathrm{T}} \\
\mathbf{T}_4 &= (1,0,0,0,1,0,0,0,0,0,0,0)^{\mathrm{T}} \\
\mathbf{T}_5 &= (1,0,0,0,0,1,0,0,0,0,0,0)^{\mathrm{T}} \\
\mathbf{T}_6 &= (1,0,0,0,0,0,1,0,0,0,0,0)^{\mathrm{T}}
\end{aligned} \tag{3}$$

Using the template $\mathbf{T}_5$, we obtain the feature value F_5 as denoted in Equation (1).

S3 Tonal Complexity Features

Beyond concrete tonal structures such as intervals, we consider a more abstract property that we refer to as *tonal complexity*. There are several approaches for estimating tonal complexity from audio recordings (Streich, 2006; Honingh & Bod, 2010; Di Giorgi et al., 2017). We use a set of features proposed in previous work (Weiß & Müller, 2014, 2015; Weiß, 2017). The features rely on statistical measures calculated from NNLS chroma features (see Section 1).

The notion of tonal complexity applies to different time scales. On a fine temporal level, tonal complexity relates to the characteristics of chords or scales. Looking at a coarser level, the presence of modulations typically leads to an increase of tonal complexity. To account for these different aspects, we compute the complexity features from chroma features in different temporal resolutions. For this purpose, we make use of the approach proposed by Müller et al. (Müller et al., 2005; Müller, 2015) for the CENS features. We use a smoothing window length defined by w and a downsampling factor d, both given in frames. After smoothing, we again normalize the feature frames using the ℓ_1 norm obtaining the smoothed features NNLS_d^w. For our experiments, we consider the following resolutions:

- $\mathrm{NNLS}_{\mathrm{local}}$: Local 10 Hz features (for feature F_{10})
- NNLS_5^{10}: Medium resolution (for feature F_9)
- $\mathrm{NNLS}_{100}^{200}$: Medium resolution (for feature F_8)
- $\mathrm{NNLS}_{\mathrm{global}}$: Global chroma histogram (for feature F_7).

Based on the different chroma representations, we calculate the complexity measure Γ_{Fifth}, which measures the spread of chroma energy around the circle of fifths. To this end, we re-sort the chroma values to an ordering of perfect fifth intervals (7 semitones):

$$c_q^{\mathrm{fifth}} = c_{(q \cdot 7 \mod 12)} \tag{4}$$

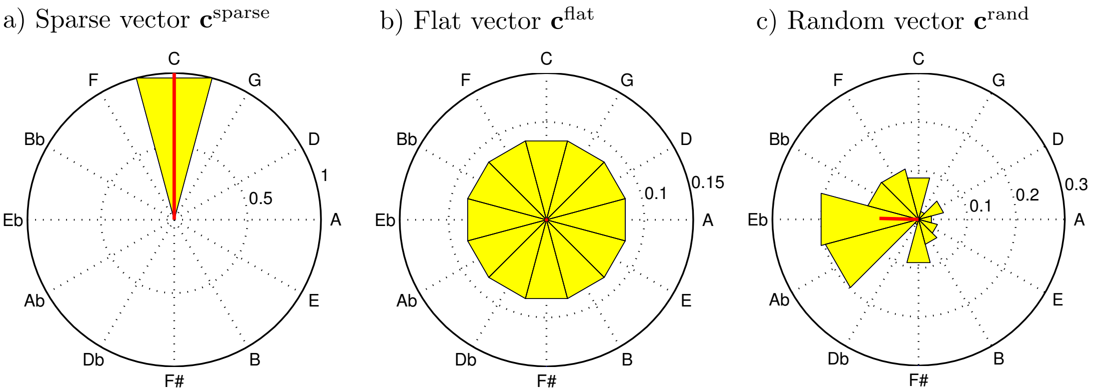

Figure S1. Circular interpretation of chroma vectors. The length of the yellow shapes corresponds to the chroma vector entries c_q with $q \in [0:11]$. The red line indicates the resultant vector. For a sparse chroma vector $\mathbf{c}^{\mathrm{sparse}}$, the resultant vector has length 1 (a). A flat vector $\mathbf{c}^{\mathrm{flat}}$ obtains length 0 (b). In (c), we illustrate this principle for a random-like chroma vector.

