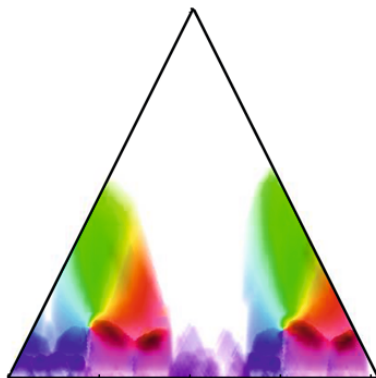


Chapter 4

Music Structure Analysis



One of the attributes distinguishing music from random sound sources is the hierarchical structure in which music is organized. At the lowest level, one has events such as individual notes, which are characterized by the way they sound, their timbre, pitch, and duration. Combining various sound events, one obtains larger structures such as motifs, phrases, and sections, and these structures again form larger constructs that determine the overall layout of the composition. This higher structural level is also referred to as the musical structure of the piece, which is specified in terms of musical parts and their mutual relations. For example, in popular music such parts can be the intro, the chorus, and the verse sections of the song. Or in classical music, they can be the exposition, the development, and the recapitulation of a movement. The general goal of **music structure analysis** is to divide a given music representation into temporal segments that correspond to musical parts and to group these segments into musically meaningful categories.

Let us consider a concrete example. Figure 4.1a shows a sheet music representation of the Mazurka Op. 6, No. 4 by the Polish composer Frédéric Chopin. This piano piece can be subdivided into five sections, where the third and fifth sections are repetitions of the first section. Therefore, these sections belong to the same category denoted by the symbol A . Similarly, the fourth section is a repetition of the second one. These two sections belong to another group labeled by the symbol B . Hence, at an abstract level, the overall musical structure can be described by the sequence $A_1B_1A_2B_2A_3$ (see Figure 4.1d). Instead of using the musical score, one typical scenario is to derive structural information from a given audio recording

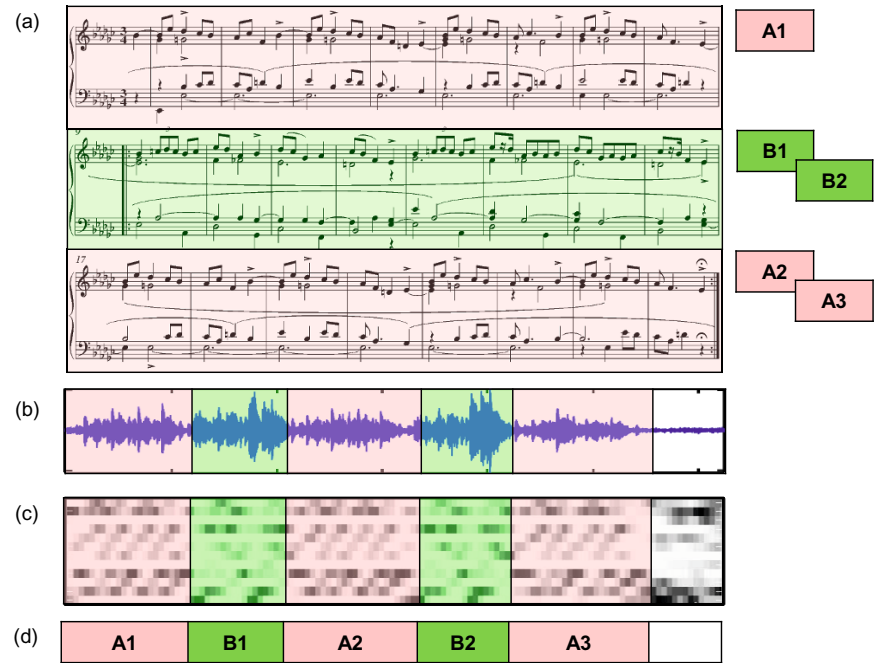


Fig. 4.1 Musical structure of the Mazurka Op. 6, No. 4 by Chopin. **(a)** Sheet music representation. **(b)** Waveform of an audio recording. **(c)** Chroma representation derived from (b). **(d)** Manually annotated segmentation of the audio recording.

(see Figure 4.1b). To this end, one needs to convert the waveform into a suitable feature representation that captures musical properties relevant for the structure of interest. In our example, as shown by Figure 4.1c, the repetition-based structure can be seen in a chroma representation that captures harmonic information.

As demonstrated by the previous example, the musical structure is often related to recurring patterns such as repeating sections. In general, however, there are many more criteria for segmenting and structuring music. For example, certain musical sections may be characterized by some homogeneity property such as a consistent timbre, the presence of a specific instrument, or the usage of certain harmonies. Furthermore, segment boundaries may go along with sudden changes in musical properties such as tempo, dynamics, or the musical key. These various segmentation principles require different methods, which may be loosely categorized into repetition-based, homogeneity-based, and novelty-based approaches.

In this chapter, we study general techniques for deriving structural information from a given music recording. In Section 4.1, we start by giving an overview of different segmentation principles, while introducing a working definition of the structure analysis problem as used in the subsequent sections. Furthermore, we discuss some feature representations that account for different musical dimensions. The con-

cept of self-similarity matrices, which we study in Section 4.2, is of fundamental importance in computational music structure. In particular, we show how the various segmentation principles are reflected in such matrices and how this can be exploited for deriving structural information. As a first application of self-similarity matrices, we discuss in Section 4.3 a subproblem of music structure analysis known as audio thumbnailing. The goal of this problem is to determine the audio segment that best represents a given music recording. Providing a compact preview, such audio segments are useful for music navigation applications similar to visual thumbnails that help in organizing and accessing large photo collections. While we apply repetition-based principles for audio thumbnailing, we discuss in Section 4.4 some segmentation procedures that rely on novelty-based principles. The objective of such procedures is to specify points within a given audio recording where a human listener would recognize a change, a sudden event, or the transition between two contrasting parts. Finally, in Section 4.5, we address the issue of evaluating analysis results, which itself constitutes a nontrivial problem.

4.1 General Principles

Music structure analysis is a multifaceted and often ill-defined problem that depends on many different aspects. First of all, the complexity of the problem depends on the kind of music representation to be analyzed. For example, while it is comparatively easy to detect certain structures such as repeating melodies in sheet music, it is often much harder to automatically identify such structures in audio representations. Second, there are various principles including homogeneity, repetition, and novelty that a segmentation may be based on. While the musical structure of the piano piece shown in Figure 4.1 is based on repetition, musical parts in other music may be characterized by a certain instrumentation or tempo. Third, one also has to account for different musical dimensions, such as melody, harmony, rhythm, or timbre. For example, in Beethoven's Fifth Symphony the "fate motif" is repeated in various ways—sometimes the motif is shifted in pitch; sometimes only the rhythmic pattern is preserved. Finally, the segmentation and structure largely depend on the musical context and the temporal hierarchy to be considered. For example, the recapitulation of a sonata may be considered a kind of repetition of the exposition on a coarse temporal level even though there may be significant modifications in melody and harmony on a finer temporal level. Figure 4.2 gives an overview of various aspects that need to be considered when dealing with musical structures. In the following, we discuss these aspects in more detail. In particular, our goal is to raise the awareness that computational procedures as described in the subsequent sections are often based on simplifying model assumptions that only reflect certain aspects of the complex structural properties of music.

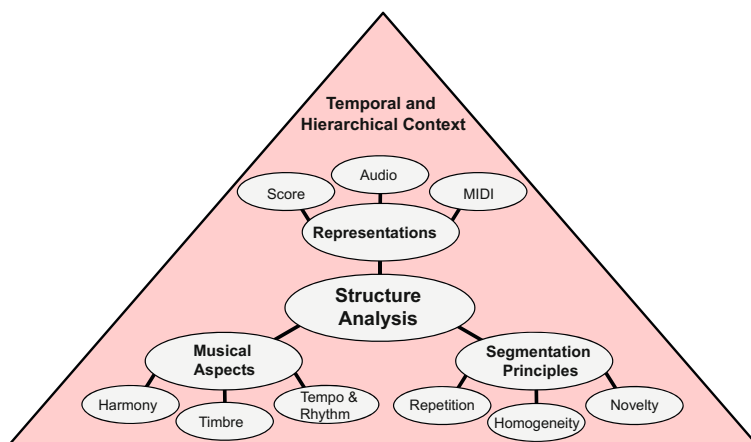


Fig. 4.2 Overview of various segmentation and structure principles.

4.1.1 Segmentation and Structure Analysis

The tasks of segmenting and structuring multimedia documents are of fundamental importance not only for the processing of music signals but also for general audio-visual content. **Segmentation** typically refers to the process of partitioning a given document into multiple segments with the goal of simplifying the representation into something that is more meaningful and easier to analyze than the original document. For example, in image processing the goal is to partition a given image into a set of regions such that each region is similar with respect to some characteristic such as color, intensity, or texture (see Figure 4.3 for an illustration). Region boundaries can often be described by contour lines or edges at which the image brightness or other properties change sharply and reveal discontinuities. In music, the segmentation task is to decompose a given audio stream into acoustically meaningful sections each corresponding to a continuous time interval that is specified by a start and end boundary. At a fine level, the segmentation may aim to find the boundaries between individual notes or to find the beat intervals specified by beat positions. At a coarser level, the goal may be to detect changes in instrumentation or harmony or to find the boundaries between verse and chorus sections. Also, discriminating between silence, speech, and music, finding the actual beginning of a music recording, or separating the applause at the end of a performance are typical segmentation tasks.

