MODELLING HIERARCHICAL KEY STRUCTURE WITH PITCH SCAPES

Robert Lieck
Digital and Cognitive Musicology Lab
EPFL, Switzerland
research@robert-lieck.com

Martin Rohrmeier
Digital and Cognitive Musicology Lab
EPFL, Switzerland
martin.rohrmeier@epfl.ch

ABSTRACT

Musical form and syntax in Western classical music are hierarchically organised on different timescales. One of the most important features of this structure is the organisation of modulations between different keys throughout a piece. Music theoretical research has established taxonomies of prototypical modulation plans for different modes and musical forms. However, these prototypes still require empirical validation based on quantitative statistical methods and cannot be retrieved automatically so far.

In this paper, we present a novel method to infer prototypical modulation plans from musical corpora. A modulation plan is formalised as a transposition-invariant probabilistic model over the underlying pitch class distributions based on a hierarchical *pitch scape* representation. Prototypical modulation plans can be learned in an unsupervised manner by training a mixture model (similar to a Gaussian mixture model) on the data, so that different prototypes appear as distinct clusters.

We evaluate our approach by performing hierarchical clustering on a corpus of more than 150 Baroque pieces, with the extracted clusters showing excellent agreement with the most common prototypes postulated in music theory. Our method bears a great potential for modelling, analysis and discovery of hierarchical key structure and prototypes in corpora across a broad range of musical styles. An accompanying library is available at: github.com/robert-lieck/pitchscapes.

1. INTRODUCTION

The hierarchical structure of a piece in Western classical music is strongly determined by musical form [1] and harmonic syntax [2, 3], based on different aspects, such as repetition and variation of the rhythmic, melodic and harmonic content and hierarchical relations between different harmonies.

A central aspect that links musical form and harmonic syntax is the modulation plan of a piece. Western musicology assumes a number of prototypical modulation plans that describe the overarching tonal structure of a piece, such as I–V–I for pieces in major or i–III–i for pieces in minor [1]. These prototypes have a long-standing history in musicology and have emerged from inspection of numerous individual pieces and agreement among experts. However, a quantitative validation based on statistical methods constitutes an important supplement to confirm and refine the music theoretic findings. Furthermore, they cannot be automatically retrieved from musical data, which impedes large-scale investigations and the application to other styles and genres of music.

In this paper, we present a method to retrieve prototypical modulation plans from large corpora of musical pieces in an unsupervised manner. This is achieved by modelling the overall corpus as a mixture of multiple prototypes, similar to how Gaussian mixture models [4] can be applied to clustering in Euclidean space. A prototype is represented by a transposition-invariant Bayesian model that describes the pitch content of a piece (pitch class distributions) on multiple time scales. Modelling is based on a novel *pitch scape* representation of the musical content, which allows to account for the hierarchical structure inherent to both musical form and harmonic syntax. We evaluate our model on a corpus of more than 150 Baroque pieces, with the extracted clusters showing excellent agreement with the most common prototypes postulated in music theory.

By providing a solid statistical approach to modelling prototypical modulation plans, we make an important contribution to connecting music theory and empirical science. Our approach relies on minimal prior assumptions, works on simple pitch data, and learns prototypes in an unsupervised manner, which bears a great potential for modelling, analysis and discovery of hierarchical key structure and prototypes in corpora across a broad range of musical styles.

In the remainder of the paper, we describe the underlying *pitch scape* representation in Section 2, introduce the probabilistic Bayesian model that is used to learn prototypes and prototype mixtures from musical corpora in Section 3, and present and discuss the results of our evaluation in Section 4.

2. PITCH SCAPES

We model prototypical modulation plans based on a novel *pitch scape* representation of the musical content. Pitch scapes (see Figure 1 for an illustration) represent the pitch content of a piece on multiple time scales and can be for-

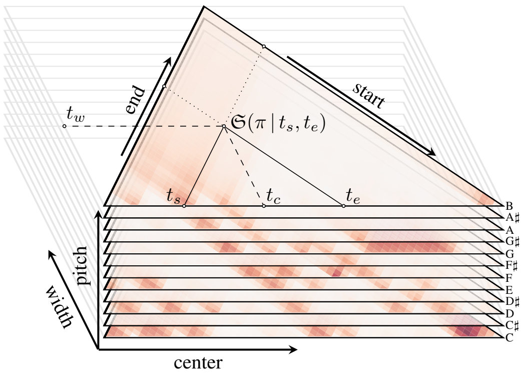


Figure 1. Pitch scape (Prelude in C major, BWV 846, Johann Sebastian Bach). The two time values can be specified in start-end-coordinates (t_s and t_e) or in center-width-coordinates (t_c and t_w).

mally defined as the conditional probability distribution of the pitch classes for a given section of the piece:

Definition 1 (Pitch Scape)**.** *A pitch scape $\mathfrak{S}$ is a function that maps each proper time interval $[t_s, t_e]$ ($t_s < t_e$) to a pitch class distribution*

$$\mathfrak{S} : \mathbb{R} \times \mathbb{R} \to [0,1]^{12} \,, \quad \sum_{\pi=0}^{11} \mathfrak{S}(\pi \,|\, t_s, t_e) = 1 \,. \tag{1}$$

A pitch scape can equivalently be conceived as a conditional probability distribution $\mathfrak{S}(\pi \,|\, t_s, t_e)$ with three variables or a vector-valued function $\mathfrak{S}(t_s, t_e)$ in two variables.

Pitch scapes are inspired by scape plot visualisations, to which we draw the connection in Section 2.1, while Section 2.2 describes how to compute pitch scape estimates for a given piece.

2.1 Pitch Scape Visualisation

Scape plot visualisations were introduced in [5,6] to depict key estimates for different sections of a piece in a hierarchical triangular plot and have since been used for a variety of visualisation tasks [7–11].

Visualising the entire information contained in a three-dimensional pitch scape in a single two-dimensional plot is difficult. However, there are two convenient ways to visualise the relevant information. First, the 12 components can be visualised separately by creating one scape plot per pitch class. This preserves the entire information but does not foster musical intuition because information about simultaneous events is scattered across multiple plots. Alternatively, a key finding algorithm can be employed [12–15] to map the pitch class distribution of each point to a colour value. This corresponds to a key-scape plot of the pitch scape. For illustration, we show in Figure 2 an overlay of the 12 separate pitch-scape plots and the corresponding key-scape plot.

The colour mapping for a key-scape plot can be realised in different ways. We use a template-based key finder that

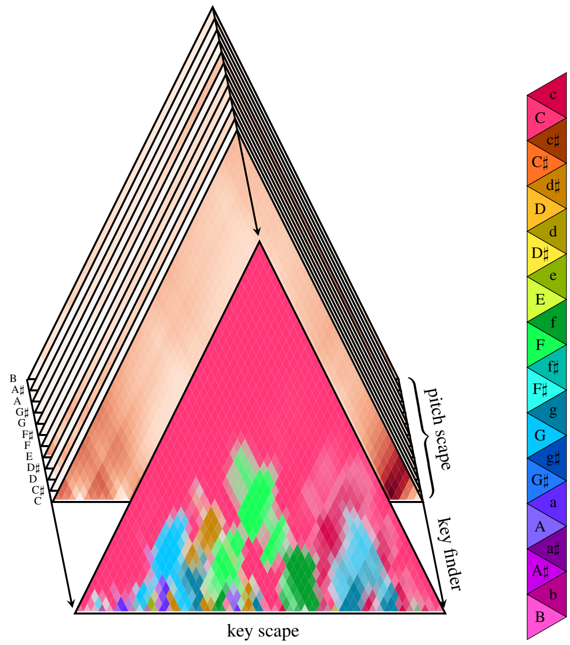


Figure 2. Separate pitch-scape plots and resulting key-scape plot for the prelude in C major, BWV 846, Johann Sebastian Bach (colour legend for keys on the right).

provides a score value for each major and minor key. The scores can be transformed into a probability distribution $p(k)$ using a soft-max function. After choosing a unique colour for each key, $p(k)$ can be used to interpolate between colours by computing their weighted average. To define the colour for each key, we let the hue value vary either along the circle of fifths or chromatically, which has complementary advantages. Fifth-based hue maps related keys to similar colours, while chromatic hue allows to better distinguish them. We add a lightness offset to distinguish major and minor keys and map the entropy of $p(k)$ to saturation. Entropy-based saturation allows to indicate regions with uncertain key classification and avoids uninterpretable colour blends.

2.2 Pitch Scape Estimates

The pitch scape of a piece is computed from its musical content and reflects the probability of a certain pitch class to occur within the specified time interval. As we are working in a Bayesian framework, we model the pitch scape of a piece as a posterior estimate given a prior distribution and the observed notes. To formally define the pitch scape estimate of a piece, we first define its pitch class density:

Definition 2 (Pitch Class Density)**.** *The pitch class density $\delta(\pi \,|\, t)$ for pitch class π at time t corresponds to the normalised pitch class counts over all tones that sound at time t*

$$\delta(\pi \,|\, t) := \frac{1}{\max\{1, |T_t|\}} \sum_{\tau \in T_t} [\![\tau \bmod 12 = \pi]\!] \,, \tag{2}$$

where T_t is the multiset of all tones (as integers in MIDI pitch representation) sounding at time t; $[\![\cdot]\!]$ is the Iverson *bracket, which equals 1 if its argument is true and 0 otherwise; and the* max *avoids division by zero for silent parts where $T_t = \emptyset$ is the empty set.*

Using the pitch class density, we define the pitch scape estimate as follows:

Definition 3 (Pitch Scape Estimate)**.** *The posterior estimate of the pitch scape $\mathfrak{S}(\pi \mid t_s, t_e)$ for pitch class π and time interval $[t_s, t_e]$ is*

$$\mathfrak{S}(\pi \mid t_s, t_e) := \underbrace{\frac{1}{t_e - t_s + 12c}}_{\textit{normalisation}} \left[\overbrace{c}^{\textit{prior counts}} + \underbrace{\int_{t_s}^{t_e} \delta(\pi \mid t)\, dt}_{\textit{overall pitch class counts}} \right] , \quad (3)$$

where the integral over the pitch class density computes the overall pitch class counts, $c \geq 0$ specifies the prior counts, and the leading term ensures proper normalisation.