with $q \in [0:11]$ obtaining a circular distribution of the pitch class energies. From this, we calculate the length of the mean resultant vector

$$r_{\mathrm{fifth}}(\mathbf{c}) = \left| \sum_{q=0}^{11} c_q^{\mathrm{fifth}} \exp\left(\frac{2\pi i q}{12}\right) \right|. \tag{5}$$

From the resultant vector, we obtain the complexity value $\Gamma_{\mathrm{Fifth}}(\mathbf{c})$ as the angular deviation

$$\Gamma_{\mathrm{Fifth}}(\mathbf{c}) := \sqrt{1 - r_{\mathrm{fifth}}(\mathbf{c})}. \tag{6}$$

This way, Γ_{Fifth} describes the spread of the pitch classes. A short resultant vector—corresponding to a flat chroma vector—results in a high complexity value $\Gamma_{\mathrm{Fifth}} \approx 1$. A long vector leads to small values corresponding to low complexity. In previous work, we have demonstrated the behaviour of this feature (and other complexity features) for isolated chords as well as for the head movements on L. van Beethoven's piano sonatas (Weiß & Müller, 2014; Weiß, 2017).

S4 Chord Progression Features

In our paper, we also analyze chord transitions since such sequential properties may constitute meaningful style markers. For estimating the underlying chord symbols, we use the Vamp plugin Chordino.[1] This algorithm relies on NNLS chroma features (see Section 1) and incorporates Hidden Markov Models for concurrently estimating and smoothing the chord labels (Mauch & Dixon, 2010). We use the plugin together with the command line tool Sonic Annotator.[2]
The Chordino plugin allows for an adaptation of possible chord types using a dictionary file ("chord.dict", see Table S1). We modified this dictionary for our purpose by only using the four

Table S1. Dictionary file for the Chordino algorithm. This is the "chord.dict" file for configuring the Chordino Vamp plugin. We used this configuration to estimate the chords for our analyses. The first twelve entries refer to the bass notes, which we did not use. The last twelve entries indicate the active pitch classes for the respective chord type. We have considered the four basic triad types as well as five types of seventh chords. Regarding the nomenclature, the part after the first underscore relates to the quality of the basic triad (major, minor, diminished, or augmented). For the seventh chords, we indicate the quality of the seventh interval over the root note after the second underscore. The algorithm automatically generates circularly shifted versions of these templates to account for all twelve possible root notes.

```
_maj = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0
_min = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,0,0,0,0
_dim = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,0,0
_aug = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0
_dim_dim7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,1,0,0
_dim_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,1,0
_maj_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,1,0
_maj_maj7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,1
_min_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,0,0,1,0
```

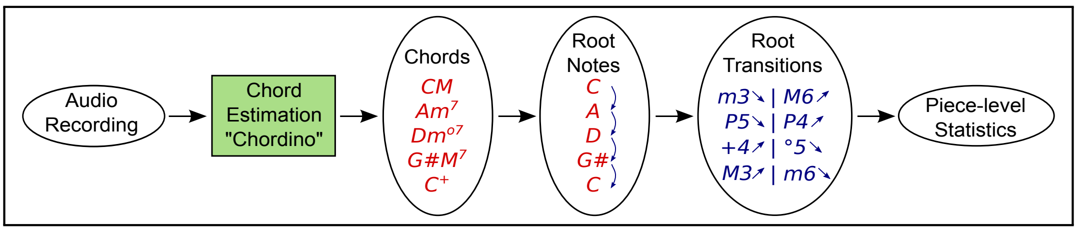

Figure S2. Estimation of root note transitions. In this schematic overview, we show the processing flow for estimating the frequency of root note transitions. First, we reduce the output of the chord estimator by only considering the root notes (without octave information). From this root note sequence, we calculate interval statistics on the piece level.