Going beyond mere segmentation, the goal of **structure analysis** is to also find and understand the relationships between the segments. For example, certain segments may be characterized by the instrumentation. There may be sections played only by strings. Sections played by the full orchestra may be followed by solo sections. The verse sections with a singing voice may be alternated with purely instrumental sections. Or a soft and slow introductory section may precede the main theme played in a much faster tempo. Furthermore, sections are often repeated. Most

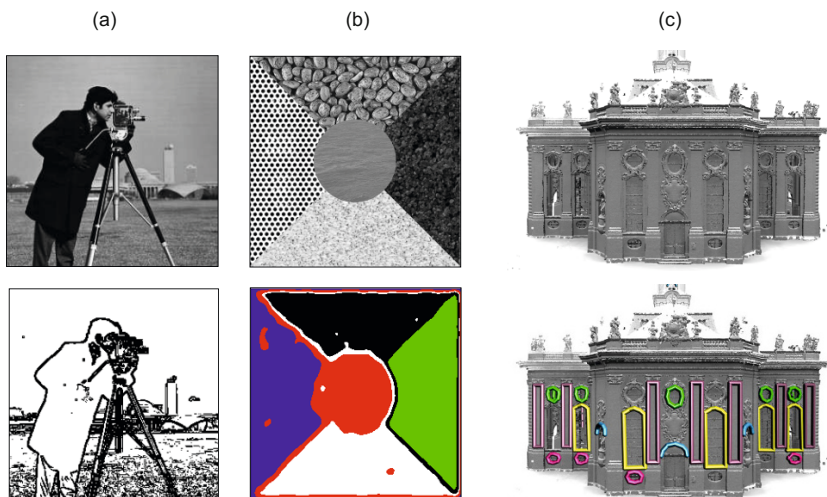


Fig. 4.3 Examples for segmentation results for image and 3D data. **(a)** Novelty-based image segmentation using edge detection. **(b)** Homogeneity-based texture segmentation. **(c)** Repetition-based segmentation of 3D geometry (from [66]).

events of musical relevance are repeated in a musical work in one way or another. However, repetitions are rarely identical copies of the original section, but undergo modifications in aspects such as the lyrics, the instrumentation, or the melody. One main task of structure analysis is to not only segment the given music recording, but to also group the segments into musically meaningful categories (e.g., intro, chorus, verse, outro).

The challenge in computational music structure analysis is that structure in music arises from many different kinds of relationships including repetition, contrast, variation, and homogeneity [53]. As we have already noted, **repetitions** play a particularly important role in music, where sounds or sequences of notes are often repeated [39]. Recurrent patterns can be of rhythmic, harmonic, or melodic nature. On the other hand, **contrast** is the difference between successive musical sections of different character. For example, a quiet passage may be contrasted by a loud one, a slow section by a rapid one, or an orchestral part by a solo. A further principle is that of **variation**, where motifs and parts are picked up again in a modified or transformed form. Finally, a section is often characterized by some sort of inherent **homogeneity**; for example, the instrumentation, the tempo, or the harmonic material may be similar within the section. All these principles need to be considered in the temporal context. Music happens in time (as opposed to, say, a painting), and it is the **temporal order** of events that is essential for building up musically and perceptually meaningful entities such as melodies or harmonic progressions [3].

In view of the various principles that crucially influence the musical structure, a large number of different approaches to music structure analysis have been developed. In this chapter, we want to roughly distinguish three different classes of

methods. First, **repetition-based** methods are used to identify recurring patterns. Second, **novelty-based** methods are employed to detect transitions between contrasting parts. Third, **homogeneity-based** methods are used to determine passages that are consistent with respect to some musical property. Note that novelty-based and homogeneity-based approaches are two sides of a coin: novelty detection is based on observing some surprising event or change after a more homogeneous segment. While the aim of novelty detection is to locate the changes' time positions, the focus of homogeneity analysis lies in the identification of longer passages that are coherent with respect to some musical property. In the following section, we will study various procedures for structure analysis following one or several of these paradigms.

4.1.2 Musical Structure

As already mentioned in the introduction of this chapter, our focus is to analyze a given music recording on a rather coarse structural level. This level corresponds to what is often referred to as the **musical structure**, which describes the overall structural layout of a piece of music. In particular for Western classical music, one also encounters the term **musical form**, which refers to specific structural categories exploiting the principles of contrast and variety in one way or another. In this chapter, we use the term “musical structure” loosely, including with it the concept of musical form.

To specify musical structures, we now introduce some terminology as used in the remainder of this book. First of all, we want to distinguish between a piece of music (in an abstract sense) and a particular audio recording (an actual performance) of the piece. The term **part** is used in the context of the abstract music domain, whereas the term **segment** is used for the audio domain. Furthermore, we use the term **section** in a rather vague way for both domains to denote either a segment or a part. Musical parts are typically denoted by the capital letters A, B, C, \dots in the order of their first occurrence, where numbers (often written as subscripts) indicate the order of repeated occurrences. For example, the sequence $A_1B_1A_2B_2A_3$ describes the musical structure of the piano piece shown in Figure 4.1, which consists of three repeating A -parts and two repeating B -parts. Hence, given a recording of this piece of music, the goal of the structure analysis problem (as considered in this chapter) is to find the segments within the recording that correspond to the A - and B -parts.

In Western music, the musical structure often follows certain structural patterns (see Figure 4.4). The simplest of these patterns is the **strophic form**, which basically consists of a sequence of a part being repeated over and over again. The form $A_1A_2A_3A_4\dots$ is, for example, used in folk songs or nursery rhymes, where the A -parts correspond to the stanzas of the underlying poem. Another structural pattern is referred to as **chain form**, which is simply a sequence of self-contained and unrelated parts ($ABCD\dots$), sometimes with repeats ($A_1A_2B_1B_2C_1C_2D_1D_2\dots$). This form is often used in a composition that consists of a concatenation of favorite tunes from

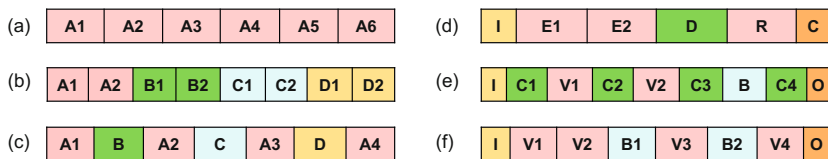


Fig. 4.4 Examples for musical structures as encountered in Western music. (a) Strophic form. (b) Chain form with repetitions. (c) Rondo form. (d) Sonata form. (e) Beatles song “Tell Me Why.” (f) Beatles song “Yesterday.”

popular songs, dances, or operettas. Examples are **medleys** or **potpourris**, which are pieces composed from parts of existing pieces that are simply juxtaposed with no strong connection or relationship. Another form is the **rondo form**, where a recurring theme alternates with contrasting sections, yielding the musical structure $A_1BA_2CA_3DA_4 \dots$

In Western classical music, one of the most important musical structures is known as the **sonata form**, which is a large-scale musical structure typically used in the first movements of sonatas and symphonies. The basic sonata form consists of an **exposition** (E), a **development** (D), and a **recapitulation** (R), where the exposition is repeated once. Sometimes, one can find an additional **introduction** (I) and a closing **coda** (C), thus yielding the form IE_1E_2DRC . In particular, the exposition and the recapitulation stand in close relation to each other, both containing two subsequent contrasting subject groups (often simply referred to as the first and second theme) connected by some transition. As previously noted, at least at a coarse level, the recapitulation can be regarded as a kind of repetition of the exposition. However, at a finer level, there are significant differences. For example, the subject groups and transition in the recapitulation are musically altered and can be quite different from their corresponding occurrences in the exposition. Finally, we want to discuss some typical structural elements one finds in popular music. As with the sonata form, one sometimes uses generic names to denote the musical parts instead of using capital letters. The most important parts of a pop song are the **verse** (V) and the **chorus** (C) sections. Each verse usually employs the same melody (possibly with slight modifications), while the lyrics change for each verse. The chorus (sometimes also called the **refrain**) typically consists of a melodic and lyrical phrase which is repeated. Sometimes, pop songs may start with an **intro** (I) and close with an **outro** (O). Finally, verse and chorus sections may be connected by an additional part called a **bridge** (B). The verse and chorus are usually repeated throughout a song, while the intro and the outro appear only once. Some pop songs may have a **solo** section, where one or more instruments play a melodic line, typically following the melody previously introduced by the singer.

We have presented only a small selection of musical structures. In practice, there are many more structures as well as variations and deviations from standard forms as illustrated by the last two examples of Figure 4.4. A musical structure can be rather vague, and even music experts may argue about the construction of a given compo-

The figure displays a musical score for the violin part of the Hungarian Dance No. 5 by Johannes Brahms. The score is organized into nine distinct sections, each highlighted with a different background color and labeled with a letter and number. The sections are: A1 (pink), A2 (pink), B1 (green), B2 (green), C (light blue), A3 (pink), B3 (green), B4 (green), and D (yellow). The musical notation includes various tempo markings such as 'Allegro', 'Vivace', 'poco ritard', 'in tempo', 'poco rallent', 'marcato', and 'poco rallent'. The score is written in a single staff, and the key signature is one flat (B-flat). The sections are arranged in a non-linear fashion, reflecting the complex structure of the dance.

Fig. 4.5 Sheet music representation and musical structure of the Hungarian Dance No. 5 by Johannes Brahms. Only the voice for the violin of an arrangement for full orchestra is shown.

sition. In particular, what we call a repetition of a musical section is often far from being an exact copy. Segments that are considered to correspond to the same musical part may differ in instrumentation and tempo, or a segment may be transposed to another key, the melody may be changed while only the underlying harmonic progression is kept, and so on. Furthermore, musical structure is typically ordered in hierarchies, and it is often not clear which level should be considered when specifying the musical structure. For example, in the piece shown in Figure 4.1, the *A*-part can be further subdivided into substructures consisting of two or even four subparts. Similarly, the *B*-part can be regarded as a repetition of two subparts. These repeating substructures also become visible in the chroma representation derived from the music recording (see Figure 4.1c). In music notation, such subparts are often indicated using small letters *a*, *b*, *c*, ...

As a final example, we want to consider the Hungarian Dance No. 5 by Johannes Brahms, which will also serve as our running example in the next sections. This piece is part of a set of 21 dance tunes composed by Brahms up to 1869 and based mostly on traditional Hungarian themes. Each dance has been arranged for a wide variety of instruments and ensembles, ranging from piano versions to versions for

full orchestra. Figure 4.5 shows a sheet music representation for the violin voice of an arrangement for full orchestra. The musical structure as indicated in the figure is $A_1A_2B_1B_2CA_3B_3B_4D$, which consists of three repeating A -parts, four repeating B -parts, as well as a C -part and a short closing D -part. The A -part has a substructure consisting of two more or less repeating subparts. Furthermore, as becomes apparent when looking at the musical score, the middle C -part may be further subdivided into a substructure that may be described by $d_1d_2e_1e_2e_3e_4$ (see Figure 4.28).

