AUTOMATED SYNCHRONIZATION OF SCANNED SHEET MUSIC WITH AUDIO RECORDINGS

Frank Kurth, Meinard Müller, Christian Fremerey, Yoon-ha Chang, and Michael Clausen
Bonn University
Department of Computer Science III

ABSTRACT

In this paper, we present a procedure for automatically synchronizing scanned sheet music with a corresponding CD audio recording, where suitable regions (given in pixels) of the scanned digital images are linked to time positions of the audio file. In a first step, we extract note parameters and 2D position information from the scanned images using standard software for optical music recognition (OMR). We then use a chroma-based synchronization algorithm to align the note parameters to the given audio recording. Our experiments show that even though the output of current OMR software is often erroneous, the music parameters extracted from the digital images still suffice to derive a reasonable alignment with the audio data stream. The resulting link structure can be used to highlight the current position in the scanned score or to automatically turn pages during playback of an audio recording. Such functionalities have been realized as plug-in for the SyncPlayer, which is a free prototypical software framework for bringing together various MIR techniques and applications.

1 INTRODUCTION

Modern digital music libraries contain large amounts of textual, visual, and audio data as well as a variety of associated data representations. In particular for Western classical music, two prominent examples of widely-used and digitally available types of music representations are *scanned sheet music* (available as digital images) and *audio recordings* (e. g., in CD or MP3 format). These two representations complement each other describing music on different semantic levels. On the one hand, *sheet music*, which in our context denotes a printed form of musical score notation, is used to visually describe a piece of music in a compact and human readable from. Sheet music not only allows a musician to create a performance but it also reveals structural, harmonic, or melodic aspects of the music that may not be obvious from mere listening. On the other hand, an *audio recording* encodes the soundwave of an acoustic realization, which allows the listener to play-back a specific interpretation.

Given various representations of musically relevant information, e. g., as encoded by sheet music or as given by a specific audio recording, the identification of semantically related events is of great relevance for music retrieval and browsing applications [1, 4, 5]. Here, we will discuss the problem of *scan-audio synchronization*, which refers to the problem of linking regions (given as pixel coordinates) within the scanned images of given sheet music to semantically corresponding physical time positions within an audio recording. Such linking structures can be used to highlight the current position in the scanned score during playback of the recording, thus enhancing the listening experience as well as providing the user with tools for intuitive and multimodal music exploration. The importance of such a functionality, which is illustrated by Figure 4, has been emphasized in the literature, see, e. g., [4].

In this paper, we present a procedure for scan-audio synchronization, which is, to the best of the author's knowledge, the first algorithm for performing this task in a fully automated fashion. The general idea is to transform both the scanned images as well as the corresponding audio recording into chroma-based mid-level representations, which can then be time-aligned via dynamic time warping. The details of our synchronization approach will be explained in Section 2.

As one important subtask, we extract musical note events as well as the corresponding 2D pixel regions from the scanned sheet music using standard software for optical music recognition (OMR). In Section 3, we will discuss this OMR step and the problems arising from OMR recognition errors. The mid-level representations, as our experiments show, are robust enough to be successfully used for the scan-audio synchronization task even when corrupted by erroneous note events or by missing signatures.

In Section 4, as a further contribution of this paper, we present a novel visualization plug-in for the SyncPlayer framework [6]. This interface synchronously displays the score position within the scanned images along with the audio playback. In the SyncPlayer context, the scan-audio synchronization constitutes a key component for a comprehensive tool facilitating multimodal navigation and retrieval in complex and inhomogeneous music collections.

We conclude this paper with Section 5. Further references to related work as well as prospects on future work are given in the respective sections.

2 SYNCHRONIZATION PROCEDURE

In this section, we describe our approach to scan-audio synchronization and summarize the background on the underlying techniques from MIR and audio signal processing.

As discussed in the introduction, the input of the scan-audio synchronization algorithm consists of a scanned version of a musical score and a corresponding audio recording of the same piece of music. As an example, Figure 1 shows the first few measures of Beethoven's Piano Sonata No. 23, Op. 57 ("Appassionata") as well as the waveform of a recording by Barenboim of the same measures. In the first step of our synchronization algorithm, we transform the scanned score as well as the audio recording into a common mid-level representation, which allows for comparing and relating music data in various realizations and formats. To be more specific, we use chroma-based features, where the *chroma* correspond to the twelve traditional pitch classes of the equal-tempered scale [2]. In Western music notation, the chroma correspond to the set $\{\mathrm{C}, \mathrm{C}^\sharp, \mathrm{D}, \ldots, \mathrm{B}\}$ consisting of the twelve pitch spelling attributes.