Using zero prior counts $c = 0$ thus corresponds to using the average pitch class density as pitch scape estimate (in Bayesian terms this would be a maximum likelihood estimate). In contrast, using a prior count of $c = 1$, which is done throughout the paper, corresponds to a Bayesian maximum posterior estimate with a uniform prior over pitch classes ($c = 1$ corresponds to a uniform Dirichlet distribution, which is the appropriate conjugate prior for the categorical distribution over pitch classes). Note that choosing $c > 0$ also circumvents the zero-count problem for silent parts.

The relative weight of the overall pitch class counts, computed in the integral in (3), depends on the scale on which time is measured. Throughout the paper, we measure time in quarter notes, so that a time interval of one quarter note has the weight of a single observation. That means, for instance, a single pitch sustained for two quarter note adds two to the respective overall pitch class counts; two different pitches sustained for one quarter note add half a count each; and three pitches sustained for an eight note add one sixth count each. Thus, for small time intervals the prior counts c introduce a significant bias towards a uniform pitch class distribution, while for large time intervals they have a vanishingly small weight relative to the overall pitch class counts (e.g. a 32-bar piece in 4/4 yields a total of 128 pitch class counts, so that the prior counts do not cause a major change of the estimated pitch class distribution for the entire piece).

3. MODELLING KEY STRUCTURE

We define our model for mixtures of prototypes in two steps. First, we define a probabilistic pitch scape model of a transposition-invariant modulation plan (Section 3.1). Based on this model for single prototypes, we define a mixture model (Section 3.2) that incorporates explicit transposition and models a musical corpus as a mixture of multiple prototypes.

3.1 Prototypes

The idea of a prototype is to specify an object that represents a subset of the data. In probabilistic modelling this corresponds to defining a probability distribution for which a subset of the data has a high likelihood. Additionally, this distribution should be unimodal so that its mode can be taken as a representative of all data points belonging to that prototype. In n-dimensional Euclidean space, prototypes can for instance be defined using multivariate Gaussian distributions.

When defining prototypes for pitch scapes, we are facing some additional challenges that will be addressed in the following. First, the output of a pitch scape is a categorical distribution (over pitch classes), which has to be normalised. Second, time is continuous so that a pitch scape itself is an inherently continuous object. Third, the prototypes postulated in music theory are formulated in terms of scale degrees, which makes them transposition-invariant (i.e. transposing a piece does not affect its relation to a specific prototype). The first two points will be addressed in the following Section 3.1.1, the third point is resolved in Section 3.1.3 and incorporated in the mixture model in Section 3.2.

3.1.1 Definition

We address the first two points by defining a prototype as a point-wise Dirichlet distribution with a time-dependent parameter vector α. The Dirichlet distribution is the conjugate prior of a categorical distribution and acts as a likelihood function if the observations themselves are categorical distributions, as it is the case for individual points in a pitch scape (first point). Making its parameter vector time-dependent additionally allows it to vary continuously over the pitch scape (second point). Formally, a prototype is defined as follows:

Definition 4 (Prototype)**.** *Given a function*

$$\alpha : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+^{12} \quad (4)$$

that maps each proper time interval $[t_s, t_e]$ ($t_s < t_e$) to a vector with positive entries, a prototype is defined as the point-wise Dirichlet distribution with parameter vector α. The likelihood of observing a pitch class distribution Π for the interval $[t_s, t_e]$ given α is

$$p(\Pi \mid \alpha, t_s, t_e) = \mathrm{Dir}(\Pi; \alpha(t_s, t_e)) \, . \quad (5)$$

The log-likelihood of observing a full pitch scape $\mathfrak{S}$ given α is

$$\log p(\mathfrak{S} \mid \alpha) = \frac{2}{T^2} \iint\limits_{0 \leq t_s < t_e \leq T} \log \mathrm{Dir}\big(\mathfrak{S}(t_s, t_e); \alpha(t_s, t_e)\big) dt_s dt_e \, , \quad (6)$$

where T is the duration of the piece.

The definition of the log-likelihood in (6) is equivalent to the (negative) cross-entropy of an infinite number of uniform samples from the pitch scape. It differs from a simple integration of (5) only by the normalisation $\frac{2}{T^2}$, which rescales it to the magnitude of a single observation and makes it invariant to the duration of the piece. Note that both (5) and (6) are probability *density* functions with the usual implications (i.e. they can be greater than 1; their log can be positive; their cross-entropy can be negative).

3.1.2 *Proxy Function*

To learn prototypes from data, we define α via a three dimensional real-valued proxy function $\widetilde{\alpha}$ that has a set of adjustable parameters θ and an open parameter τ

$$\widetilde{\alpha}^{(\theta,\tau)} : \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \mathbb{R} \,. \tag{7}$$

The domain of interest is $[0, 1] \times [0, 1] \times \mathbb{Z}_{12}$ with the first two arguments specifying the time interval in normalised center-width-coordinates and the third specifying the pitch class. The π^{th} component of α is then defined to be

$$\alpha_\pi^{(\theta,\tau)}(t_s, t_e) := e^{\widetilde{\alpha}^{(\theta,\tau)}(\bar{t}_c,\bar{t}_w,\pi)} \tag{8}$$

with

$$\bar{t}_c = \tfrac{1}{2T}(t_s + t_e) \qquad \bar{t}_w = \tfrac{1}{T}(t_e - t_s)\,, \tag{9}$$

where T is the duration of the piece that is to be modelled.

3.1.3 *Fourier Representation*

We parameterise $\widetilde{\alpha}$ as a Fourier series in three dimensions [16]

$$\widetilde{\alpha}^{(\theta,\tau)}(\mathbf{x}) = \sum_{\mathbf{n}} \theta_{\mathbf{n}} e^{2\pi i \mathbf{k}_{\mathbf{n}}\cdot\mathbf{x}}\,, \tag{10}$$

where π (only in this equation!) is the mathematical constant. The index vector $\mathbf{n}$, wave vector $\mathbf{k}_{\mathbf{n}}$, and location vector $\mathbf{x}$ are

$$\mathbf{x} := (\bar{t}_c, \bar{t}_w, \pi) \qquad n_c \in \{-N_c, \ldots, N_c\} \tag{11}$$

$$\mathbf{n} := (n_c, n_w, n_\pi) \qquad n_w \in \{-N_w, \ldots, N_w\} \tag{12}$$

$$\mathbf{k}_{\mathbf{n}} := \left(\sigma_c n_c, \sigma_w n_w, \tfrac{n_\pi+\tau}{12}\right) \; n_\pi \in \{-6, \ldots, 6\}\,. \tag{13}$$

N_c and N_w allow to independently control the smoothness (or bandwidth) of $\widetilde{\alpha}$ for the center and width dimension, respectively; $\tau \in \mathbb{Z}_{12}$ represents the transposition of the prototype (see below); and $\sigma = 1 - \frac{1}{2N}$ is a scaling factor. Scaling is required because we do not want $\widetilde{\alpha}$ to have periodic boundaries conditions in the time dimensions. The Nyquist frequency of the unscaled function is $2N$, the scaling factor thus stretches the function such that a critical fraction of $\frac{1}{2N}$ is moved out of the interval $[0, 1]$. This is not relevant for the pitch dimension because the space of pitch classes is inherently periodic and, moreover, we have a complete discrete Fourier series that allows to represent any function exactly. As $\widetilde{\alpha}$ is real-valued, $\theta_{\mathbf{n}}$ and $\theta_{-\mathbf{n}}$ are complex conjugates and (due to the properties of the discrete Fourier transform) all coefficients with $n_\pi = \pm 6$ are real-valued. We can thus store the parameters θ in a real-valued array of dimensions $(2N_c + 1, 2N_w + 1, 12)$.

As $\widetilde{\alpha}$ (and thus α) are periodic in the pitch dimension, the Fourier representation can be understood as a transposition-invariant formulation of a prototype. When creating a concrete instance of the prototype, τ needs to be specified and defines a specific transposition by inducing a corresponding phase shift through the cyclic pitch class space.

3.2 Mixture Model

In Section 3.1 we defined prototypes that have a point-wise Dirichlet distribution (Definition 4) and adjustable parameters θ. We will now build a transposition-invariant mixture model using these prototypes. The overall structure of the model is shown in Figure 3 as a graphical model [17] and will be explained in detail below.