The overall musical structure of this piece can be explained in terms of repeating elements. However, there are also many other musical cues that reinforce the musical structure. For example, the C -part stands in contrast to the remaining parts. First, there is a change of the musical key in the C -part (changing from G minor to G major). Then, there is a change in the notated tempo (changing from ‘Allegro’ to ‘Vivace’). While the A - and B -parts have catchy tunes, there is no such melody in the C -part. Instead, the entire C -part is rather homogeneous with regard to harmony. However, this does not hold for other musical properties such as dynamics and tempo. For example, while the d -part segments are played in forte, the e -part segments are played in piano. Also there are many sudden tempo changes within the C -part. Therefore, in this case, a novelty-based segmentation procedure using tempo cues may be used to reveal the substructures of the C -part, whereas a homogeneity-based segmentation procedure using harmonic properties may be suited to distinguish the C -part from the other parts. We further develop this example in the next sections.

4.1.3 Musical Dimensions

We have already seen that the applicability of the different segmentation principles very much depends on the musical and acoustic properties of the audio signal to be analyzed. Since the sampled waveform of an audio signal is relatively uninformative by itself, the first step in automated structure analysis is to transform the given music recording into a suitable feature representation. As explained in the music synchronization scenario (Section 3.1), finding such a representation constitutes a delicate trade-off between robustness and expressiveness. Also, it is often unclear which musical properties are actually relevant for the given music signal and the considered segmentation scenario. For example, structural boundaries may be based on changes in harmony, timbre, or tempo. One major task in music processing is to transform a given audio signal into feature representations that correlate to the various musical aspects. In the following, we discuss this issue in more detail by considering three conceptually different feature representations (see Figure 4.6 for an overview).

As a first representation, we consider chroma features as introduced in Section 3.1.2. Recall that a normalized chroma vector describes the signal’s local energy distribution over an analysis window (frame) across the twelve pitch classes of the equal-tempered scale (ignoring octave information). Capturing pitched content, a chroma-based feature sequence relates to harmonic and melodic properties

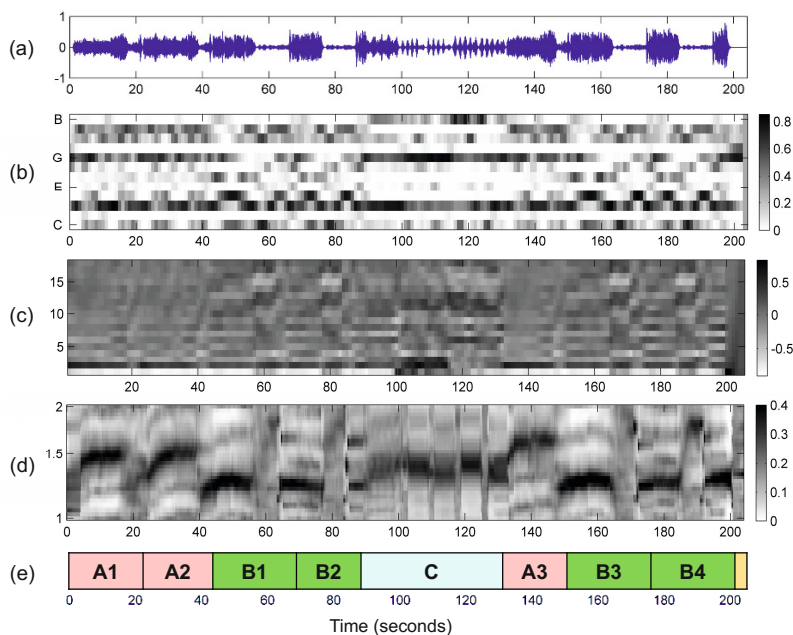


Fig. 4.6 Feature representations for a recording of the Hungarian Dance No. 5 by Johannes Brahms. **(a)** Waveform. **(b)** Chroma-based features. **(c)** MFCC-based features. **(d)** Tempo-based features. **(e)** Manually generated annotation.

of the music recording. Figure 4.6b shows a chroma representation derived from a recorded performance of our Brahms example, the Hungarian Dance No. 5. The patterns visible in the chromagram reveal important structural information. For example, the four repeating *B*-part segments are clearly visible as four similar characteristic subsequences in the chromagram. Furthermore, the *C*-part segment stands out in the chromagram by showing a high degree of homogeneity throughout the entire section. Indeed, for all chroma features of this segment, most of the signal's energy is contained in the G-, B-, and D-bands (which is not surprising since the *C*-part is in G major). In contrast, as for the *A*-part segments, many chroma vectors have dominant entries in the G-, B^b-, and D-bands (which nicely reflects that this part is in G minor).

Besides melody and harmony, the instrumentation and timbral characteristics are of great importance for the human perception of music structure. As we have discussed in Section 1.3.4, timbre is a rather vaguely defined perceptual property of sound, which is hard to describe and to extract from a music recording. For example, the automated recognition of musical instruments within polyphonic music signals is an extremely difficult problem. In applications such as structure analysis, it is often unnecessary to determine such information explicitly. Instead, mid-level representations that somehow correlate to aspects such as instrumentation and timbre

may be sufficient. In the context of timbre-based structure analysis, one often uses **mel-frequency cepstral coefficients** (MFCCs), which were originally developed for automated speech recognition. Parametrizing the rough shape of the spectral envelope, MFCC-based features capture timbral properties of the signal. At this point, we do not want to give a technical description on how these features are computed. Instead, let us have a look at Figure 4.6c, which shows an MFCC-based feature representation for our Brahms example. One can recognize that MFCC features within the *A*-part segments are different from the ones in the *B*-part and *C*-part segments. For many music recordings such as pop songs, where sections with singing voice alternate with purely instrumental or percussive sections, MFCC-based feature representations are well suited for novelty-based and homogeneity-based segmentation.

As a third musical dimension, we consider properties that are related to beat, tempo, and rhythmic information. Estimation of the tempo and beat positions is one of the central topics in music processing, which we cover in Chapter 6. In the music segmentation context, such techniques are often applied to derive **beat-synchronous** feature representations, where the time axis is segmented according to musically meaningful beat positions. Such beat-synchronous representations are very useful to compensate for tempo changes in repeating parts. On the downside, beat tracking errors introduced by automated procedures may have negative consequences for the subsequent music processing tasks to be solved (see Section 6.3.3 for more details).

In music structure analysis, tempo and beat information may also be used in combination with homogeneity-based segmentation approaches. Instead of extracting such information explicitly, a mid-level feature representation that correlates to tempo and rhythm may suffice for deriving a meaningful segmentation at a higher structural level. As an example, Figure 4.6d shows such a mid-level representation, a **tempogram**, which encodes local tempo information. More precisely, a cyclic variant of a tempogram is shown, where tempi differing by a power of two are identified—similar to cyclic chroma features, where pitches differing by octaves are identified. Technical details on how to compute such tempograms can be found in Section 6.2.4. Having a look at Figure 4.6d, one can notice that the different musical parts are played in different tempi (even though the representation does not reveal the exact tempi). Furthermore, there are sections where the tempogram features do not have any dominating entries, which may indicate that there is no clear notion of a tempo in the recording. This kind of information is also important and can be used for segmentation purposes. As this example indicates, a tempogram may yield information that is complementary to the information obtained by chroma-based or MFCC-based feature representations.

Besides the various musical dimensions, there is another aspect one should keep in mind when looking for suitable feature representations: the temporal dimension. In all of the above-mentioned feature representations, an analysis window is shifted over the music signal. As we have already seen for the STFT in Section 2.5.2, the length of the analysis window as well as the hop size parameter have a crucial influence on the quality of the feature representation. For example, long window sizes and large hop sizes may be beneficial for smoothing out irrelevant local variations,

which is often a desired property in homogeneity-based segmentation. On the downside, the temporal resolution decreases and important details may get lost, which can lead to problems when locating the exact segmentation boundaries.

In summary, a suitable choice of feature representations and parameter settings very much depends on the application context. Humans constantly and often unconsciously adapt themselves to the musical and acoustic characteristics of what they listen to. The richness and variety of musical structures make computational structure analysis a challenging problem.

4.2 Self-Similarity Matrices

We have seen that the principles of repetition, homogeneity, and novelty are fundamental for partitioning a given audio recording into musically meaningful structural elements. To study musical structures and their mutual relations, one general idea is to convert the music signal into a suitable feature sequence and then to compare each element of the feature sequence with all other elements of the sequence. This results in a **self-similarity matrix** (SSM), a tool which is of fundamental importance not only for music structure analysis but also for the analysis of many kinds of time series. In this section, we look at these matrices in detail. As we will see, one crucial property of self-similarity matrices is that repetitions typically yield path-like structures, whereas homogeneous regions yield block-like structures. These structural elements are exploited by most algorithms for visualizing, analyzing, and computing musical structures in one way or another. In Section 4.2.1, we introduce the concept of self-similarity matrices and discuss their basic structural properties. For applications, the improvement of these properties at an early state of the processing pipeline is of great importance, which is the topic of Section 4.2.2.

4.2.1 Basic Definitions and Properties

As said before, the concept of self-similarity matrices is fundamental for capturing structural properties of music recordings. Generally, one starts with a feature space \mathcal{F} containing the elements of the feature sequence under consideration as well as with a similarity measure

$$s : \mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R} \quad (4.1)$$

that makes it possible to compare these elements. Typically, the value $s(x, y)$ is high in case the elements $x, y \in \mathcal{F}$ are similar and small otherwise. Given a feature sequence $X = (x_1, x_2, \dots, x_N)$, the idea is to compare all elements of the sequence with each other. This results in an N -square **self-similarity matrix** $\mathbf{S} \in \mathbb{R}^{N \times N}$ defined by

$$\mathbf{S}(n, m) := s(x_n, x_m), \quad (4.2)$$

where $x_n, x_m \in \mathcal{F}$, $n, m \in [1 : N]$. In the following, a tuple $(n, m) \in [1 : N] \times [1 : N]$ is also called a **cell** of **S**, and the value $\mathbf{S}(n, m)$ is referred to as the **score** of the cell (n, m) .