basic triad types (major, minor, diminished, and augmented triads) as well as five seventh chord types (major, dominant, minor, half-diminished, and full-diminished seventh chords). We do not use the bass chromagram to estimate chord inversions since, for classical music, the harmonic bass notes (i. e., the lowest note in a given voicing of a chord) do not necessarily lie within a low pitch range—in contrast to most popular music, which often includes an electric bass guitar. Regarding the chroma pre-processing, we use a window size of 16384 samples and a window increment of 4410 samples resulting in a feature resolution of 10 Hz (sampling rate 44100 Hz). For all other system parameters, we use the default values.

For our analyses, we only consider the relative root note distance of the chord transitions. To this end, we reduce the Chordino output by only keeping the root note information of the chords.

We count the occurrence of different melodic intervals between these root notes for all pairs of chord symbols (see Figure S2). Next, we divide the resulting numbers by the total number of chord transitions in order to obtain relative values for each piece. We do not consider transitions between chords with the same root note such as, for instance, the transition C major → C minor.

S5 Feature Aggregation and PCA

In Data Mining and Machine Learning, the set of computed features is usually given in a *feature matrix* $\mathcal{F} \in \mathbb{R}^{N \times I}$ where N is the number of features and I is the number of instances (in our cases: pieces, years, or composers). Typically, the number of features can be quite large ($N \gg 100$). Often, the feature matrix shows some kind of redundancy so that a lower dimensionality may be sufficient to capture the relevant information. In this case, feature aggregation techniques can be useful in order to obtain a representation of lower dimensionality $L < N$. An unsupervised aggregation method is *Principal Component Analysis* (PCA). This method constitutes a transformation of the feature vectors into a new basis with orthonormal basis vectors $\mathbf{w}^l := (w_1^l, \ldots, w_N^l)^\mathrm{T} \in \mathbb{R}^N,\ l \in [1 : N]$. The entries of $\mathbf{w}^l$ are called weights or *loadings*. The first component $\mathbf{w}^1$ points towards the *maximum variance* direction of the feature space. With increasing index l, a vector $\mathbf{w}^l$ describes a smaller fraction of the data's variance. Therefore, we can reduce the dimensionality of the feature space by only keeping the first $L < N$ components while still describing a large part of the variance. As an important preprocessing step for PCA, we have to subtract the mean vector over all instances from the initial feature vectors. Furthermore, it can be useful to divide the feature values by the standard deviation over all instances in order to equalize the contribution of the feature dimensions (Alpaydin, 2004).

In our experiments, we have an initial dimensionality of $N = 65$. For example, as basis for the K-means clustering in the paper's Figure 9, we have a year-wise feature matrix denoted by $\mathcal{F}_{\text{Years}} \in \mathbb{R}^{65 \times 315}$ (considering the 315 years of the timespan 1661–1975). Applying PCA, we obtain the aggregated feature F^* as a linear combination

$$F^* = w_1 F_1 + w_2 F_2 + \ldots + w_N F_N \tag{7}$$

where the weights $w_1, \ldots, w_N$ are determined by the PCA procedure.

We may regard feature aggregation as a more general procedure. For example, we can add up the root note transitions of ascending and descending perfect fifth intervals to a new feature. In this case, we need to choose the weights $w_{15} = w_{17} = 1$ while all other weights are zero. In the paper, we aggregate all plagal and all authentic progressions, respectively. Calculating the ratio of these quantities is the basis for Figure 4 in the paper.

