Extracting Expressive Tempo Curves from Music Recordings

Verena Konz, Meinard Müller, Andi Scharfstein
Saarland University and MPI Informatik, Saarbrücken, Germany, {vkonz,meinard,ascharfs}@mpi-inf.mpg.de

Introduction

Musicians give a piece of music their personal touch by continuously varying tempo, dynamics, and articulation. Instead of playing mechanically they speed up at some places and slow down at others in order to shape the piece of music. Similarly, they continuously change the sound intensity and stress certain notes. The automated analysis of different interpretations, also referred to as *performance analysis*, has become an active research field [2, 4, 5, 7, 9, 8]. Here, one goal is to find commonalities between different interpretations, which allow for the derivation of general performance rules. A kind of orthogonal goal is to capture what is characteristic for the style of a particular musician. Before one can actually analyze a specific performance, one needs the information about when and how the notes of the underlying piece of music are played, see Fig. 1. Such information typically comprises parameters that make explicit the exact timing and intensity of the various note events occurring in the performance. Most of the current algorithms for automated performance analysis rely on accurate annotations of the music material by means of such parameters. Here, the annotation process, in particular in the case of music recordings, is often done manually, which is prohibitive in view of large audio collections. In this paper, we present a fully automatic approach for extracting temporal information from music recordings. This information is given in the form of tempo curves that reveal the relative tempo difference between an actual performance and some reference representation (in the form of a MIDI file) of the underlying musical piece. We conclude this paper with a short summary on previous work in the field of performance analysis and give an outlook on future work.

Extracting Tempo Parameters

Before we present our fully automatic approach for extracting tempo parameters, we first discuss some general ways on how one may obtain such information. General orchestral music is notoriously difficult to process automatically in the context of performance analysis. In the following, we therefore restrict ourselves to piano pieces from Western classical music, which is somewhat better tractable. This allows us to exploit certain characteristics of the piano sound. In particular, when pressing a piano key, there typically is a sudden energy increase in the sound pressure. Furthermore, the starting time when a note is played is comparatively well defined, since the only time when artistic shaping occurs is determined by when and how a specific piano key is played.[1] The specific starting time of a note event is referred to as the *note onset*. Most approaches for automated performance analysis rely on the knowledge of precise note onset information in the form of suitable annotations. Such information can be obtained in various ways.

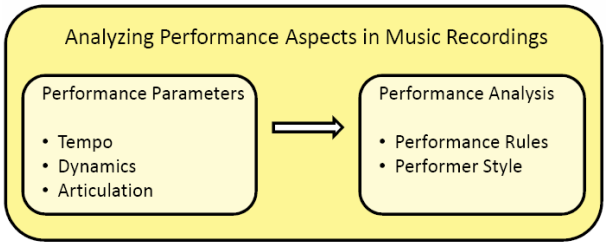

Figure 1: Analyzing performance aspects in music recordings.

1. **Manual annotations.** Many researchers manually annotate the audio material by locating salient data points in the audio stream. Using novel music analysis interfaces such as the Sonic Visualiser [6], experienced annotators can locate note onsets very accurately even in complex audio material. However, being very labor-intensive, such a manual process is prohibitive in view of large audio collections. In practice, semi-automatic approaches are often used, where one first roughly computes beat timings using beat tracking software, which are then adjusted manually to yield precise note onsets.

2. **Direct annotations.** Another way to generate highly accurate annotations is to use a computer-monitored *player piano*. Equipped with optical sensors and electromechanical devices, such pianos allow for recording the key movements along with the acoustic audio data, from which one directly obtains the desired note onset information.

3. **Automatic annotations.** Using automated methods such as *beat tracking* or *onset detection* algorithms, one can try to estimate the precise timings of note events within an audio recording. Even though great research efforts have been directed towards such tasks, the results are still unsatisfactory, in particular for classical music with weak onsets and strongly varying beat patterns. Therefore, the usage of automated methods for extracting musical parameters is still problematic in view of subsequent performance analysis applications.