Obviously, the concept of self-similarity matrices is closely related to the concept of cost matrices, which we have already encountered in Section 3.2.1. However, instead of a cost measure c as in (3.12), we now use a similarity measure s . And instead of comparing two sequences X and Y with each other, we now compare a single sequence X with itself. Depending on the application context and notion that is used to compare the data, there are many related concepts known under different names such as recurrence plot or self-distance matrix just to name a few. In this chapter, we only consider self-similarity matrices, but the techniques to be explained can easily be transferred to other types of matrices.

In the following discussion, we assume that the feature space is a Euclidean space $\mathcal{F} = \mathbb{R}^D$ of some dimension $D \in \mathbb{N}$. For simplicity and illustration purposes, we use as similarity measure s the absolute value of the inner product defined by

$$s(x, y) := |\langle x | y \rangle| \quad (4.3)$$

for two vectors $x, y \in \mathcal{F}$ (see (2.37)). With this similarity measure, the score between two orthogonal feature vectors is zero and otherwise it is positive. In the case that the feature vectors are normalized with respect to the Euclidean norm, the similarity values $s(x, y)$ lie in the interval $[0, 1]$. Obviously, there are many more possibilities to define a similarity measure (see Exercise 4.1). The suitability of a similarity measure depends on the properties of the considered features and vice versa.

Given a feature sequence $X = (x_1, x_2, \dots, x_N)$, it seems reasonable to require that an element x_n should be maximally similar to itself. Using normalized features and the similarity measure from (4.3), the similarity measure assumes its maximal value $s(x_n, x_n) = 1$ for all $n \in \mathbb{N}$. Therefore, the resulting SSM has a diagonal with large values. More generally, recurring patterns of the given feature sequence become visible in the SSM in the form of structures with large similarity values. The two most prominent structures induced by such patterns are often referred to as blocks and paths (see Figure 4.7a for an illustration). First, if the feature sequence captures musical properties that stay somewhat constant over the duration of an entire musical part, each of the feature vectors is similar to all other feature vectors within this segment. As a result, an entire **block** of large values appears in the SSM. In other words, homogeneity properties correspond to block-like structures. Second, if the feature sequence contains two repeating subsequences (e.g., two segments corresponding to the same musical part), the corresponding elements of the two subsequences are similar to each other. As a result, a **path** (or **stripe**) of high similarity running parallel to the main diagonal becomes visible in the SSM. In other words, repetitive properties correspond to path-like structures.

Before we further formalize these properties, let us have a look at Figure 4.7, which shows different self-similarity matrices for our Brahms example. Figure 4.7a shows an idealized SSM. For example, assuming that the three repeating A-part segments are homogeneous, the SSM has a quadratic block relating the segment

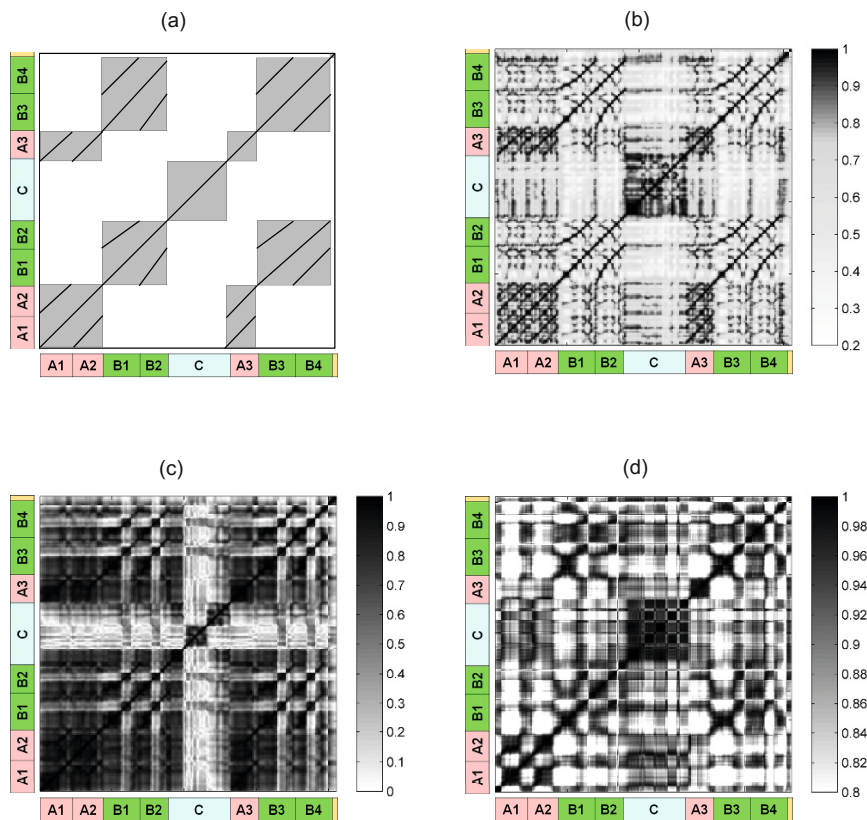


Fig. 4.7 Self-similarity matrices for the Hungarian Dance No. 5 by Johannes Brahms derived from various feature representations shown in Figure 4.6. (a) Idealized SSM. (b) SSM using chroma-based features. (c) SSM using MFCC-based features. (d) SSM using tempo-based features.

corresponding to A_1A_2 to itself and another quadratic block relating the A_3 -part segment to itself. Furthermore, there are two rectangular blocks, one relating the A_1A_2 -part segment to the A_3 -part segment and the other relating the A_3 -part segment to the A_1A_2 -part segment. In case that the three repeating A-part segments are not homogeneous, the SSM reveals path structures that run (more or less) parallel to the main diagonal. For example, there is a path with large similarity values relating A_1 with A_2 and one relating A_1 with A_3 .

How are such structures reflected in the case of “real” SSMs? Besides the idealized SSM, Figure 4.7 shows different self-similarity matrices for our Brahms example obtained from the three conceptually different feature sequences of Figure 4.6. In the visualization, large values of S are indicated by dark gray and small values by light gray. First, one can notice that properties of a self-similarity matrix crucially depend on the respective feature type. The SSM in Figure 4.7b, which is obtained

from chroma-based features, resembles the idealized SSM to a large extent. The block-like structures corresponding to *A*-part segments indicate that these segments are quite homogeneous with respect to harmony. The same holds for the *C*-part segment. Furthermore, the small similarity values outside the *C*-part block (i.e., all cells relating the *C*-part frames to frames of other segments) show that the *C*-part segment is harmonically more or less unrelated to all other parts. For the *B*-part segments, there are path-like structures and no block-like structures. This shows that the *B*-part segments share the same harmonic progression (i.e., are repetitions with regard to harmony), but are not homogeneous with respect to harmony. An interesting observation is that, even though repeating, the *B*-part segments are played in different tempi and therefore have different lengths. For example, the shorter B_2 -section is played faster than the B_1 -section. As a result, the corresponding path does not run exactly parallel to the main diagonal. The gradient of the path indicates the relative tempo difference between the two related segments. Recall that we have discussed a similar issue already in the music synchronization context, where we derived a tempo curve from a warping path (see Section 3.3.2).

Looking at the other two self-similarity matrices the structures are not so clear. The SSM of Figure 4.7c, which results from MFCC-based features, mainly possesses block-like structures. In particular, the *C*-part segment has a low similarity to all other segments, which indicates a difference in timbre or instrumentation. Now, let us have a look at the tempogram-based SSM shown in Figure 4.7d. Again the *C*-part segment stands out, thus emphasizing its contrasting role. Furthermore, the SSM indicates the many tempo changes occurring in this music recording. In summary, the musical structure of the Brahms example can be best explained by the repetitive structure of the chroma-based SSM. Since this is the case with many musical works, in particular for melodic and harmonic Western music, we will mainly focus on this type of SSM in the subsequent sections.

We now formalize the concept of paths and blocks (see Figure 4.8). Let $X = (x_1, x_2, \dots, x_N)$ be a feature sequence and \mathbf{S} the resulting self-similarity matrix. We formally define a segment to be a set $\alpha = [s : t] \subseteq [1 : N]$ specified by its starting point s and its end point t (given in terms of feature indices). Let

$$|\alpha| := t - s + 1 \quad (4.4)$$

denote the length of α . Next, a **path** over α of length L is a sequence

$$P = ((n_1, m_1), \dots, (n_L, m_L)) \quad (4.5)$$

of cells $(n_\ell, m_\ell) \in [1 : N]^2$, $\ell \in [1 : L]$, satisfying $m_1 = s$ and $m_L = t$ (boundary condition) and $(n_{\ell+1}, m_{\ell+1}) - (n_\ell, m_\ell) \in \Sigma$ (step size condition), where Σ denotes a set of admissible step sizes. Note that this definition is very similar to the one of a warping path (see Section 3.2.1.1). In the case of $\Sigma = \{(1, 1)\}$, one obtains paths that are strictly diagonal. In the following, we typically use the set

$$\Sigma = \{(2, 1), (1, 2), (1, 1)\}, \quad (4.6)$$

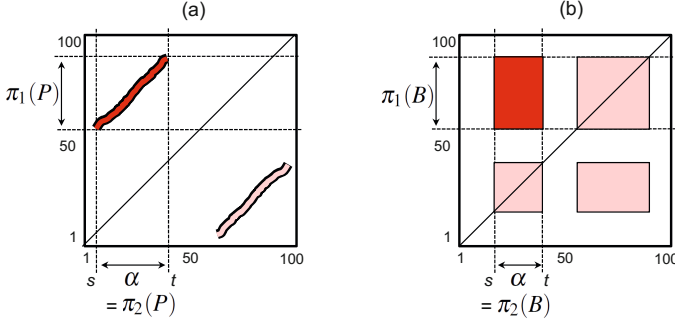


Fig. 4.8 Schematic view of self-similarity matrix with (a) a path and (b) a block.

which is the step size condition introduced in (3.30). For a path P , one can associate two segments defined by the projections

$$\pi_1(P) := [n_1 : n_L] \quad \text{and} \quad \pi_2(P) := [m_1 : m_L], \quad (4.7)$$

respectively (see Figure 4.8a). The boundary condition enforces $\pi_2(P) = \alpha$. The other segment $\pi_1(P)$ is referred to as the **induced segment**. The **score** $\sigma(P)$ of P is defined as

$$\sigma(P) := \sum_{\ell=1}^L \mathbf{S}(n_\ell, m_\ell). \quad (4.8)$$

Note that each path over the segment α encodes a relation between α and an induced segment, where the score $\sigma(P)$ yields a quality measure for this relation.