S6 Diversity Curve of Piece Clustering

In the paper, we have presented clustering results for individual pieces. Figure 10 of the paper shows the evolution curves for the resulting cluster assignments as spindle plots. The spindles describe the fraction of pieces belonging to each cluster over the years. To analyze the homogeneity of styles in more detail, we present an analysis of diversity based on the year-wise clustering results. Inspired by Mauch et al. (2015), we use a diversity measure relating to the

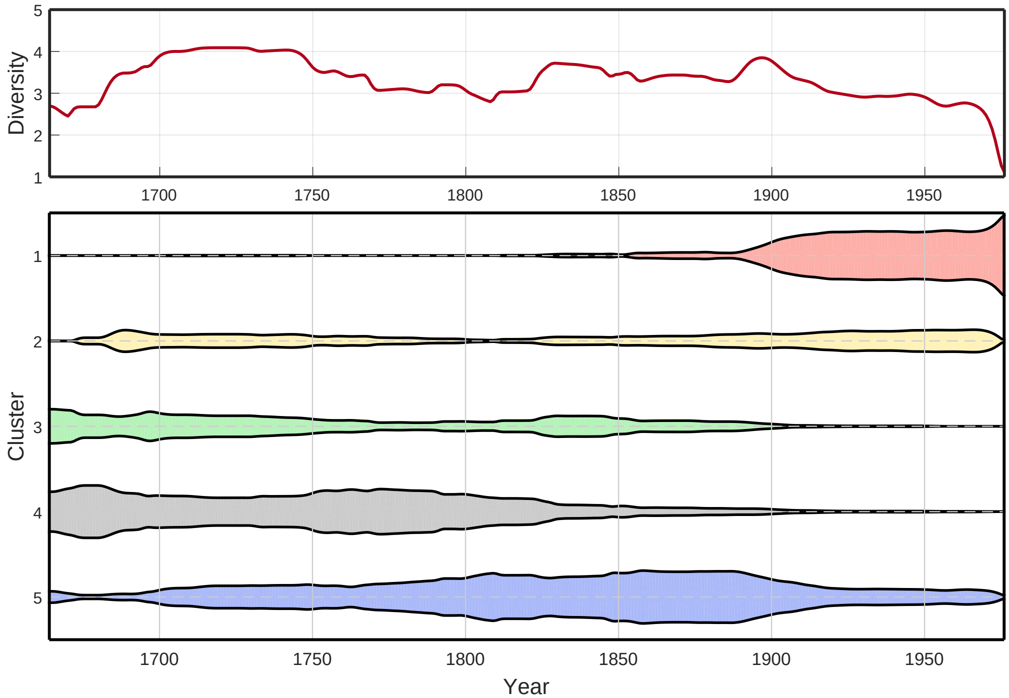

Figure S3. K-means clustering of individual pieces with $K = 5$. For each year, the fraction of pieces belonging to each cluster is indicated by the width of the respective spindle in the lower plot. The upper plot shows the diversity (effective number of clusters) over the years.

entropy (Jost, 2006). Let A_i^k be the fraction of pieces assigned to the cluster k in the year i, with $k \in [1:5]$ and $i \in [1661:1975]$. Then, we calculate the diversity measure D_i in the year i as follows:

$$D_i = \exp\left(-\sum_{k=1}^{5} A_i^k \ln A_i^k\right). \tag{8}$$

The upper plot of Figure S3 shows the resulting diversity curve. For completeness, and for a better comparison, we show the clustering results again in the lower plot. We observe low diversity values for the beginning and ending years of the analyzed timespan. Again, this is in accordance with our expectation since only few composers contribute there. The early 18th century constitutes a period of high diversity. This is an interesting observation since, in that time, both the old Baroque style with composers such as J. S. Bach and G. F. Handel and a variety of pre-classical styles with composers such as C. P. E. Bach, G. F. Telemann, J. Stamitz, or L. Mozart contribute. The classical period (roughly 1770–1825) seems to be an era of lower diversity. In contrast, the 19th century appears rather diverse Around 1900, we find a peak. This local maximum is probably caused by the rise of atonal style (Cluster 1) while other styles such as the Romantic style (Cluster 5) are still present. Though the interpretation of this diversity analysis is rather speculative, some of the observations provide meaningful information on the variety of composition styles over time.