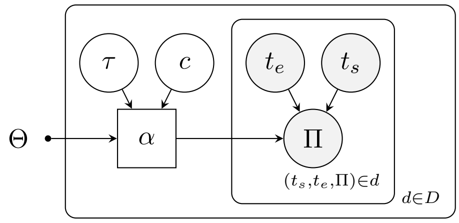


Figure 3. Graphical representation of our mixture model. Θ are the prototype parameters; c and τ the piece-specific cluster index and transposition; α the deterministically generated prototype instance; and $\Pi = \mathfrak{S}(t_s, t_e)$ the pitch scape values at intervals $[t_s, t_e]$ (see text for more details).

Our model is similar to classical topic models for corpora [18–20] with two nested levels. Each piece (or document) d in the data set D is generated independently from a specific prototype with parameters $\theta = \Theta_c$ and transposition τ (outer plate) and for a specific piece, each point Π in its pitch scape is generated independently (inner plate). [1]

3.2.1 *Inference*

We want to find parameters Θ^* that minimise the cross-entropy (i.e. maximise the likelihood) of our data D

$$\Theta^* = \operatorname{argmin}_\Theta -\tfrac{1}{|D|} \log p(D \,|\, \Theta)\,, \tag{14}$$

where

$$\log p(D \,|\, \Theta) = \sum_{d\in D} \log \sum_{c,\tau} p(d \,|\, \alpha^{(\Theta_c,\tau)})\, p(c)\, p(\tau)\,, \tag{15}$$

is the data log-likelihood with the latent variables c and τ being marginalised out. The prior terms $p(c)$ and $p(\tau)$ are assumed to be constant so that a priori no specific prototype or transposition is preferred. We use

$$\log p(d \,|\, \alpha) = \frac{1}{|d|} \sum_{(t_s,t_e,\Pi)\in d} \log \mathrm{Dir}\big(\Pi; \alpha(t_s, t_e)\big) \tag{16}$$

to approximate the piece likelihood (6) based on a finite number of uniform samples. Marginalising out c and τ also readily yields the normalisation factor for the cluster and transposition probability for a piece

$$p(c, \tau \,|\, d) \propto p(d, c, \tau) = p(d \,|\, \alpha^{(\Theta_c,\tau)})\, p(c)\, p(\tau)\,. \tag{17}$$

The optimal parameters Θ^* can be found by performing gradient descent on the cross-entropy (14).

[1] The assumption of different points in the pitch scape being generated independently is obviously incorrect, which is common to all topic models and the reason why they are not well suited to generate coherent data (e.g. text or music). In fact, in a pitch scape the values at different locations are highly correlated and would ideally be modelled as a single continuous latent function. One approach to achieve this are Gaussian processes (GPs) [21]. However, GPs are computationally expensive and GPs for multi-class classification have complex kernel functions and require approximations of the analytically intractable posterior distribution [21–23]. As we are primarily interested in extracting the mean pitch scape (corresponding to the GP prior), which represents a specific prototype, we therefore chose the simpler approach of defining prototypes as a point-wise Dirichlet distribution.

3.2.2 Hierarchical Clustering

Training the mixture model on a data set allows to perform unsupervised clustering with a fixed number of clusters. However, our motivation is a comparison with the prototypes described in the music theory literature. Instead of choosing a fixed number of clusters, we are rather interested in how clusters split hierarchically from more generic prototypes to more specific ones. We therefore take a hierarchical top-down clustering approach.

We start by training a single prototype on the whole corpus and perform a binary split of this cluster by using it to initialise a mixture of two prototypes, while adding minimal noise (10^{-8}) to the parameters θ to allow the clusters to properly split. This procedure is then recursively and *separately* applied to the resulting prototypes. To this end, the probability $p(d \mid c')$ of a piece d to fall into the parent cluster c' is used as a weight in (15) when training the two child clusters. This ensures a clear assignment between parent and child clusters and implies that only pieces that fell into the parent cluster influence the children.

After establishing a hierarchy of prototypes in this way, we perform a joint refinement of the resulting clusters. To this end, *all* final child clusters are combined in a single model while lifting the parent-specific piece weights. This serves a two-fold purpose. First, the prototypes may be sharpened as interactions between the clusters can now be exploited. Second, it acts as a sanity check for the established hierarchy: If the child clusters remain stable in the refinement phase, this indicates consistency of the hierarchical splitting.

4. EVALUATION

4.1 Experimental Setup

The model was implemented in `PyTorch` [24] and the parameters Θ were optimised via gradient descent using the Adam optimiser [25]. The "warm start" with pre-initialised clusters was realised by using a small initial learning rate (10^{-5}) to allow for the mean and variance estimators (internals of the Adam optimiser) to stabilise before reaching the normal learning rate (10^{-3}).

We trained our model on a corpus of 155 Baroque pieces in MIDI format by Johann Sebastian Bach (84%: WTK I/II; Brandenburgisches Konzert No. 5; Inventions and Sinfonias), Georg Friedrich Händel (4%: HWV 264, Movement 2, 4, 9, 10, 11, and 13; HWV 435), and Domenico Scarlatti (12%: Sonatas), see Appendix C for a complete list. As opposed to later periods with an increasing amount of chromaticism, the modulation plans of Baroque pieces are expected to more closely conform to the respective prototypes. Each piece was sampled by choosing interval start and end points on a uniform time grid of $n = 50$ points, resulting in $n(n-1)/2 = 1225$ samples per piece.

Hierarchical clustering was performed up to depth 3 (8 final clusters) with subsequent refinement (see Section 3.2.2). The hierarchy and final prototypes are shown in Figure 4 and discussed below.

4.2 Results and Discussion

The root cluster, which was trained on the entire corpus, represents a generic diatonic prototype. It does not unambiguously belong to a particular mode, being classified as minor based on 'Albrecht' profiles (as in Figure 4) and major based on 'Temperley' profiles. This is also reflected in the separate pitch-scape plots (Table 1 in Appendix B), which have strong weights for the entire pentatonic segment of the line of fifths (C–G–D–A–E). This can be interpreted as confirming the view that Baroque music is fundamentally diatonic.

The initial split results in a clear separation of major and minor keys, with cluster (0) and all its descendants being globally classified as minor pieces while (1) and its descendants are classified as major. From now on we can see a pronounced weight on the tonic pitch class (C for major, A for minor) at the beginning and end of a piece in the separate pitch-scape plots. This split into major and minor prototypes again is an important finding that confirms these two modes to be dominant in Baroque music.

4.2.1 Prototypes in Minor

The next split of the minor cluster separates the two most common prototypes. Prototypical minor-mode pieces are assumed to either modulate to the key of v (the dominant) or to the key of III (relative major) before returning to i (tonic), which corresponds to (0,0) and (0,1) and their descendants, respectively. Note that the cluster (0,1,0) also has a strong tendency to modulate to III, which becomes more apparent when using 'Temperley' profiles.

The i–v–i modulation plan of cluster (0,0,0) and (0,0,1) is one of the two standard prototypes for minor pieces. In (0,0,1), the v is more pronounced and the middle section also has a certain tendency to modulate to III, possibly even including a short VII passage (see 'Temperley' profiles). This corresponds to a i–III–(VII)–v–III–i modulation plan, which is a common subtype of the i–v–i prototype that features two fifth-related, modally distinct key pairs: i–v and III–VII.

Cluster (0,1,0) and (0,1,1) both fall under the general i–III–i prototype. The (0,1,0) cluster has a less pronounced III, which may be partly due to the III being at different locations in the corresponding pieces, thus leading to smoothing/averaging. According to 'Krumhansl' profiles, there is a tendency for modulation to v in the middle section and 'Temperley' profiles classify larger parts of the middle section as III. Cluster (0,1,1) has an additional modulation to the iv after the III, possibly with a short return to the i in between (again this could also be an effect of averaging over multiple pieces), representing the common subtype i–III–(i)–iv–i.

4.2.2 Prototypes in Major

Major pieces are generally assumed to modulate to V before going back to I. However, this general prototype can be elaborated in different ways. For the split of the major cluster (1), we see a very pronounced I–V–I prototype on the left with (1,0) and its child (1,0,0).