For blocks, we also introduce corresponding notions. A **block** over a segment $\alpha = [s : t]$ is a subset

$$B = \alpha' \times \alpha \subseteq [1 : N] \times [1 : N] \quad (4.9)$$

for some segment $\alpha' = [s' : t']$. Similar as for a path, we define the two projections $\pi_1(B) = \alpha'$ and $\pi_2(B) = \alpha$ for the block B and call α' the **induced segment** (see Figure 4.8b). Furthermore, we define the score of block B by

$$\sigma(B) = \sum_{(n,m) \in B} \mathbf{S}(n, m). \quad (4.10)$$

Based on paths and blocks, we can now consider different kinds of similarity relations between segments. We say that a segment α_1 is **path-similar** to a segment α_2 , if there is a path P of high score with $\pi_1(P) = \alpha_1$ and $\pi_2(P) = \alpha_2$. Similarly, α_1 is **block-similar** to α_2 , if there is a block B of high score with $\pi_1(B) = \alpha_1$ and $\pi_2(B) = \alpha_2$. Obviously, in case that the similarity measure s is symmetric, both the self-similarity matrix \mathbf{S} and the above-defined similarity relations between segments are symmetric as well. Another important property of a similarity relation is **transitivity**, i.e., if a segment α_1 is similar to a segment α_2 and segment α_2 is similar

to a segment α_3 , then α_1 should also be similar to α_3 (at least to a certain degree). Also this property holds for path- and block-similarity in case that the similarity measure s has this property. As a consequence, path and block structures often appear in groups that fulfill certain symmetry and transitivity properties—at least in the ideal case. For example, if there is a block $B = \alpha' \times \alpha$ of high score, then the symmetry property implies that there is also a block $\alpha \times \alpha'$ of high score. Furthermore, if every frame belonging to α is similar to every other frame of α' , then also the frames within the segments α and α' are similar to each other. This leads to additional blocks $\alpha \times \alpha$ and $\alpha' \times \alpha'$ (see Figure 4.8b). Figure 4.7 shows that such groups of similarity relations also appear in “real” SSMs.

Most computational approaches to music structure analysis exploit path- and block-like structures of SSMs in one way or another, and the overall algorithmic pipelines typically contain the following general steps:

1. The music signal is transformed into a suitable feature sequence.
2. A self-similarity matrix is computed from the feature sequence based on a similarity measure.
3. Blocks and paths of high overall score are derived from the SSM. Each block or path defines a pair of similar segments.
4. Entire groups of mutually similar segments are formed from the pairwise relations by applying a clustering step.

The last step can be considered as forming a kind of transitive closure of the pairwise segment relations induced by block and path structures. For example, in the case of Brahms’ Hungarian Dance No. 5 (see Figure 4.7), the objective of the last step would be to find one group that contains all *A*-part segments and another group that contains all *B*-part segments.

In practice, this general processing pipeline leaves a lot of freedom and needs to be adjusted to account for particular properties of the underlying type of music and the requirements of the intended application. Furthermore, as mentioned before, major challenges arise from the fact that musical parts are rarely repeated in precisely the same way. Instead, audio segments that are considered as repetitions may differ significantly in aspects such as dynamics, orchestration, articulation, tempo, harmony, melody, or any combination of these. As a result, structure analysis becomes a hard and often ill-posed task. In particular, musical and acoustic variations may cause significant deteriorations in the path and block structures and their induced relations. This makes both steps, i.e., the block and path extraction step as well as the grouping step, error-prone and fragile. In the following, we discuss various strategies to cope with such challenges, e.g., by enhancing structural properties of SSMs (Section 4.2.2) or by jointly performing the two error-prone steps of path extraction and grouping within a joint optimization scheme (Section 4.3).

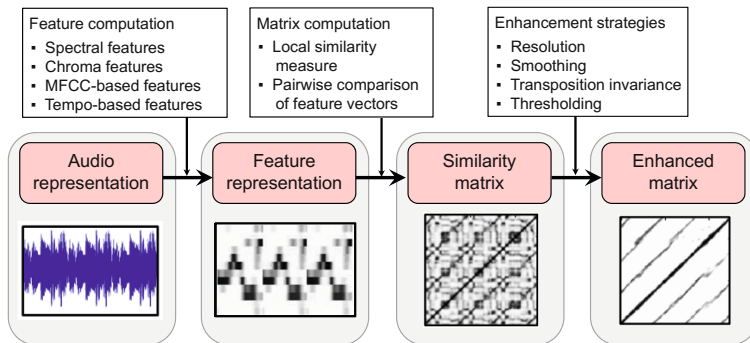


Fig. 4.9 Overview of the similarity matrix computation.

4.2.2 Enhancement Strategies

In this section, we describe various strategies for enhancing structural properties of self-similarity matrices (see Figure 4.9 for an overview). In particular, we focus on augmenting path-like structures, which play a central role in repetition-based structure analysis. Even though all the enhancement strategies are described for self-similarity matrices, similar strategies can be applied for more general similarity or cost matrices.

4.2.2.1 Feature Representation

In the first step, the given waveform-based audio recording is transformed into a suitable feature representation, which captures specific acoustic and musical properties. As we have already discussed in Section 4.2.1 and as illustrated by Figure 4.7, the structural properties of an SSM decisively depend on the feature type used. For example, MFCC-based and related spectral-based features may be suitable to capture aspects such as instrumentation and timbre. Other features based on onset information or tempograms are used to capture beat, tempo, and rhythmic information. In the following, we only consider the case of chroma-based audio features, which relate to harmonic and melodic properties as discussed in Section 3.1.2.

By considering a family of modified chroma representations similar to the ones used in Figure 3.9, we now demonstrate the influence of different parameter settings on the properties of the resulting SSM. Starting with a chroma representation of a given feature rate, this family comes along with two parameters: a length parameter $\ell \in \mathbb{N}$ (given in frames), which is used to smooth or average the feature values over ℓ consecutive frames, as well as a downsampling parameter d , which reduces the feature rate by a factor of d . For a more detailed description of such a procedure, we refer to Section 7.2.1 and Figure 7.10.

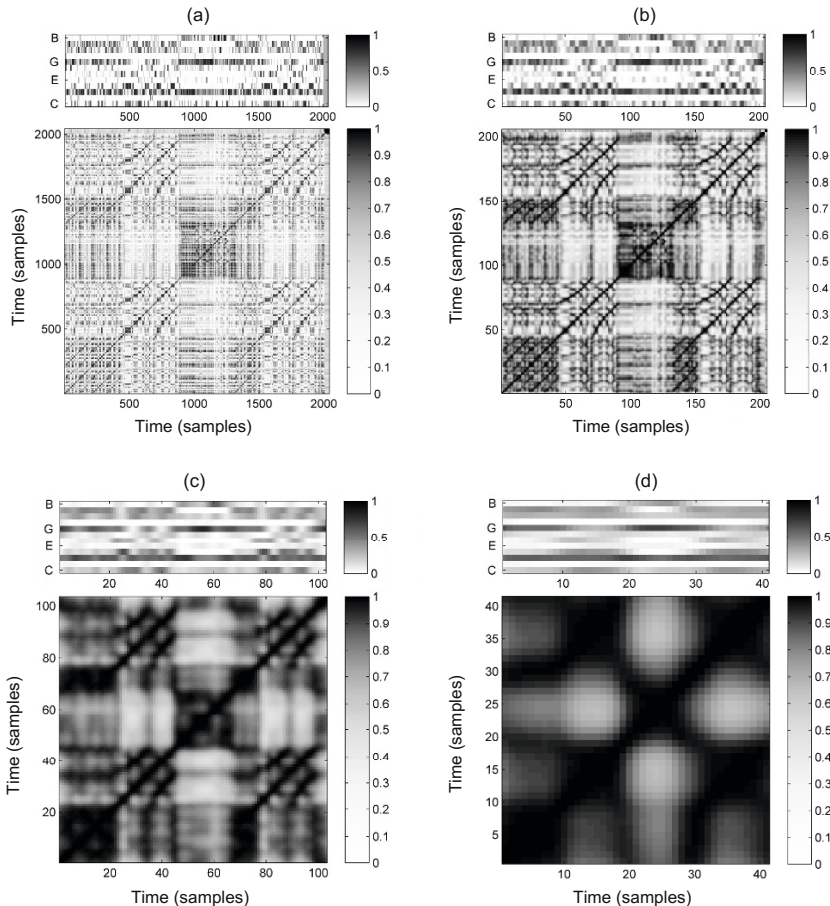


Fig. 4.10 Various chroma representations and resulting SSMs for the Hungarian Dance No. 5 by Johannes Brahms. (a) Usage of original normalized chroma features (10 Hz). (b) Applying $\ell = 40$ and $d = 10$ (1 Hz). (c) Applying $\ell = 160$ and $d = 20$ (0.5 Hz). (d) Applying $\ell = 480$ and $d = 50$ (0.2 Hz).

As an example, we start with normalized chroma features with a feature rate of 10 Hz. Figure 4.10a shows the resulting SSM, which yields a very detailed description of repetitive structures. Even though the path structures that correspond to the repeating *A*-part and *B*-part segments are visible, the SSM looks quite noisy and many of the shown details are irrelevant when only the overall musical structure is of interest.