Notes

1. http://isophonics.net/nnls-chroma
2. http://www.vamp-plugins.org/sonic-annotator

References

Alpaydin, E. (2004). *Introduction to machine learning.* Cambridge, MA: MIT Press.

Bartsch, M. A., & Wakefield, G. H. (2005). Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on Multimedia*, *7*(1), 96–104.

Di Giorgi, B., Dixon, S., Zanoni, M., & Sarti, A. (2017). A data-driven model of tonal chord sequence complexity. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *25*(11), 2237–2250.

Ellis, D. P., & Poliner, G. E. (2007). Identifying 'cover songs' with chroma features and dynamic programming beat tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (Vol. 4, pp. 1429–1432). Honolulu, HI.

Forte, A. (1973). *The structure of atonal music.* New Haven and London: Yale University Press.

Fujishima, T. (1999). Realtime chord recognition of musical sound: A system using common lisp music. In *Proceedings of the International Computer Music Conference (ICMC)* (pp. 464–467). Beijing, China.

Gómez, E. (2006). Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing*, *18*(3), 294–304.

Gómez, E., & Herrera, P. (2004). Estimating the tonality of polyphonic audio files: Cognitive versus machine learning modelling strategies. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)* (pp. 92–95). Barcelona, Spain.

Hanson, H. (1960). *Harmonic materials of modern music: Resources of the tempered scale.* New York, NY: Appleton Century Crofts.

Honingh, A., & Bod, R. (2010). Pitch class set categories as analysis tools for degrees of tonality. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)* (pp. 459–464). Utrecht, The Netherlands.

Honingh, A., Weyde, T., & Conklin, D. (2009). Sequential association rules in atonal music. In *Mathematics and computation in music (MCM)* (pp. 130–138). Berlin and Heidelberg, Germany: Springer.

Jost, L. (2006). Entropy and diversity. *Oikos*, *113*(2), 363–375.

Lee, K. (2006). Automatic chord recognition from audio using enhanced pitch class profile. In *Proceedings of the International Computer Music Conference (ICMC)* (pp. 306–311). New Orleans, LA.

Mauch, M., & Dixon, S. (2010). Approximate note transcription for the improved identification of difficult chords. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)* (pp. 135–140). Utrecht, The Netherlands.

Mauch, M., MacCallum, R. M., Levy, M., & Leroi, A. M. (2015). The evolution of popular music: USA 1960–2010. *Royal Society Open Science*, *2*(5).

Müller, M. (2015). *Fundamentals of music processing.* Cham, Switzerland: Springer.

Müller, M., & Ewert, S. (2010). Towards timbre-invariant audio features for harmony-based music. *IEEE Transactions on Audio, Speech, and Language Processing*, *18*(3), 649–662.

Müller, M., Kurth, F., & Clausen, M. (2005, October). Chroma-based statistical audio features for audio matching. In *Proceedings of the IEEE Workshop on Applications of Signal Processing (WASPAA)* (pp. 275–278). New Paltz, NY.

Quinn, I. (2001). Listening to similarity relations. *Perspectives of New Music*, *39*(2), 108–158.

Streich, S. (2006). *Music complexity: A multi-faceted description of audio content* (Doctoral dissertation). Universitat Pompeu Fabra, Barcelona, Spain.

Weiß, C. (2017). *Computational methods for tonality-based style analysis of classical music audio recordings* (Doctoral dissertation). Ilmenau University of Technology, Ilmenau, Germany.

Weiß, C., Mauch, M., & Dixon, S. (2014). Timbre-invariant audio features for style analysis of classical music. In *Proceedings of the Joint International Computer Music Conference (ICMC) and Sound and Music Computing Conference (SMC)* (pp. 1461–1468). Athens, Greece.

Weiß, C., & Müller, M. (2014). Quantifying and visualizing tonal complexity. In *Proceedings of the Conference on Interdisciplinary Musicology (CIM)* (pp. 184–187). Berlin, Germany.

Weiß, C., & Müller, M. (2015). Tonal complexity features for style classification of classical music. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (pp. 688–692). Brisbane, Australia.