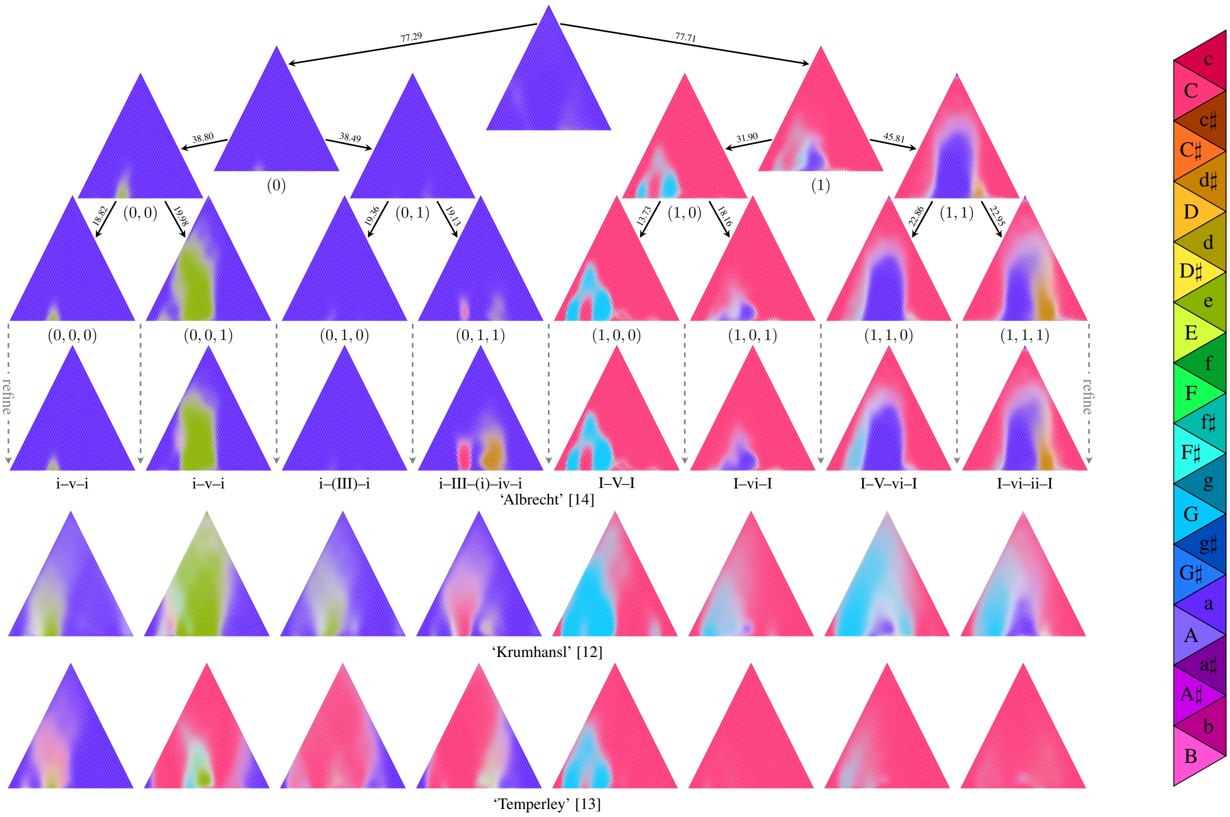


Figure 4. Results of hierarchical clustering with subsequent refinement. To visualise the prototypes, the transposition parameter τ was fixed to minimise the accidentals of the diatonic root cluster. The corresponding absolute keys are shown in the chromatic colour scale (right). However, only relative keys (scale degrees) bear interpretable meaning as the prototypes are inherently transposition-invariant. Prototypes are labelled with a hierarchical index; the final prototypes (after refinement) are labelled with the corresponding modulation plan in Roman numeral notation; numbers on the arrows indicate the number of pieces falling into the respective cluster. Key estimates for colouring are computed using 'Albrecht' [14] templates; the final prototypes are repeated using 'Krumhansl' [12] and 'Temperley' [13] templates to improve interpretability. For better disambiguation, Figure 5 and Figure 6 in Appendix A show chromatic and fifth-based colouring in comparison.

The remaining three clusters all belong to one of the most common elaborations of the I–V–I prototype with an additional vi (relative minor) passage after the V. The V passage is most clearly pronounced in the (1,1,0) cluster, to a lesser extent in the (1,1,1) and even less in the (1,0,1) cluster (see especially the 'Krumhansl' profiles). Notably, (1,1,1) has an additional ii passage after the vi.

The I–vi–I and the I–vi–ii–I cluster taken separately do not contradict expectations from music theory, but due to the missing V they are less typical than the other prototypes so far. However, when being combined with the I–V–vi–I cluster, theses clusters form the very common subtype I–V–vi–ii–I [1]. This is typical in Baroque music but also in modern Pop music, where on the chord-level this sequence is known as "the four chord song" (optionally with a IV as an equivalent pre-dominant replacement of the ii). This combination of multiple prototypes suggests the existence *prototype sub-spaces*.

5. CONCLUSION

To address the problem of modelling and automatic retrieval of prototypical modulation plans from a corpus of musical pieces, a probabilistic Bayesian model of transposition-invariant prototypes was introduced. This model was based on a novel hierarchical *pitch scape* representation of the musical content. We learned prototypical modulation plans from a corpus of Baroque pieces, empirically confirming common prototypes postulated in music theory. Extending the conventional music theoretical concepts, we found that continuous *prototype sub-spaces* can be generated as the superposition of multiple prototypes.

Our approach relies on minimal prior assumptions, works on simple pitch data and delivers robust results while being scalable to large data sets. It can therefore be applied to model, analyse and discover hierarchical key structures and prototypes in a wide range of musical styles and genres, including diachronic studies of musical form and syntax in Western classical music, the influence of style- and composer-specific elements, and the investigation of modulation plans in other genres such as Jazz, Pop and Rock music. Therefore, our approach is suited for numerous applications and contributes a valuable method for music information retrieval.

6. ACKNOWLEDGMENTS

We thank Markus Neuwirth, Fabian Moss, Johannes Hentschel, Uri Rom, and Dror Chawin for valuable discussions and feedback about the musical interpretation, and Leona Wall for critically revising the initial draft of the paper. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 760081 – PMSB.

7. REFERENCES

[1] E. Aldwell and C. Schachter, *Harmony and Voice Leading*. Thomson/Schirmer, 2003.

[2] M. Rohrmeier and M. Pearce, "Musical Syntax I: Theoretical Perspectives," in *Springer Handbook of Systematic Musicology*, R. Bader, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, pp. 473–486.

[3] M. Rohrmeier, "The Syntax of Jazz Harmony: Diatonic Tonality, Phrase Structure, and Form," *Music Theory and Analysis (MTA)*, vol. 7, no. 1, pp. 1–63, Apr. 2020.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed. Springer, 2007.

[5] C. S. Sapp, "Harmonic Visualizations of Tonal Music." in *ICMC*, vol. 1. Citeseer, 2001, pp. 419–422.

[6] ——, "Visual hierarchical key analysis," *Computers in Entertainment*, vol. 3, no. 4, p. 3, Oct. 2005.

[7] M. Müller and N. Jiang, "A Scape Plot Representation for Visualizing Repetitive Structures of Music Recordings." in *ISMIR*. Citeseer, 2012, pp. 97–102.

[8] N. Jiang and M. Müller, "Towards efficient audio thumbnailing," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2014, pp. 5192–5196.

[9] C. Weiß and M. Müller, "Quantifying and visualizing tonal complexity," in *Proceedings of the Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 184–187.

[10] C. Vaquero, "A quantitative study of seven historically informed performances of Bach's bwv1007 Prelude," *Early Music*, vol. 43, no. 4, pp. 611–622, Nov. 2015.

[11] S. Park, T. Kwon, J. Lee, J. Kim, and J. Nam, "A Cross-Scape Plot Representation for Visualizing Symbolic Melodic Similarity," in *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019*, A. Flexer, G. Peeters, J. Urbano, and A. Volk, Eds., 2019, pp. 423–430.

[12] C. L. Krumhansl and E. J. Kessler, "Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys." *Psychological review*, vol. 89, no. 4, p. 334, 1982.

[13] D. Temperley and E. W. Marvin, "Pitch-Class Distribution and the Identification of Key," *Music Perception*, vol. 25, no. 3, pp. 193–212, Feb. 2008.

[14] J. Albrecht and D. Shanahan, "The use of large corpora to train a new type of key-finding algorithm: An improved treatment of the minor mode," *Music Perception: An Interdisciplinary Journal*, vol. 31, no. 1, pp. 59–67, 2013.

[15] J. D. Albrecht and D. Huron, "A Statistical Approach to Tracing the Historical Development of Major and Minor Pitch Distributions, 1400-1750," *Music Perception*, vol. 31, no. 3, pp. 223–243, Feb. 2014.

[16] B. G. Osgood, *Lectures on the Fourier Transform and Its Applications*. American Mathematical Soc., 2019, vol. 33.

[17] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT press, 2009.

[18] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, pp. 993–1022, Mar. 2003.

[19] M. Steyvers and T. Griffiths, "Probabilistic Topic Models," *Handbook of latent semantic analysis*, vol. 427, no. 7, pp. 424–440, 2007.

[20] D. Hu and L. K. Saul, "A probabilistic topic model for unsupervised learning of musical key-profiles," in *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe International Conference Center, Kobe, Japan, October 26-30, 2009*, K. Hirata, G. Tzanetakis, and K. Yoshii, Eds. International Society for Music Information Retrieval, 2009, pp. 441–446.

[21] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts / London, England: The MIT Press, 2006.

[22] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, "Kernels for Vector-Valued Functions: A Review," *Foundations and Trends® in Machine Learning*, vol. 4, no. 3, pp. 195–266, 2012.

[23] D. Milios, R. Camoriano, P. Michiardi, L. Rosasco, and M. Filippone, "Dirichlet-based Gaussian processes for large-scale calibrated classification," in *Advances in Neural Information Processing Systems*, 2018, pp. 6005–6015.