[1] Ignoring some (admittedly important) aspects like pedalling and key release times

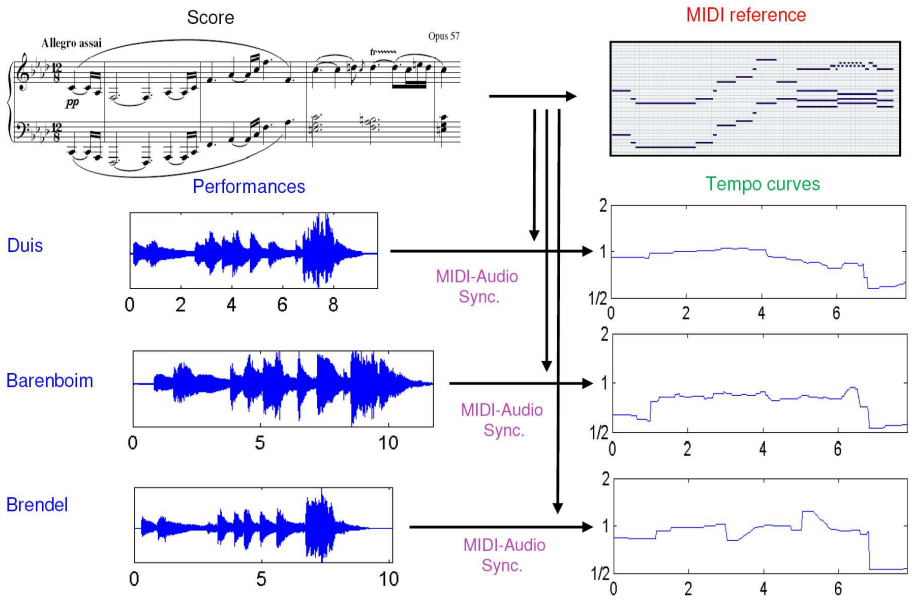

Figure 2: Extracting tempo curves from music recordings.

In the following, we describe how one can extract timing information from audio recordings using *music synchronization* algorithms. In particular, we exploit the fact that for most classical pieces there exists a kind of "neutral" representation in the form of a musical score. A score contains high-level information on the notes such as musical onsets time, pitch, and duration. In the following, we assume that the score is represented by a MIDI file that explicitly provides the musical onset and pitch information of all occurring note events. On the other hand, we have the audio recording of a specific performance to be annotated. Now, the idea is to use conventional *MIDI-audio synchronization* techniques to temporally align the MIDI events with their corresponding physical occurrences in the audio recording. The synchronization result can be regarded as an automated annotation of the audio recording with the available note events given by the MIDI file. Most synchronization algorithms rely on some variant of dynamic time warping (DTW) and can be summarized as follows. First, the MIDI file and the audio recording to be aligned are converted into feature sequences, say $V := (v_1, v_2, \ldots, v_N)$ and $W := (w_1, w_2, \ldots, w_M)$, respectively. Then, an $N \times M$ cost matrix C is built up by evaluating a local cost measure c for each pair of features, i. e., $C(n, m) = c(v_n, w_m)$ for $1 \leq n \leq N, 1 \leq m \leq M$. Finally, an optimum-cost alignment path is determined from this matrix via dynamic programming, which encodes the synchronization result. Our synchronization approach follows these lines using the standard DTW algorithm, see [3] for a detailed account on DTW and music synchronization. In our synchronization step, we employ an implementation based on high-resolution audio features that combine the high temporal accuracy of onset features with the robustness of chroma features. These features specifically exploit the above mentioned characteristics of piano music yielding robust music alignments of high temporal accuracy, see [1] for details.

Based on a MIDI-audio alignment, we derive a *tempo curve* that exhibits the expressive tempo information for the music recording. Here, a tempo curve describes for each time position within the reference the associated tempo of the performance in the form of a multiplicative factor. We illustrate this idea by means of an example scenario referring to the first four measures of Beethoven's Sonata Op. 57 ('Appassionata'), see Fig. 2. Here, the score is represented by a MIDI file shown in the piano roll representation. The MIDI representation serves as reference, where the notes are played with a constant tempo in a kind of mechanical way. In our example scenario, we consider three performances of these measures by the three pianists Duis, Barenboim, and Brendel, respectively. The performances are given as audio recordings. For each audio recording, we compute a MIDI-audio alignment with respect to the same MIDI reference and derive a corresponding tempo curve. The tempo curves, as shown in Fig. 2, reveal global and local characteristics of the different performances. For example, it is obvious that Barenboim plays much slower than the other two pianists, in particular at the beginning of the piece. Also note that all performances show a slight acceleration at the beginning and a significant slow-down towards the end.