The audio recording is transformed into a sequence of normalized 12-dimensional chroma vectors, where each vector expresses the signal's local energy distribution among the 12 pitch classes. Such a chroma representation can be obtained from a spectrogram by suitably pooling Fourier coefficients; for details, we refer to [2]. Figure 1 shows the resulting audio chromagram for our Beethoven example. Chroma-based audio features absorb variations in parameters such as dynamics, timbre, and articulation and closely correlate to the short-time harmonic content of the underlying audio signal.

The transformation of the scanned score representation into a chroma representation consists of several steps. First, the score data such as note events, the key signature, the time signature, and other musical symbols are extracted from the scanned images using standard software for optical music recognition (OMR). Note that in addition to the musical score data, the OMR process provides us with pixel coordinates of the extracted data, allowing us to exactly localize all musical symbols within the scanned images. The technical details and the OMR-involved problems of this step will be discussed separately in Section 3. Then, in the second step, a sequence of chroma features is synthesized from the OMR result. This is done in a straightforward fashion by using the sequence of the extracted note events, which are encoded by parameters for pitch as well as musical onset time and duration. Physical onset times and durations are calculated assuming a constant tempo. Note that in our case the particular choice of tempo is not crucial because differences in tempo will be compensated in the subsequent synchronization step. We choose a standard value of 120 bpm. A sequence of chroma features is then synthesized by sliding across the time axis with a temporal window while adding energy to the chroma bands that correspond to pitches that

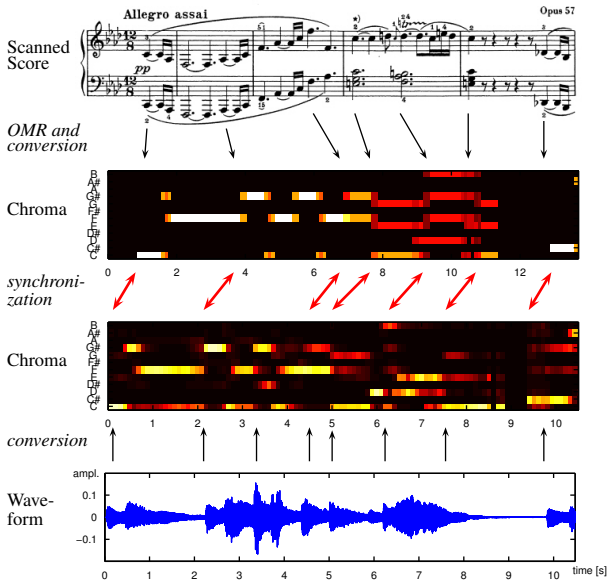

Figure 1. Illustration of the scan-audio synchronization by means of the first few measures of Beethoven's Piano Sonata No. 23, Op. 57 ("Appassionata"). The figure shows a scanned musical score and the resulting score chromagram (upper part) as well as the waveform of a recording by Barenboim of the same measures and the resulting audio chromagram (lower part). The synchronization result between the two chroma representations is indicated by red bidirectional arrows.

are active during the current temporal window. The resulting chroma vectors are then normalized. A similar strategy for synthesizing chromagrams from symbolic data has been suggested in [5]. Figure 1 shows the resulting score chromagram for the Beethoven example.

Having transformed both the audio recording as well as the scanned score into sequences of chroma vectors, one can use standard algorithms based on dynamic time warping (DTW) to time-align the two sequences. For details, we refer to [5, 7] and the references therein. Here, the main idea is to build up a cross-similarity matrix by computing the pairwise distance between each score chroma vector and each audio chroma vector. In our implementation, we simply use the inner vector product for the comparison. An optimum-cost alignment path is determined from this matrix via dynamic programming. To speed up this computationally expensive procedure, we use an efficient multiscale version of DTW. The details of this procedure can be found in [7]. The overall synchronization procedure is illustrated by Figure 1.

To finally link spatial positions within the audio recording to regions within the scanned images, we combine the synchronization and OMR-results as follows: From the pixel coordinates obtained in the OMR step, we derive a correspondence between the extracted OMR note events and the spatial regions within the image data displaying these note events. The spatial regions are encoded by a