[24] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner,

L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

A. HIERARCHICAL CLUSTERING (DIFFERENT COLOURMAPS)

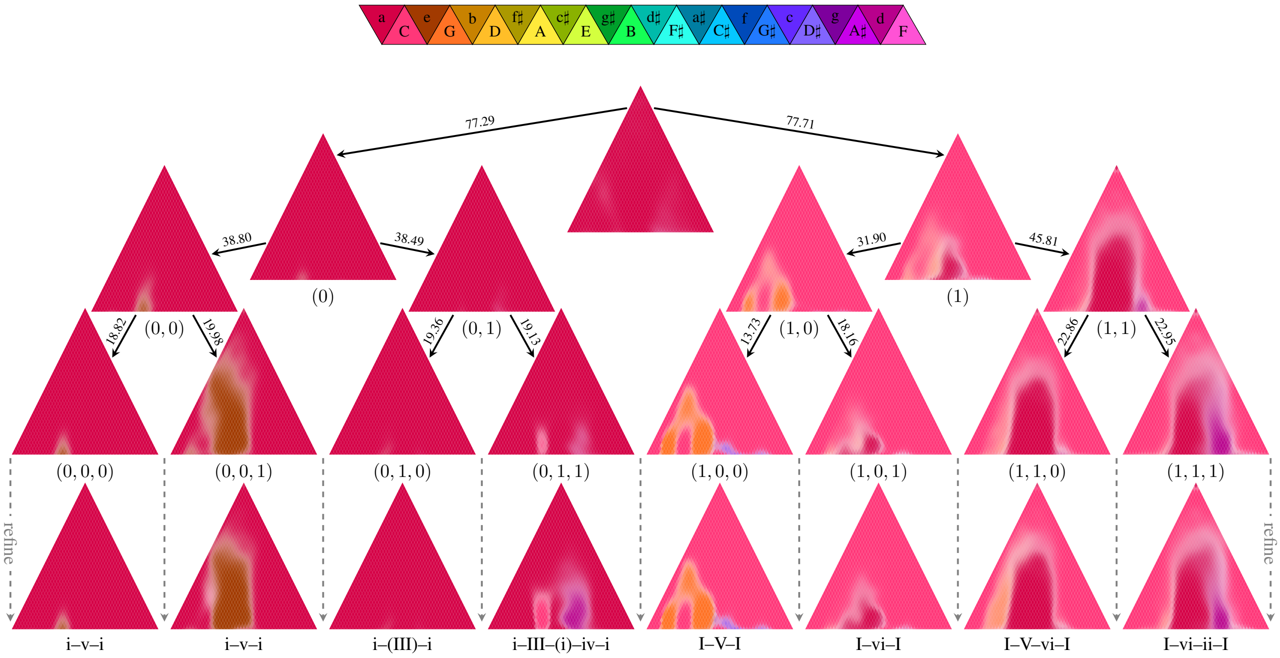


Figure 5. Version of Figure 4 with colouring along the circle of fifths using 'Albrecht' [14] templates.

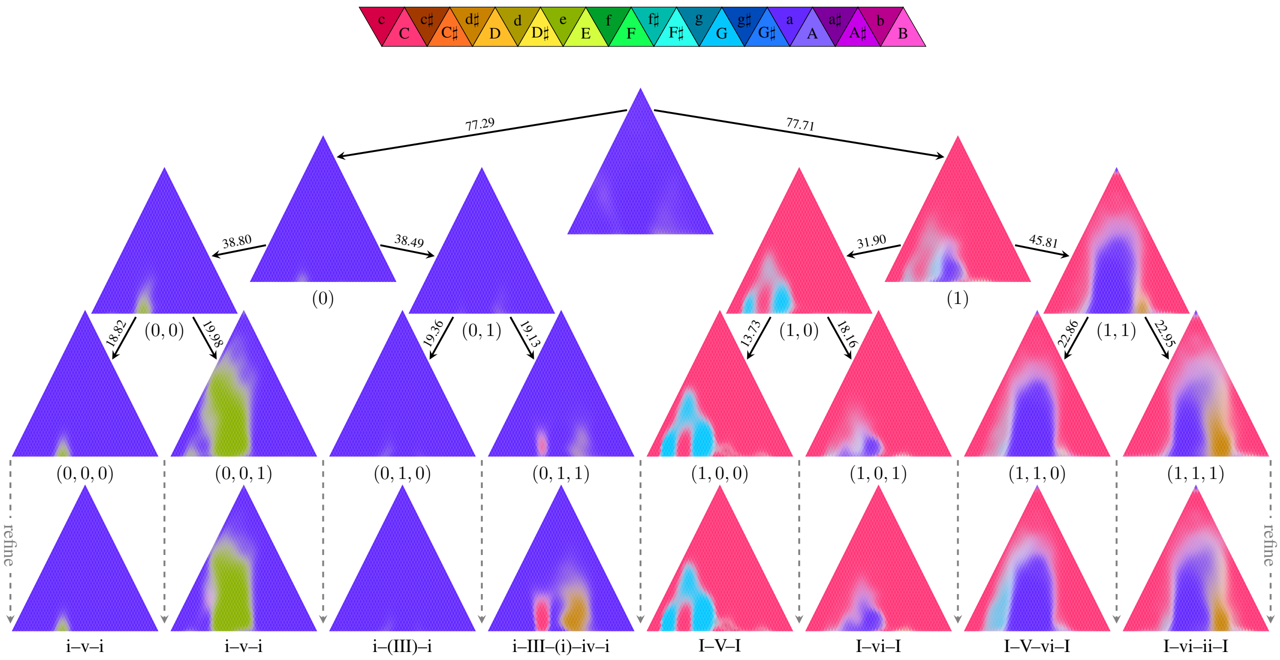
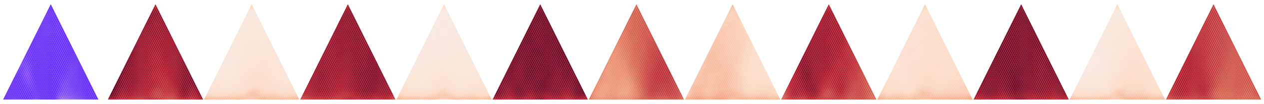
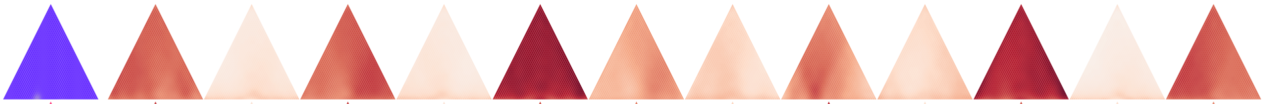
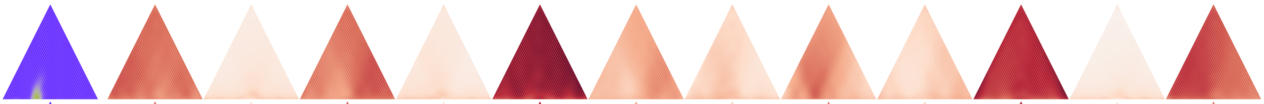
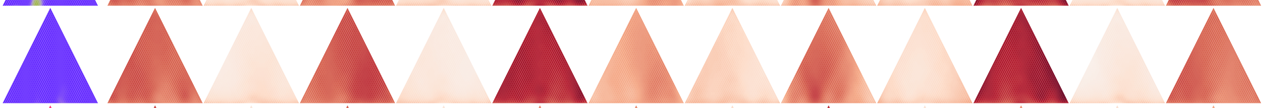
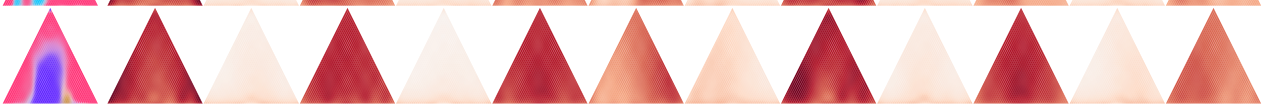
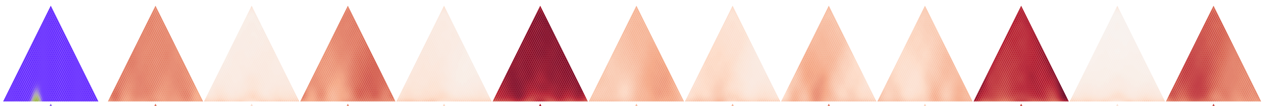
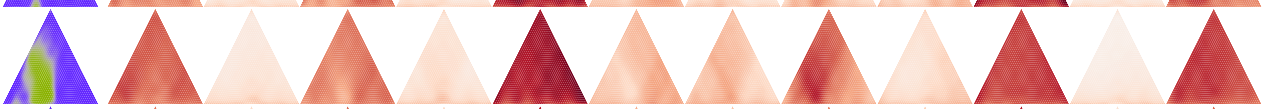
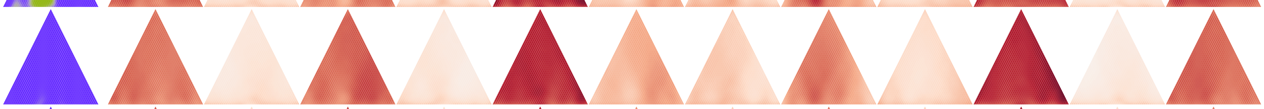
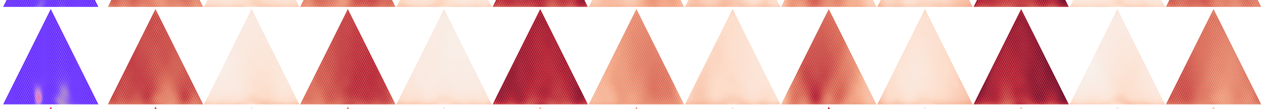
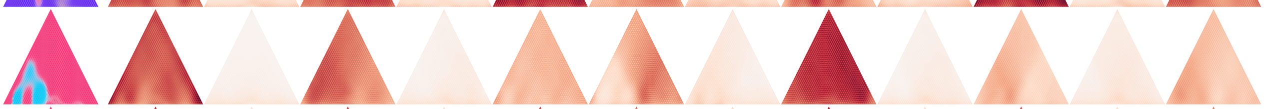
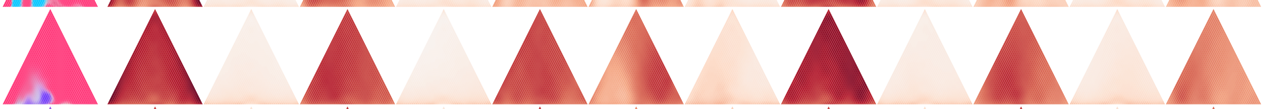
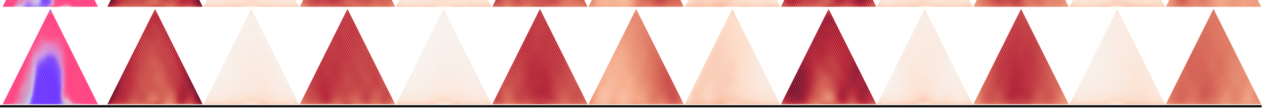