The actual computation of the tempo curve can be sketched in the following way: First, the segment of the reference for which the tempo should be determined is chosen; it is characterized by its border points v_l and v_r. Using the alignment path, one then computes the semantically corresponding points in the performance w_l and w_r. Now the tempo in this segment is given by $\frac{v_r - v_l}{w_r - w_l}$. The whole process is iterated till all segments of the sequence are covered. The main parameter here is the choice of segment size. Several options are possible, e. g., a fixed size for all segments or an adaptive size that aligns borders with note onsets. One can also perform preprocessing steps to enhance the quality of

the warping path, e. g. by eliminating outliers that are likely to be caused by synchronization errors. Based on such tempo curves, one can then continue with the actual performance analysis.

Performance analysis

Some interesting techniques for processing tempo/dynamics data will be presented in this section. As stated in the introduction, the techniques can be largely divided into approaches concerned with commonalities across interpretations and approaches dealing with differences between various interpretations. In [7], the authors describe an interesting approach that falls into the former category. The objective is to derive elementary *rules of performance* that capture basic principles every performer adheres to[2]. This is done without falling back on domain knowledge; instead, the rules are induced empirically from a large data set of piano music using machine learning methods. In their investigations, the authors reverted to direct annotations which were used as input for the learning algorithms. The employed data set was created specifically for this undertaking by recording 13 complete Mozart piano sonatas performed on a player piano.

One technique that concentrates on capturing systematic differences between artists (even across different pieces) is described in [8, 9]. Here, the objective is to formally specify the basic *musical gestures* individual artists are prone to use.[3] The authors employ an innovative visualization method called the *Performance Worm* as introduced in [2], which is a pseudo-3D depiction of the performance's progression in a tempo-loudness space.

The last approach to be mentioned here allows for the comparative analysis of multiple performances at once, see [4, 5]. Like the previously discussed approach, it features an innovative visualization method referred to as *Scape Plot*[4], which is used to derive semantically interesting features of the analyzed data.

Outlook

Preliminary experiments indicate that our automated methods for deriving tempo curves yield good estimations of the overall tempo, and for piano music of even finer tempo nuances. Problems arise when the input data exhibits errors, which in this case means faulty alignment paths as a result of synchronization problems. Determining whether a given change in the tempo curve is due to an alignment error or the result of an actual tempo change in the performance is not possible in general, although implausible tempo curves may be taken as indicators of a bad synchronization. Here, one idea for future work is to use tempo curves as a means for revealing problematic passages in the music representations where synchronization errors may have occurred with high probability. Another issue for further research concerns the choice of the segment size that determines the resolution of the tempo curve: Larger segments are less susceptible to these errors, but at the same time less sensitive with regard to timing nuances of the performance. Lastly, other performance parameters lend themselves to extraction using alignment information as well. Dynamics curves could be computed analogously to tempo curves, simply by extracting, e. g., the signal's local energy from the music recording at salient points in time indicated by the MIDI reference.

Acknowledgements: The authors are funded by the Cluster of Excellence on Multimodal Computing and Interaction at Saarland University.

[2] Even though this approach used data by a single pianist only.

[3] E. g., combining crescendo and accelerando into one gesture.

[4] Derives from "landscape" paintings, where, according to the author, "the interesting parts lie somewhere in the middle-ground".

References

[1] S. Ewert, M. Müller, and P. Grosche. High resolution audio synchronization using chroma onset features. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, 2009. IEEE.

[2] J. Langner and W. Goebl. Visualizing expressive performance in tempo-loudness space. *Computer Music Journal*, 27(4):69–83, 2003.

[3] M. Müller. *Information Retrieval for Music and Motion*. Springer, 2007.

[4] C. S. Sapp. Comparative analysis of multiple musical performances. In *ISMIR Proceedings*, pages 497–500, 2007.

[5] C. S. Sapp. Hybrid numeric/rank similarity metrics. In *ISMIR Proceedings*, 2008.

[6] Sonic Visualiser. Retrieved 19.03.2009, `http://www.sonicvisualiser.org/`.

[7] G. Widmer. Machine discoveries: A few simple, robust local expression principles. *Journal of New Music Research*, 31(1):37–50, 2002.

[8] G. Widmer. Musikalisch intelligente Computer – Anwendungen in der klassischen und populären Musik. *Informatik Spektrum*, Oct.:363–368, 2005.

[9] G. Widmer, S. Dixon, W. Goebl, E. Pampalk, and A. Tobudic. In search of the Horowitz factor. *AI Magazine*, 24(3):111–130, 2003.