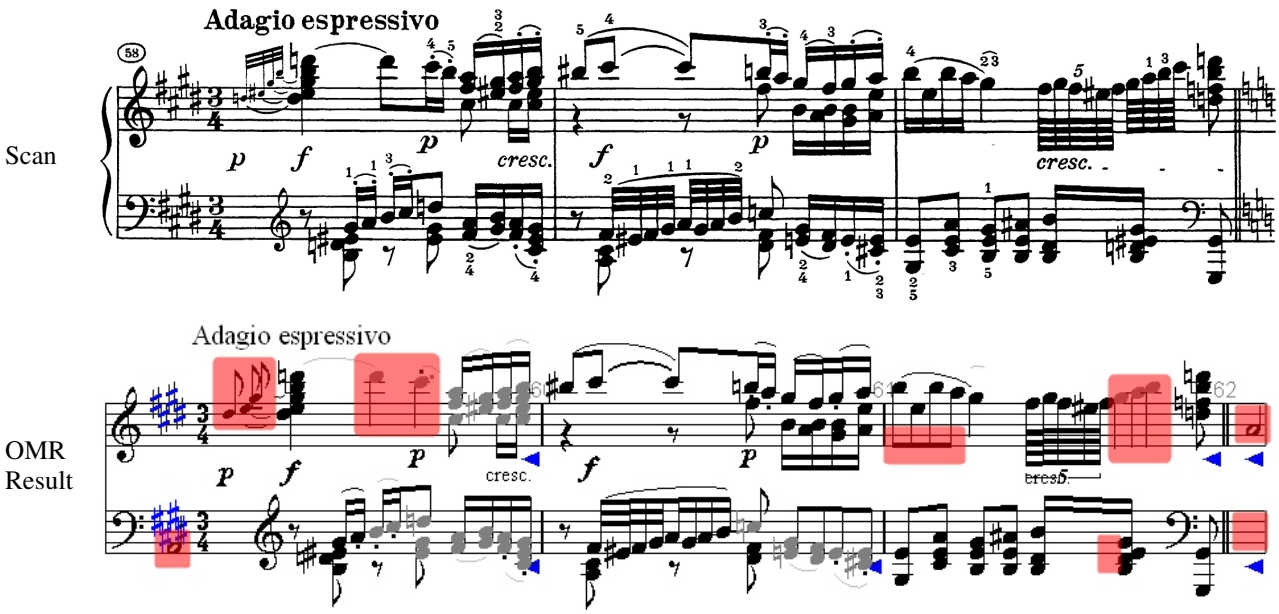

Figure 2. Comparison of a scanned score and the corresponding results acquired by OMR (excerpt of Beethoven's Piano Sonata No. 30, Op. 109). Errors in the OMR result are highlighted by shaded boxes. Triangles at the end of a bar indicate that the sum of the durations of the notes does not match the expected duration of the measure. To correct this, SharpEye has "deactivated" several notes which is indicated by greyed-out note heads. Although in this example, the OMR result shows several recognition errors, there is still enough correspondence to allow for a measure-wise synchronization with a corresponding audio recording.

page number and suitable 2D coordinates. Combining the spatial information and the above synchronization result, we have all linking information needed to track and highlight note events in the scanned score during the playback of the audio recording. Such a functionality will be discussed in more detail in the subsequent sections and is illustrated by Figure 4.

3 OPTICAL MUSIC RECOGNITION

The first step in converting scanned sheet music to chroma features is the recognition of symbols from the given image data and the interpretation of their musical semantics. We use the SharpEye 2.68 software to batch process all scanned pages corresponding to a particular piece of music. The input scanned image files are given in the `TIFF` format using 1 bit color (black and white) and a resolution of 600 dpi. The output is a SharpEye proprietary ASCII file with extension `*.mro` that contains the recognition results and the pixel positions of the recognized symbols within the image data.

The quality of OMR results in general strongly depends on the quality of the input image data and the complexity of the underlying scores. In the context of this paper, we consider high quality scans of piano music. Generally, we obtain reasonably accurate recognition results for this class of music. Despite a good overall quality in the OMR results, some problems may still occur because in score notation, a recognition error of a single symbol can have a high impact on the global semantics. For example, a missing accidental at the beginning of a staff corrupts the pitch of all corresponding notes in that staff. Mistakes in note durations, e.g. the recognition of an eighth as a quarter, can lead to inconsistencies in measure lengths. More severe errors result from misinterpretation of score semantics, e.g., the interpretation of a grand-staff as two individual staffs.

For highlighting regions in the score images one has to choose a suitable spatial and temporal granularity. Even though the scan-audio synchronization is reasonable on a note level resolution for most of the time, we choose a granularity where the displayed region on the score corresponds to one musical measure. This leads to stable and accurately synchronized highlightings even in the case of typical local OMR errors. Furthermore, highlighting entire measures also takes into account that a musician typically follows entire note groups or more general musical structures rather than individual notes.