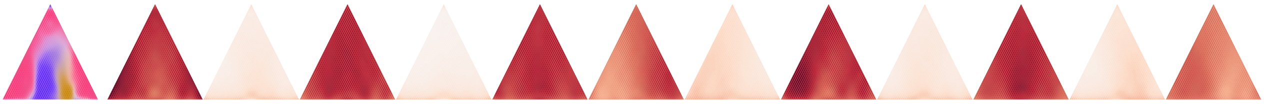
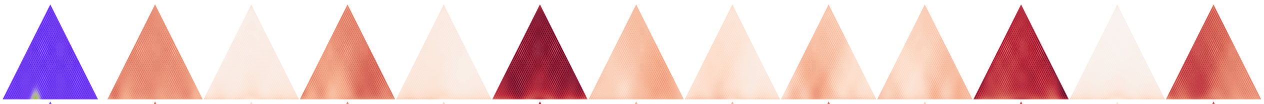
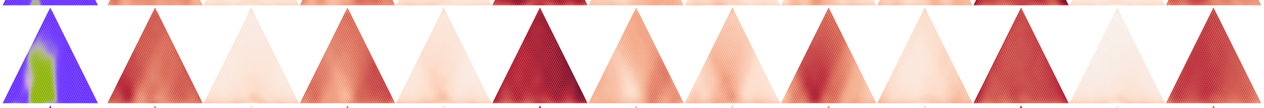
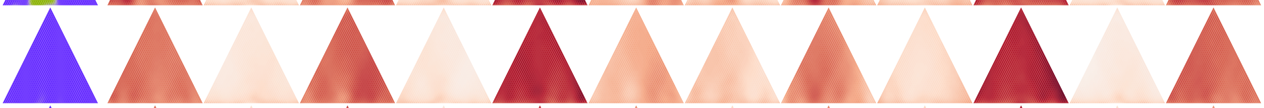
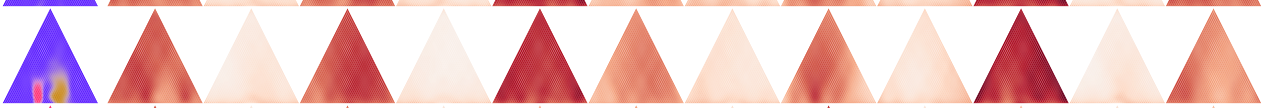
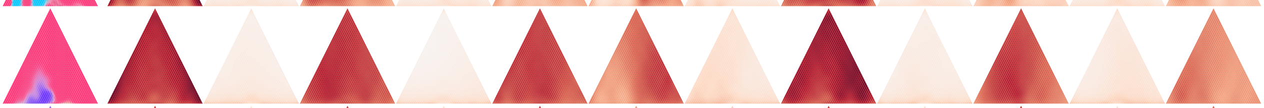
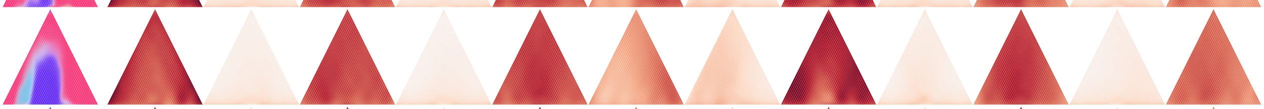
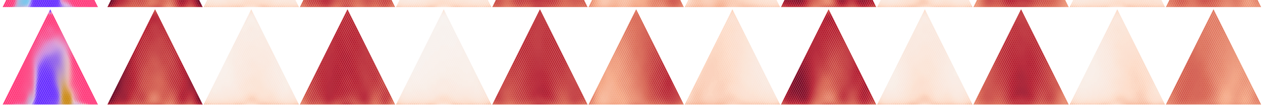
Figure 6. Version of Figure 4 with chromatic colouring using 'Albrecht' [14] templates.

B. SEPARATE PITCH-SCAPE PLOTS FOR CLUSTERS

Table 1: Separate Pitch-Scape Plots

	Separate Pitch-Scape Plots												
	Key Scape	C	C♯	D	D♯	E	F	F♯	G	G♯	A	A♯	B
root													
(0)													
(1)													
(0, 0)													
(0, 1)													
(1, 0)													
(1, 1)													
(0, 0, 0)													
(0, 0, 1)													
(0, 1, 0)													
(0, 1, 1)													
(1, 0, 0)													
(1, 0, 1)													
(1, 1, 0)													

(continued on next page)

Separate Pitch-Scape Plots (continued)												
	Key Scape	C	C♯	D	D♯	E	F	F♯	G	G♯	A	A♯	B
(1, 1, 1)													
(0, 0, 0) refined													
(0, 0, 1) refined													
(0, 1, 0) refined													
(0, 1, 1) refined													
(1, 0, 0) refined													
(1, 0, 1) refined													
(1, 1, 0) refined													
(1, 1, 1) refined													