Using a smoothing length of $\ell = 40$ (corresponding to four seconds of audio) and a downsampling by $d = 10$ (resulting in a feature rate of 1 Hz), one obtains the SSM shown in Figure 4.10b. Many of the details have been smoothed out, and some of the structurally relevant path and block structures have become more prominent.

In particular, this holds for the paths that relate to the *B*-part segments. Moreover, reducing the feature rate improves the computational efficiency for subsequent processing steps.

Further increasing the smoothing length and reducing the feature rate results in an emphasis of the rough harmonic content. In particular, neighboring elements in the feature sequence come closer together, which leads to an enhancement of block-like structures. For example, Figure 4.10c shows the SSM when using $\ell = 160$ (16 seconds) and $d = 20$ (feature rate of 0.5 Hz) and Figure 4.10d the SSM using $\ell = 480$ (48 seconds) and $d = 50$ (feature rate of 0.2 Hz). Using large smoothing windows, relevant path structures may be smeared out and lost for the subsequent steps. For other applications such as homogeneity-based structure analysis, however, averaging over large windows may be beneficial.

In summary, this example shows the importance not only of the feature type but also of the size of the analysis window and the feature rate. Knowing the temporal level of the music processing task is of great help for choosing suitable parameters. For example, for tasks such as extracting the musical structure from a given audio recording, smoothing and downsampling already on the feature level can lead to substantial improvements, not to speak of computational benefits in subsequent analysis steps. In particular, running time and memory requirements are important issues when employing concepts such as SSMs, which are quadratic in the length of the input feature sequence. As already mentioned in Section 4.1.3, another important strategy for adjusting and reducing the feature rate is based on **adaptive windowing**, where the analysis windows are determined by previously extracted onset and beat positions. This strategy will be discussed in more detail in Section 6.3.3.

4.2.2.2 Path Smoothing

We have seen that important structural elements of similarity matrices are paths of high similarity that run parallel to the main diagonal. Even though it is often easy for humans to recognize these structures, the automated extraction of paths constitutes a difficult problem due to significant distortions that are caused by variations in parameters such as dynamics, timbre, execution of note groups (e.g., grace notes, trills, arpeggios), modulation, articulation, or tempo progression. As an example, let us have a look at Figure 4.11a, which shows the SSM of a recording of the Waltz No. 2 from Dimitri Shostakovich's Suite for Variety Orchestra No. 1. This piece has the (rough) musical structure $A_1A_2BC_1C_2A_3A_4D$, where the theme, represented by the *A*-part, appears four times. However, there are significant variations in the four *A*-parts concerning instrumentation, articulation, as well as dynamics. For example, in A_1 the theme is played by a clarinet, in A_2 by strings, in A_3 by a trombone, and in A_4 by the full orchestra. As is illustrated by Figure 4.11a, these variations result in a rather poor and fragmented path structure. This makes it hard to identify the musically similar segments $\alpha_1 = [4 : 40]$, $\alpha_2 = [43 : 78]$, $\alpha_3 = [145 : 179]$, and $\alpha_4 = [182 : 217]$ corresponding to A_1 , A_2 , A_3 , and A_4 , respectively. In particular, as

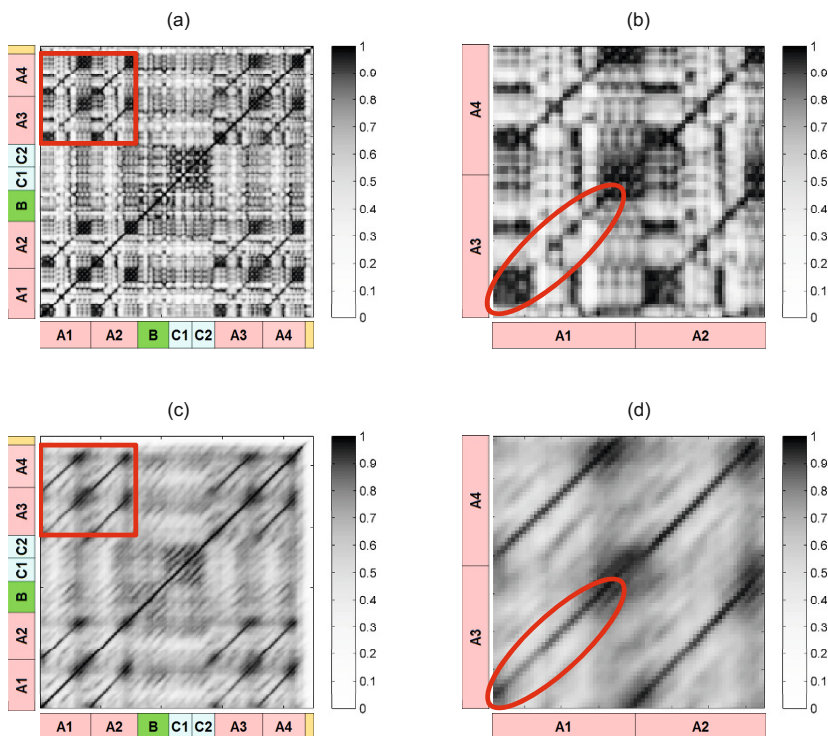


Fig. 4.11 Variants of SSMs for a recording of the Waltz No. 2 from Dimitri Shostakovich's Suite for Variety Orchestra No. 1. **(a)** Original SSM using chroma features (resolution of 1 Hz). **(b)** Enlargement of the submatrix indicated by the rectangular frame in (a). The path corresponding to segments α_1 (part A1) and α_3 (part A3) is highlighted by the oval. **(c)** SSM after applying diagonal smoothing. **(d)** Enlargement of the submatrix indicated by the rectangular frame in (c).

can be seen in the enlargement shown in Figure 4.11b, the path corresponding to the segments α_1 and α_3 is quite problematic.

To some extent, as we have seen above, structural properties of the SSM may be augmented by using longer analysis windows in the feature computation step. This, however, may also smooth out important details. As an alternative, we now show how to enhance the path structure of an SSM by applying image processing techniques. Recall that the relevant paths run along the direction of the main diagonal in the case that repeating parts are played in the same tempo. Therefore, in order to augment such paths, the general idea is to apply an averaging filter (or low-pass filter) in the direction of the main diagonal, which results in an emphasis of diagonal information and a softening of other, nondiagonal structures.

We now give a mathematical description of this procedure. Let \mathbf{S} be an SSM of size $N \times N$ and let $L \in \mathbb{N}$ be a length parameter. Then we define the smoothed self-similarity matrix \mathbf{S}_L by setting

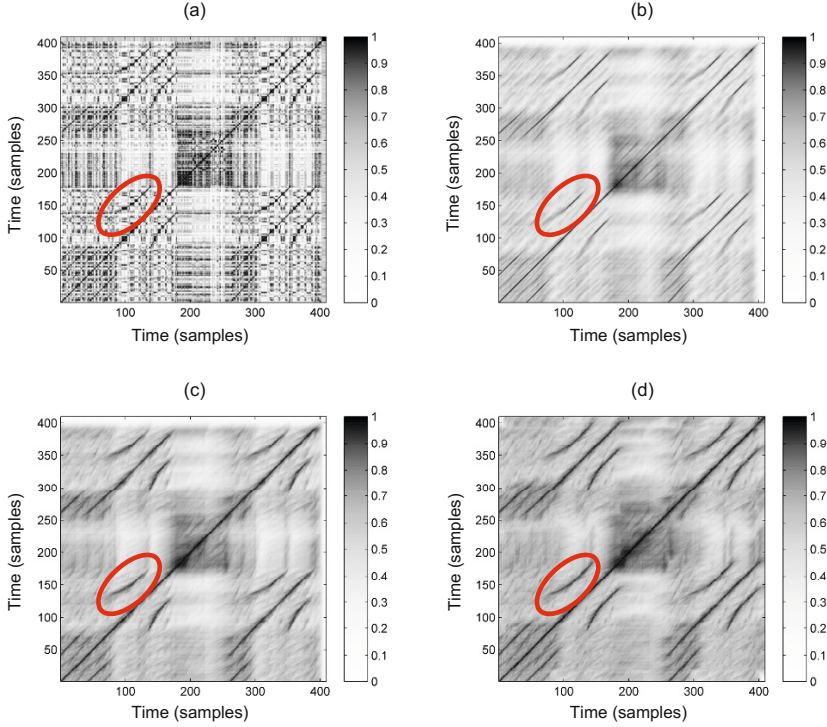


Fig. 4.12 Variants of SSMs for the Hungarian Dance No. 5 by Johannes Brahms. The path corresponding to the B_1 -part and B_2 -part segments is highlighted. **(a)** Original SSM using chroma features (resolution of 2 Hz). **(b)** SSM after applying diagonal smoothing. **(c)** SSM after applying tempo-invariant smoothing. **(d)** SSM after applying forward-backward smoothing.

$$\mathbf{S}_L(n, m) := \frac{1}{L} \sum_{\ell=0}^{L-1} \mathbf{S}(n + \ell, m + \ell) \quad (4.11)$$

for $n, m \in [1 : N - L + 1]$. In other words, the value $\mathbf{S}_L(n, m)$ is obtained by averaging the similarity values of two subsequences of length L , one starting at index n and the other at index m . By suitably extending \mathbf{S} (e.g., by **zero-padding** where zero columns and rows are added), we may assume in the following that $\mathbf{S}_L(n, m)$ is defined for $n, m \in [1 : N]$.

The averaging procedure results in a smoothing effect along the main diagonal, which is also illustrated by our Shostakovich example of Figure 4.11. Using the length parameter $L = 10$, the resulting self-similarity matrix \mathbf{S}_{10} (Figure 4.11c) reveals the desired path structure much better than the original matrix \mathbf{S} (Figure 4.11a). For example, the enhanced path highlighted in Figure 4.11d reveals the relation between the segments α_1 and α_3 much better than before (see Figure 4.11b).