End of Table 1

C. PIECES

Table 2: Corpus of Baroque pieces

Index	Composer	Title	Catalogue	Work Group
0	Johann Sebastian Bach	Prelude No. 1 in C major	BWV 846	Wohltemperiertes Klavier I
1	Johann Sebastian Bach	Fugue No. 1 in C major	BWV 846	Wohltemperiertes Klavier I
2	Johann Sebastian Bach	Prelude No. 2 in C minor	BWV 847	Wohltemperiertes Klavier I
3	Johann Sebastian Bach	Fugue No. 2 in C minor	BWV 847	Wohltemperiertes Klavier I
4	Johann Sebastian Bach	Prelude No. 3 in C♯ major	BWV 848	Wohltemperiertes Klavier I
5	Johann Sebastian Bach	Fugue No. 3 in C♯ major	BWV 848	Wohltemperiertes Klavier I
6	Johann Sebastian Bach	Prelude No. 4 in C♯ minor	BWV 849	Wohltemperiertes Klavier I
7	Johann Sebastian Bach	Fugue No. 4 in C♯ minor	BWV 849	Wohltemperiertes Klavier I
8	Johann Sebastian Bach	Prelude No. 5 in D major	BWV 850	Wohltemperiertes Klavier I
9	Johann Sebastian Bach	Fugue No. 5 in D major	BWV 850	Wohltemperiertes Klavier I
10	Johann Sebastian Bach	Prelude No. 6 in D minor	BWV 851	Wohltemperiertes Klavier I
11	Johann Sebastian Bach	Fugue No. 6 in D minor	BWV 851	Wohltemperiertes Klavier I
12	Johann Sebastian Bach	Prelude No. 7 in E♭ major	BWV 852	Wohltemperiertes Klavier I
13	Johann Sebastian Bach	Fugue No. 7 in E♭ major	BWV 852	Wohltemperiertes Klavier I
14	Johann Sebastian Bach	Prelude No. 8 in E♭ minor	BWV 853	Wohltemperiertes Klavier I
15	Johann Sebastian Bach	Fugue No. 8 in E♭ minor	BWV 853	Wohltemperiertes Klavier I
16	Johann Sebastian Bach	Prelude No. 9 in E major	BWV 854	Wohltemperiertes Klavier I
17	Johann Sebastian Bach	Fugue No. 9 in E major	BWV 854	Wohltemperiertes Klavier I
18	Johann Sebastian Bach	Prelude No. 10 in E minor	BWV 855	Wohltemperiertes Klavier I
19	Johann Sebastian Bach	Fugue No. 10 in E minor	BWV 855	Wohltemperiertes Klavier I
20	Johann Sebastian Bach	Prelude No. 11 in F major	BWV 856	Wohltemperiertes Klavier I
21	Johann Sebastian Bach	Fugue No. 11 in F major	BWV 856	Wohltemperiertes Klavier I
22	Johann Sebastian Bach	Prelude No. 12 in F minor	BWV 857	Wohltemperiertes Klavier I
23	Johann Sebastian Bach	Fugue No. 12 in F minor	BWV 857	Wohltemperiertes Klavier I
24	Johann Sebastian Bach	Prelude No. 13 in F♯ major	BWV 858	Wohltemperiertes Klavier I
25	Johann Sebastian Bach	Fugue No. 13 in F♯ major	BWV 858	Wohltemperiertes Klavier I
26	Johann Sebastian Bach	Prelude No. 14 in F♯ minor	BWV 859	Wohltemperiertes Klavier I
27	Johann Sebastian Bach	Fugue No. 14 in F♯ minor	BWV 859	Wohltemperiertes Klavier I
28	Johann Sebastian Bach	Prelude No. 15 in G major	BWV 860	Wohltemperiertes Klavier I
29	Johann Sebastian Bach	Fugue No. 15 in G major	BWV 860	Wohltemperiertes Klavier I
30	Johann Sebastian Bach	Prelude No. 16 in G minor	BWV 861	Wohltemperiertes Klavier I
31	Johann Sebastian Bach	Fugue No. 16 in G minor	BWV 861	Wohltemperiertes Klavier I
32	Johann Sebastian Bach	Prelude No. 17 in A♭ major	BWV 862	Wohltemperiertes Klavier I
33	Johann Sebastian Bach	Fugue No. 17 in A♭ major	BWV 862	Wohltemperiertes Klavier I
34	Johann Sebastian Bach	Prelude No. 18 in G♯ minor	BWV 863	Wohltemperiertes Klavier I
35	Johann Sebastian Bach	Fugue No. 18 in G♯ minor	BWV 863	Wohltemperiertes Klavier I
36	Johann Sebastian Bach	Prelude No. 19 in A major	BWV 864	Wohltemperiertes Klavier I
37	Johann Sebastian Bach	Fugue No. 19 in A major	BWV 864	Wohltemperiertes Klavier I
38	Johann Sebastian Bach	Prelude No. 20 in A minor	BWV 865	Wohltemperiertes Klavier I
39	Johann Sebastian Bach	Fugue No. 20 in A minor	BWV 865	Wohltemperiertes Klavier I
40	Johann Sebastian Bach	Prelude No. 21 in B♭ major	BWV 866	Wohltemperiertes Klavier I
41	Johann Sebastian Bach	Fugue No. 21 in B♭ major	BWV 866	Wohltemperiertes Klavier I
42	Johann Sebastian Bach	Prelude No. 22 in B♭ minor	BWV 867	Wohltemperiertes Klavier I
43	Johann Sebastian Bach	Fugue No. 22 in B♭ minor	BWV 867	Wohltemperiertes Klavier I
44	Johann Sebastian Bach	Prelude No. 23 in B major	BWV 868	Wohltemperiertes Klavier I
45	Johann Sebastian Bach	Fugue No. 23 in B major	BWV 868	Wohltemperiertes Klavier I
46	Johann Sebastian Bach	Prelude No. 24 in B minor	BWV 869	Wohltemperiertes Klavier I
47	Johann Sebastian Bach	Fugue No. 24 in B minor	BWV 869	Wohltemperiertes Klavier I
48	Johann Sebastian Bach	Prelude No. 1 in C major	BWV 870	Wohltemperiertes Klavier II
49	Johann Sebastian Bach	Fugue No. 1 in C major	BWV 870	Wohltemperiertes Klavier II
50	Johann Sebastian Bach	Prelude No. 2 in C minor	BWV 871	Wohltemperiertes Klavier II
51	Johann Sebastian Bach	Fugue No. 2 in C minor	BWV 871	Wohltemperiertes Klavier II
52	Johann Sebastian Bach	Prelude No. 3 in C♯ major	BWV 872	Wohltemperiertes Klavier II
53	Johann Sebastian Bach	Fugue No. 3 in C♯ major	BWV 872	Wohltemperiertes Klavier II
54	Johann Sebastian Bach	Prelude No. 4 in C♯ minor	BWV 873	Wohltemperiertes Klavier II
55	Johann Sebastian Bach	Fugue No. 4 in C♯ minor	BWV 873	Wohltemperiertes Klavier II
56	Johann Sebastian Bach	Prelude No. 5 in D major	BWV 874	Wohltemperiertes Klavier II
57	Johann Sebastian Bach	Fugue No. 5 in D major	BWV 874	Wohltemperiertes Klavier II
58	Johann Sebastian Bach	Prelude No. 6 in D minor	BWV 875	Wohltemperiertes Klavier II
59	Johann Sebastian Bach	Fugue No. 6 in D minor	BWV 875	Wohltemperiertes Klavier II
60	Johann Sebastian Bach	Prelude No. 7 in E♭ major	BWV 876	Wohltemperiertes Klavier II
61	Johann Sebastian Bach	Fugue No. 7 in E♭ major	BWV 876	Wohltemperiertes Klavier II
62	Johann Sebastian Bach	Prelude No. 8 in D♯ minor	BWV 877	Wohltemperiertes Klavier II
63	Johann Sebastian Bach	Fugue No. 8 in D♯ minor	BWV 877	Wohltemperiertes Klavier II

(continued on next page)

Index	Composer	Title	Catalogue	Work Group
64	Johann Sebastian Bach	Prelude No. 9 in E major	BWV 878	Wohltemperiertes Klavier II
65	Johann Sebastian Bach	Fugue No. 9 in E major	BWV 878	Wohltemperiertes Klavier II
66	Johann Sebastian Bach	Prelude No. 10 in E minor	BWV 879	Wohltemperiertes Klavier II
67	Johann Sebastian Bach	Fugue No. 10 in E minor	BWV 879	Wohltemperiertes Klavier II
68	Johann Sebastian Bach	Prelude No. 11 in F major	BWV 880	Wohltemperiertes Klavier II
69	Johann Sebastian Bach	Fugue No. 11 in F major	BWV 880	Wohltemperiertes Klavier II
70	Johann Sebastian Bach	Prelude No. 12 in F minor	BWV 881	Wohltemperiertes Klavier II
71	Johann Sebastian Bach	Fugue No. 12 in F minor	BWV 881	Wohltemperiertes Klavier II
72	Johann Sebastian Bach	Prelude No. 13 in F♯ major	BWV 882	Wohltemperiertes Klavier II
73	Johann Sebastian Bach	Fugue No. 13 in F♯ major	BWV 882	Wohltemperiertes Klavier II
74	Johann Sebastian Bach	Prelude No. 14 in F♯ minor	BWV 883	Wohltemperiertes Klavier II
75	Johann Sebastian Bach	Fugue No. 14 in F♯ minor	BWV 883	Wohltemperiertes Klavier II
76	Johann Sebastian Bach	Prelude No. 15 in G major	BWV 884	Wohltemperiertes Klavier II
77	Johann Sebastian Bach	Fugue No. 15 in G major	BWV 884	Wohltemperiertes Klavier II
78	Johann Sebastian Bach	Prelude No. 16 in G minor	BWV 885	Wohltemperiertes Klavier II
79	Johann Sebastian Bach	Fugue No. 16 in G minor	BWV 885	Wohltemperiertes Klavier II
80	Johann Sebastian Bach	Prelude No. 17 in A♭ major	BWV 886	Wohltemperiertes Klavier II
81	Johann Sebastian Bach	Fugue No. 17 in A♭ major	BWV 886	Wohltemperiertes Klavier II
82	Johann Sebastian Bach	Prelude No. 18 in G♯ minor	BWV 887	Wohltemperiertes Klavier II
83	Johann Sebastian Bach	Fugue No. 18 in G♯ minor	BWV 887	Wohltemperiertes Klavier II
84	Johann Sebastian Bach	Prelude No. 19 in A major	BWV 888	Wohltemperiertes Klavier II
85	Johann Sebastian Bach	Fugue No. 19 in A major	BWV 888	Wohltemperiertes Klavier II
86	Johann Sebastian Bach	Prelude No. 20 in A minor	BWV 889	Wohltemperiertes Klavier II
87	Johann Sebastian Bach	Fugue No. 20 in A minor	BWV 889	Wohltemperiertes Klavier II
88	Johann Sebastian Bach	Prelude No. 21 in B♭ major	BWV 890	Wohltemperiertes Klavier II
89	Johann Sebastian Bach	Fugue No. 21 in B♭ major	BWV 890	Wohltemperiertes Klavier II
90	Johann Sebastian Bach	Prelude No. 22 in B♭ minor	BWV 891	Wohltemperiertes Klavier II
91	Johann Sebastian Bach	Fugue No. 22 in B♭ minor	BWV 891	Wohltemperiertes Klavier II
92	Johann Sebastian Bach	Prelude No. 23 in B major	BWV 892	Wohltemperiertes Klavier II
93	Johann Sebastian Bach	Fugue No. 23 in B major	BWV 892	Wohltemperiertes Klavier II
94	Johann Sebastian Bach	Prelude No. 24 in B minor	BWV 893	Wohltemperiertes Klavier II
95	Johann Sebastian Bach	Fugue No. 24 in B minor	BWV 893	Wohltemperiertes Klavier II
96	Johann Sebastian Bach	Two-Part Invention No. 1 in C major	BWV 772	Inventions and Sinfonias
97	Johann Sebastian Bach	Two-Part Invention in C major	BWV 772a	Inventions and Sinfonias
98	Johann Sebastian Bach	Two-Part Invention No. 2 in C minor	BWV 773	Inventions and Sinfonias
99	Johann Sebastian Bach	Two-Part Invention No. 3 in D major	BWV 774	Inventions and Sinfonias
100	Johann Sebastian Bach	Two-Part Invention No. 4 in D minor	BWV 775	Inventions and Sinfonias
101	Johann Sebastian Bach	Two-Part Invention No. 5 in E♭ major	BWV 776	Inventions and Sinfonias
102	Johann Sebastian Bach	Two-Part Invention No. 6 in E major	BWV 777	Inventions and Sinfonias
103	Johann Sebastian Bach	Two-Part Invention No. 7 in E minor	BWV 778	Inventions and Sinfonias
104	Johann Sebastian Bach	Two-Part Invention No. 8 in F major	BWV 779	Inventions and Sinfonias
105	Johann Sebastian Bach	Two-Part Invention No. 9 in F minor	BWV 780	Inventions and Sinfonias
106	Johann Sebastian Bach	Two-Part Invention No. 10 in G major	BWV 781	Inventions and Sinfonias
107	Johann Sebastian Bach	Two-Part Invention No. 11 in G minor	BWV 782	Inventions and Sinfonias
108	Johann Sebastian Bach	Two-Part Invention No. 12 in A major	BWV 783	Inventions and Sinfonias
109	Johann Sebastian Bach	Two-Part Invention No. 13 in A minor	BWV 784	Inventions and Sinfonias
110	Johann Sebastian Bach	Two-Part Invention No. 14 in B♭ major	BWV 785	Inventions and Sinfonias
111	Johann Sebastian Bach	Two-Part Invention No. 15 in B minor	BWV 786	Inventions and Sinfonias
112	Johann Sebastian Bach	Sinfonia No. 1 in C major	BWV 787	Inventions and Sinfonias
113	Johann Sebastian Bach	Sinfonia No. 2 in C minor	BWV 788	Inventions and Sinfonias
114	Johann Sebastian Bach	Sinfonia No. 3 in D major	BWV 789	Inventions and Sinfonias
115	Johann Sebastian Bach	Sinfonia No. 4 in D minor	BWV 790	Inventions and Sinfonias
116	Johann Sebastian Bach	Sinfonia No. 5 in E♭ major	BWV 791	Inventions and Sinfonias
117	Johann Sebastian Bach	Sinfonia No. 6 in E major	BWV 792	Inventions and Sinfonias
118	Johann Sebastian Bach	Sinfonia No. 7 in E minor	BWV 793	Inventions and Sinfonias
119	Johann Sebastian Bach	Sinfonia No. 8 in F major	BWV 794	Inventions and Sinfonias
120	Johann Sebastian Bach	Sinfonia No. 9 in F minor	BWV 795	Inventions and Sinfonias
121	Johann Sebastian Bach	Sinfonia No. 10 in G major	BWV 796	Inventions and Sinfonias
122	Johann Sebastian Bach	Sinfonia No. 11 in G minor	BWV 797	Inventions and Sinfonias
123	Johann Sebastian Bach	Sinfonia No. 12 in A major	BWV 798	Inventions and Sinfonias
124	Johann Sebastian Bach	Sinfonia No. 13 in A minor	BWV 799	Inventions and Sinfonias
125	Johann Sebastian Bach	Sinfonia No. 14 in B♭ major	BWV 800	Inventions and Sinfonias
126	Johann Sebastian Bach	Sinfonia No. 15 in B minor	BWV 801	Inventions and Sinfonias
127	Johann Sebastian Bach	No. 5 I Allegro	BWV 1050	Brandenburgische Konzerte
128	Johann Sebastian Bach	No. 5 II Affettuoso	BWV 1050	Brandenburgische Konzerte
129	Johann Sebastian Bach	No. 5 III Allegro	BWV 1050	Brandenburgische Konzerte
130	Georg Friedrich Händel	The ways of Zion do mourn	HWV 264	Funeral Anthem for Queen Caroline
131	Georg Friedrich Händel	She put on righteousness	HWV 264	Funeral Anthem for Queen Caroline

(continued on next page)

Index	Composer	Title	Catalogue	Work Group
132	Georg Friedrich Händel	The righteous shall be	HWV 264	Funeral Anthem for Queen Caroline
133	Georg Friedrich Händel	Their bodies are buried	HWV 264	Funeral Anthem for Queen Caroline
134	Georg Friedrich Händel	The people will tell	HWV 264	Funeral Anthem for Queen Caroline
135	Georg Friedrich Händel	The mercifull goodness	HWV 264	Funeral Anthem for Queen Caroline
136	Georg Friedrich Händel	Chaconne in G major	HWV 435	Chaconne in G major
137	Domenico Scarlatti	Sonata in F-minor	K 19	Sonata
138	Domenico Scarlatti	Sonata	K 23	Sonata
139	Domenico Scarlatti	Sonata	K 27	Sonata
140	Domenico Scarlatti	Sonata	K 63	Sonata
141	Domenico Scarlatti	Sonata (IMSLP Edition)	K 64	Sonata
142	Domenico Scarlatti	Sonata (PianoXML Edition)	K 64	Sonata
143	Domenico Scarlatti	Sonata in B-minor	K 87	Sonata
144	Domenico Scarlatti	Sonata in C-minor Allegro	K 116	Sonata
145	Domenico Scarlatti	Sonata	K 138	Sonata
146	Domenico Scarlatti	Sonata	K 159	Sonata
147	Domenico Scarlatti	Sonata in E-major	K 162	Sonata
148	Domenico Scarlatti	Sonata in A-major	K 208	Sonata
149	Domenico Scarlatti	Sonata in A-minor	K 223	Sonata
150	Domenico Scarlatti	Sonata	K 431	Sonata
151	Domenico Scarlatti	Sonata in G-major	K 455	Sonata
152	Domenico Scarlatti	Sonata	K 526	Sonata
153	Domenico Scarlatti	Sonata	K 531	Sonata
154	Domenico Scarlatti	Sonata in E-major	L 63	Sonata

End of Table 2

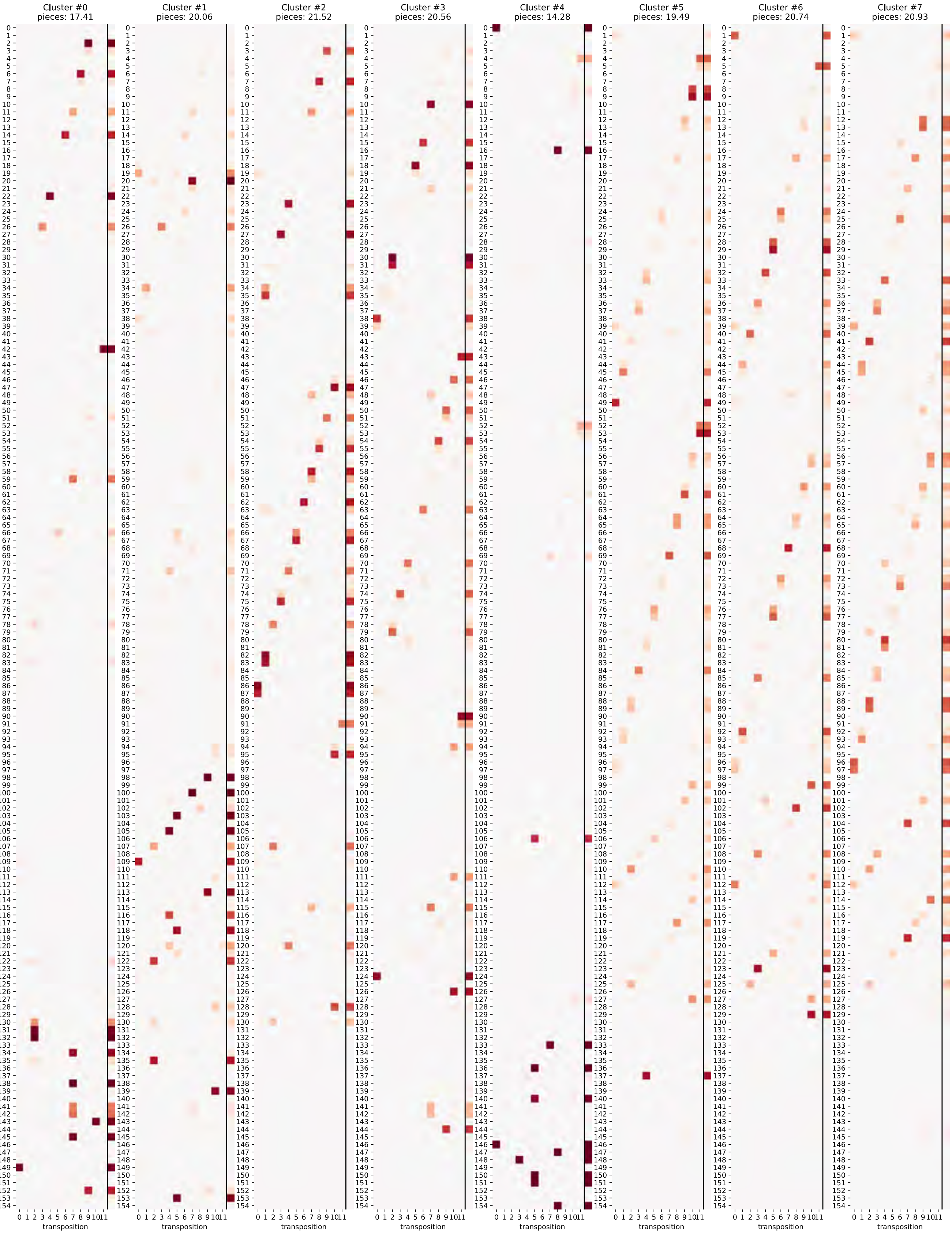


Figure 7. Piece assignments $p(c, \tau \mid d)$ after refinement. The piece index on the left of each sub-plot corresponds to those in Table 2. Each sub-plot corresponds to one cluster c (same order as in Figure 4), each column in a sub-plot (except for the right-most) corresponds to a specific transposition τ (these values are normalised over c and τ per piece). The right-most column in each sub-plot (separated by a vertical black line) represents the marginal cluster probability $p(c \mid d)$ for each piece (these values are normalised over c only as τ was marginalised out). On the top, for each cluster the number of pieces falling into that cluster $\sum_d p(c \mid d)$ is indicated.