A simple filtering along the main diagonal only works well if there are no relative tempo differences between the segments to be compared. However, this assumption is violated when a part is repeated with a faster or slower tempo. We have seen such a case in our Brahms example from Figure 4.7, where the shorter B_2 -section is played much faster than the B_1 -section. It is only the beginning of the B_2 -section that is played much faster than the beginning of the B_1 -section, whereas the two sections have roughly the same tempo towards the end of the part. This results in a path that does not run exactly parallel to the main diagonal (in particular at the beginning), so that applying an averaging filter in the direction of the main diagonal destroys some of the path structure (see Figure 4.12b). To deal with such relative tempo differences, one idea is to apply a multiple filtering approach, where the SSM is smoothed along various directions that lie in a neighborhood of the direction defined by the main diagonal. Each such direction corresponds to a tempo difference and results in a separate filtered matrix. The final self-similarity matrix is obtained by taking the cell-wise maximum over all these matrices. In this way, the path structure is also enhanced in the presence of local tempo variations as illustrated in Figure 4.12c.

To better understand the details of this procedure, first assume that we have two repeating segments α_1 and α_2 played at the same tempo. Then the direction of the resulting path is given by the gradient $(1, 1)$. Next, assume that the second segment α_2 is played at half the tempo compared with α_1 . Then the direction of the resulting path is given by the gradient $(1, 2)$. In general, if the tempo difference between the two segments is given by a real number $\theta > 0$ (the second segment played θ times slower than the first one), the resulting gradient is $(1, \theta)$. We define the self-similarity matrix smoothed in the direction of $(1, \theta)$ by

$$\mathbf{S}_{L,\theta}(n, m) := \frac{1}{L} \sum_{\ell=0}^{L-1} \mathbf{S}(n + \ell, m + \lceil \ell \cdot \theta \rceil), \quad (4.12)$$

where $\lceil \ell \cdot \theta \rceil$ denotes the integer closest to the real number $\ell \cdot \theta$. Again, by suitably zero-padding the matrix \mathbf{S} , we may assume that $\mathbf{S}_{L,\theta}$ is defined for $n, m \in [1 : N]$. Now, in practice, one does not know the local tempo difference that may occur in a given music recording. Also, the relative tempo difference between two repeating sections may change over time (as is the case with our Brahms example). Therefore, the idea is to consider a (finite) set Θ consisting of tempo parameters $\theta \in \Theta$ for different relative tempo differences. Then, we compute for each such θ a matrix $\mathbf{S}_{L,\theta}$ and obtain a final matrix $\mathbf{S}_{L,\Theta}$ by a cell-wise maximization over all $\theta \in \Theta$:

$$\mathbf{S}_{L,\Theta}(n, m) := \max_{\theta \in \Theta} \mathbf{S}_{L,\theta}(n, m). \quad (4.13)$$

In practice, one can use prior information on the expected relative tempo differences to determine the set Θ . For example, it rarely happens that the relative tempo difference between repeating segments is larger than 50 percent, so that Θ can be chosen to cover tempo variations of roughly -50 to $+50$ percent. Furthermore, in practice, the tempo range can be covered well by considering only a relatively small number of tempo parameters. For example, a typical choice could be

$\Theta = \{0.66, 0.81, 1.00, 1.22, 1.50\}$ (see Exercise 4.4). Note that choosing $\Theta = \{1\}$ reduces to the case $\mathbf{S}_{L,\Theta} = \mathbf{S}_L$.

This smoothing procedure works in the forward direction, which results in a fading out of the paths, particularly when using a large length parameter. To avoid this fading out, one idea is to additionally apply the averaging filter in a backward direction. The final self-similarity matrix is then obtained by taking the cell-wise maximum over the forward-smoothed and backward-smoothed matrices (see Exercise 4.2). The effect is illustrated in Figure 4.12d by means of the Brahms example.

4.2.2.3 Transposition Invariance

It is often the case that certain musical parts are repeated in a transposed form, where the melody is moved up or down in pitch by a constant interval. As an example, let us consider the song “In the year 2525” by Zager and Evans, which has the musical structure $IV_1V_2V_3V_4V_5V_6V_7BV_8O$. The song starts with a slow intro, which is represented by the I -part. The verse of the song, which is represented by the V -part, is repeated eight times. While the first four verse sections are in the same musical key, V_5 and V_6 are transposed by one semitone upwards, and V_7 and V_8 are transposed by two semitones upwards. Figure 4.13b shows a path-enhanced version of the resulting self-similarity matrix based on some chroma feature representation. This matrix shows path structures that relate the first four V -sections with each other as well as V_5 with V_6 and V_7 with V_8 . Because of the transpositions, however, the relation between the first four sections and the last four sections is not reflected in the SSM.

In the following, we show how repetitive structures can be made visible in the SSM even in the presence of key transpositions. We have already seen in Section 3.1.2 that such transpositions can be simulated by cyclically shifting chroma features. Mathematically, we modeled such shifts by the cyclic shift operator $\rho : \mathbb{R}^{12} \rightarrow \mathbb{R}^{12}$ defined in (3.11). Now, let $X = (x_1, x_2, \dots, x_N)$ be the chroma feature sequence. We then define the **i -transposed self-similarity matrix** $\rho^i(\mathbf{S})$ by

$$\rho^i(\mathbf{S})(n, m) := s(\rho^i(x_n), x_m) \quad (4.14)$$

for $n, m \in [1 : N]$ and $i \in \mathbb{Z}$. Obviously, one has $\rho^{12}(\mathbf{S}) = \mathbf{S}$. Intuitively, $\rho^i(\mathbf{S})$ describes the similarity relations between the original music recording (represented by $X = (x_1, x_2, \dots, x_N)$) and the music recording transposed by i semitones upwards (represented by $\rho^i(X) = (\rho^i(x_1), \rho^i(x_2), \dots, \rho^i(x_N))$). Since one does not know in general the kind of transpositions occurring in the music recording, we apply a similar strategy as before when dealing with relative tempo deviations. Taking a cell-wise maximum over the twelve different cyclic shifts, we obtain a single **transposition-invariant self-similarity matrix** \mathbf{S}^{TI} defined by

$$\mathbf{S}^{\text{TI}}(n, m) := \max_{i \in [0:11]} \rho^i(\mathbf{S})(n, m). \quad (4.15)$$

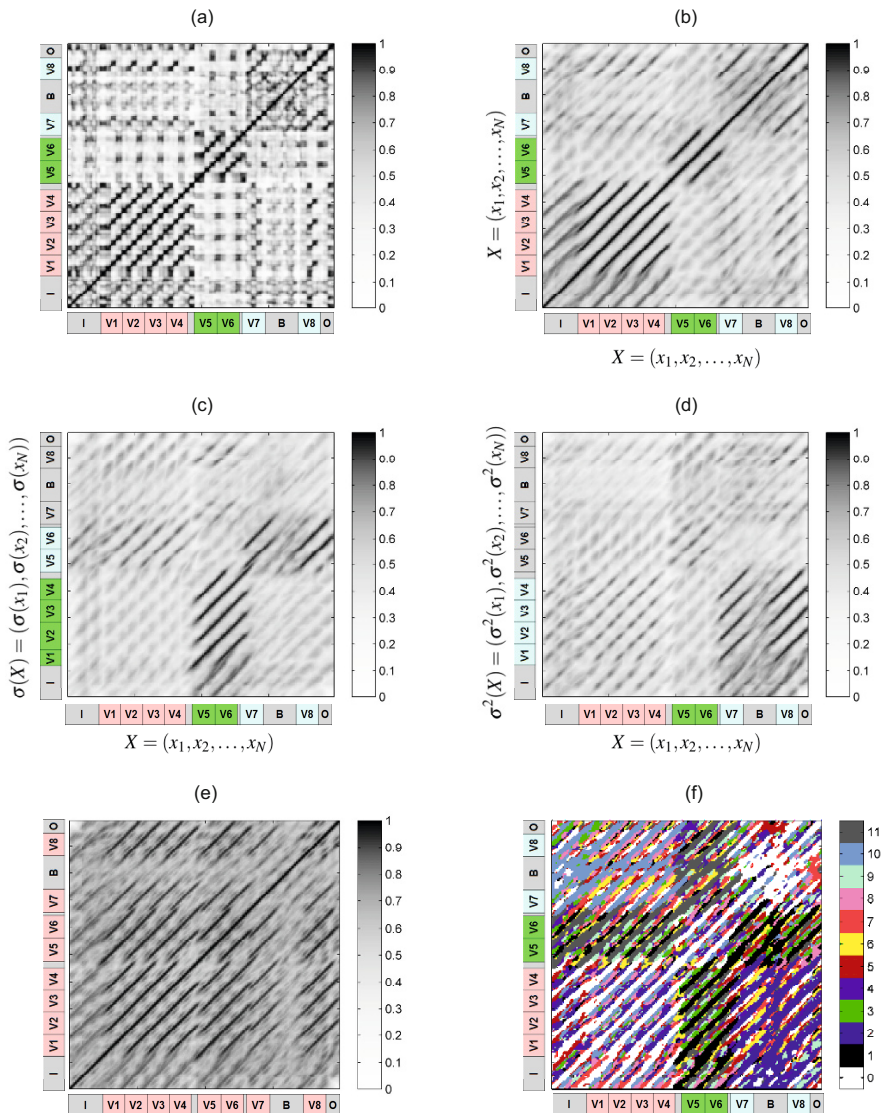


Fig. 4.13 Variants of SSMs for the song “In the year 2525” by Zager and Evans. (a) Original SSM using chroma features (resolution of 1 Hz). (b) Path-enhanced SSM. (c) 1-transposed SSM. (d) 2-transposed SSM. (e) Transposition-invariant SSM. (f) Transposition index matrix.

Furthermore, we store the maximizing shift indices in an additional N -square matrix \mathbf{I} , which we refer to as the **transposition index matrix**:

$$\mathbf{I}(n, m) := \operatorname{argmax}_{i \in [0:11]} \rho^i(\mathbf{S})(n, m). \quad (4.16)$$

We illustrate the definitions by continuing the example shown in Figure 4.13 (see Exercise 4.3). Recall from above that shifting the sections V_1 to V_4 by one semitone upwards makes them similar to the original sections V_5 and V_6 . This fact is revealed by the 1-transposed self-similarity matrix shown in Figure 4.13c. Similarly, shifting the sections V_1 to V_4 by two semitones upwards makes them similar to the original sections V_7 and V_8 (see Figure 4.13d). Putting together the information of all i -transposed self-similarity matrices by the maximization in (4.15), one obtains the transposition-invariant self-similarity matrix \mathbf{S}^{TI} shown in Figure 4.13e, where all pairwise similarity relations between the eight V -part segments become visible.

The resulting transposition index matrix is shown in Figure 4.13f in a color-coded form. We first discuss the case that the matrix \mathbf{I} assumes the value $i = 0$ (white color in Figure 4.13f). The value $i = 0$ for a cell (n, m) indicates that $s(\rho^i(x_n), x_m)$ assumes a maximal value for $i = 0$. In other words, the chroma vector x_m is closer to x_n than to any other shifted version of x_n . Note, however, that this does not necessarily mean that x_m is close to x_n in absolute terms. As may be expected, the maximizing index is $i = 0$ at all positions where the conventional self-similarity matrix shown in Figure 4.13b reveals paths of low cost. Next, we consider the case that the matrix \mathbf{I} assumes the value $i = 1$ (black color in Figure 4.13f). The value $i = 1$ for a cell (n, m) indicates that x_n becomes most similar to x_m when shifted one semitone upwards. Thus the strong path relations shown in Figure 4.13c correspond to cells assuming the value $i = 1$, and so on.

At this point, we want to note that introducing transposition invariance by cell-wise maximization over several matrices may increase the noise level in the resulting similarity matrix. Therefore, the transposition-invariant matrix should be computed on the basis of smoothed matrices, since the smoothing typically goes along with a suppression of unwanted noise. The definitions in (4.14) and (4.15) can be easily combined with the averaging approaches described by (4.11) and (4.12) to yield matrices $\rho_{L,\Theta}^i(\mathbf{S})$ and $\mathbf{S}_{L,\Theta}^{\text{TI}}$. Such matrices are shown in Figure 4.13.

4.2.2.4 Thresholding

In many music analysis applications, self-similarity matrices are further processed by suppressing all values that fall below a given threshold. On the one hand, such a step often leads to a substantial reduction of unwanted noise-like components while leaving only the most significant structures. On the other hand, weaker but still relevant information may be lost. The thresholding strategy used may have a significant impact on the final result and has to be carefully chosen in the context of the considered application. Figure 4.14 shows some examples obtained by different thresholding settings as explained below.

The simplest strategy is to apply **global thresholding**, where all values $\mathbf{S}(n, m)$ of a similarity matrix \mathbf{S} below a given threshold parameter $\tau > 0$ are set to zero:

$$\mathbf{S}_\tau(n, m) := \begin{cases} \mathbf{S}(n, m) & \text{if } \mathbf{S}(n, m) \geq \tau, \\ 0, & \text{otherwise.} \end{cases} \quad (4.17)$$

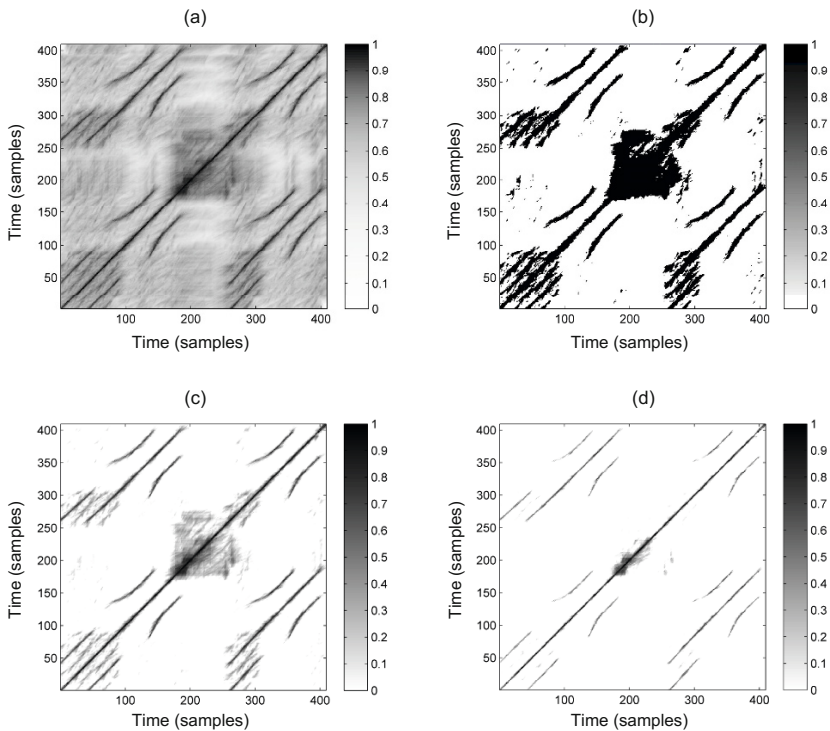


Fig. 4.14 Thresholding strategies applied to an SSM for the Hungarian Dance No. 5 by Johannes Brahms. (a) SSM from Figure 4.12d. (b) SSM after thresholding and binarization ($\tau = 0.75$). (c) SSM after thresholding and scaling ($\rho = 0.2$). (d) SSM after thresholding and scaling ($\rho = 0.05$).

Also, binarization of the similarity matrix can be applied by setting all values above or equal to the threshold to one and all others to zero. Instead of binarization, one may perform a scaling where the range $[\tau, \mu]$ is linearly scaled to $[0, 1]$ in the case that $\mu := \max_{n,m} \{S(n, m)\} > \tau$, otherwise all entries are set to zero. Sometimes it may be beneficial to introduce an additional penalty parameter $\delta \leq 0$, setting all original values below the threshold to the value δ (see Section 4.3 for an application of this variant).

The global threshold τ can also be chosen in a **relative** fashion by keeping $\rho \cdot 100\%$ of the cells with the highest values using a relative threshold parameter $\rho \in [0, 1]$. Finally, thresholding can also be performed using a more **local** strategy by thresholding in a column- and rowwise fashion. To this end, for each cell (n, m) , the value $S(n, m)$ is kept if it is among the $\rho \cdot 100\%$ of the largest cells in row n and at the same time among the $\rho \cdot 100\%$ of the largest cells in column m , all other values being set to zero (see Exercise 4.5). As said before, the suitability of a thresholding setting depends on the respective music material and the application in mind. Often, suitable thresholds are learned and optimized using supervised learning procedures.

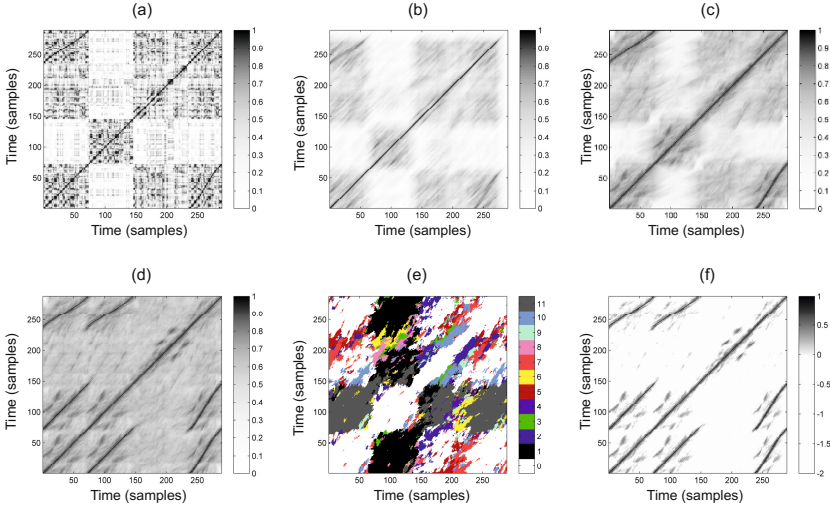


Fig. 4.15 Variants of similarity matrices for the same audio recording. (a) Original SSM using chroma features of 2 Hz resolution. (b) SSM after applying diagonal smoothing. (c) SSM after applying tempo-invariant and forward-backward smoothing. (d) Transposition-invariant SSM. (e) Transposition index matrix. (f) SSM after thresholding with penalty and scaling ($\rho = 0.2$, $\delta = -2$).

To conclude this section, Figure 4.15 summarizes the various enhancement and processing steps applied to a music recording having the musical structure $A_1A_2BA_3$. In this example, A_2 is a modulation of A_1 transposed by one semitone upwards, whereas A_3 is a repetition of A_1 , however played much faster. Figure 4.15 shows a typical processing pipeline for computing an SSM as used in structure analysis applications. First, the music recording is converted into a sequence of normalized and smoothed chroma features as in Figure 3.9. Then, based on the similarity measure (4.3), an enhanced transposition-invariant self-similarity matrix $\mathbf{S}_{L,\Theta}^{\text{TI}}$ is computed (see Figure 4.15d). In the next step, global thresholding is applied using a threshold parameter τ and a penalty parameter δ . Furthermore, the range $[\tau, 1]$ is linearly scaled to $[0, 1]$. As a result, the relevant path structure tends to lie in the positive part of the resulting SSM, whereas all other cells are given a negative score. Finally, setting $\mathbf{S}(n, n) = 1$ for $n \in [1 : N]$, one can introduce a normalization property, which may have been lost in the smoothing process due to boundary effects. The SSM shown in Figure 4.15f is obtained in this way using a feature rate of 2 Hz. Settings for the enhancement are $L = 20$ for the length parameter and $\Theta = \{0.50, 0.63, 0.79, 1.26, 1.59, 2.00\}$ for the set of relative tempo differences (see Exercise 4.4). In this example, the threshold is chosen in a relative fashion by using the relative threshold $\rho = 0.2$ and the penalty parameter is set to $\delta = -2$.

Fundamentals of Music Processing

Audio, Analysis, Algorithms, Applications

Müller, M.

2015, XXIX, 487 p. 249 illus., 30 illus. in color.,

Hardcover

ISBN: 978-3-319-21944-8