For testing and evaluating we access data provided by the Bavarian State Library (BSB), Munich, Germany, which is currently digitizing audio recordings and sheet music in the context of the PROBADO digital library project [8]. The underlying music collection consists of classical and romantic piano sonatas as well as a collection of German 19th centuries piano songs amounting to approximately 6.000 images of scanned music and about 1.200 pieces of music in total. For each scanned piece

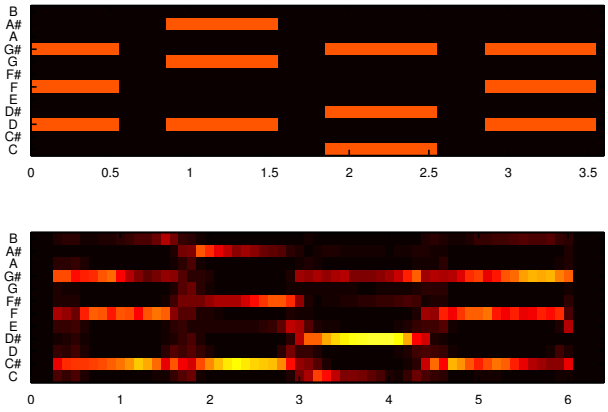

Figure 3. Comparison of two chromagrams. The lower chromagram is derived from an audio recording of a cadenza in D♭-Major, played on a piano. The upper chromagram is synthesized from a symbolic score representation that has missing accidentals. In particular, there are only three flats instead of five. The optimum alignment still delivers an accurate synchronization on the basis of the unaffected pitches.

of music, three different audio recordings are currently digitized at BSB. We have tested our algorithm for scan-audio synchronization on a selection of piano music by Beethoven, Mozart and Schubert comprising more than 160 pages of sheet music.

The OMR works well in most cases but often suffers from minor glitches like missing accidentals or mistakes in note durations. However, because the detection of bar lines is quite stable, inconsistencies and confusions due to mistakes in note durations last no longer than the measure of their occurrence. As an example, Figure 2 shows a comparison of a scanned score and the corresponding results acquired by OMR for an excerpt of Beethoven's Piano Sonata No. 30, Op. 109. Local errors in the OMR result are highlighted by shaded boxes. Our experiments show that for a global measure-wise synchronization, these local errors have a negligible influence as they are mainly absorbed by the coarseness of our chroma features.

The possible impact of missing accidentals is illustrated in Figure 3, where two chromagrams are compared. The lower chromagram is derived from an audio recording of a cadenza in D♭-Major played on a piano. The upper chromagram is synthesized from a symbolic score representation that suffers from two missing accidentals (three instead of five flats). Despite of these severe recognition errors, the unaffected pitches still overrule the effect of the corrupted pitches facilitating an accurate overall synchronization of the two chromagrams.

In case of the previously discussed OMR errors of higher semantic impact, such as the confusion of a two-staff system with two one-staff systems, regional synchronization errors may occur. However, due to our DTW-based synchronization approach, where the global scan-audio alignment is calculated, those errors affect the synchronization only locally in a neighborhood around the corresponding regions.

We conclude this section by noting that a considerable amount of OMR errors can be corrected in a postprocessing step by using suitable heuristics. As an example, the accidentals detected at the beginning of each staff may be checked for correctness by considering the global key or heuristics on detected accidentials in previous and subsequent staffs. As another approach, OMR may be improved by combining different OMR strategies as suggested by Byrd et al. [3].

4 A SYNCPLAYER PLUG-IN FOR AUTOMATIC SCORE-TRACKING

In this section, we present a prototypical implementation of a graphical interface for automatic score tracking. The interface has been implemented as a visualization plug-in of the previously proposed SyncPlayer framework [6], which is briefly described in Section 4.1. Section 4.2 describes the novel plug-in for automatic score tracking.

4.1 The SyncPlayer Framework

The SyncPlayer is a client-server based software framework that integrates various MIR applications such as multimodal presentation of audio data, music synchronization, and content-based retrieval [6]. The user operates the *client component*, which in its basic mode acts like a standard software audio player for recordings in the `*.mp3` and `*.wav` file formats. Figure 4 shows the SyncPlayer client with the main window located at the top left. A remote computer system runs the *server component*, which supplies the client with content-related data associated to the currently selected audio recording. Examples of such content-related data are lyrics of the vocal tracks, the musical score, or information on the musical form. Synchronously to acoustic audio playback, such data is then presented to the user by means of several specialized *visualization plug-ins*, which are available at the client side. In the next section, we describe how the SyncPlayer is extended by a novel plug-in for visualizing scanned sheet music where highlighting of page regions is provided to enable score tracking during audio playback. In Figure 4, three instances of this novel plug-in are depicted along with the audio player. For more details on the SyncPlayer and existing plug-ins, we refer to our web page [9].

4.2 A Plug-in for Automatic Score Tracking

For our SyncPlayer-based user scenario we assume that after suitable digitization and scan-audio synchronization as discussed in the previous sections, the scanned sheet music as well as the information linking the scans to the audio recordings are available at the SyncPlayer server. A typical scenario of a user operating the SyncPlayer client then involves the following steps:

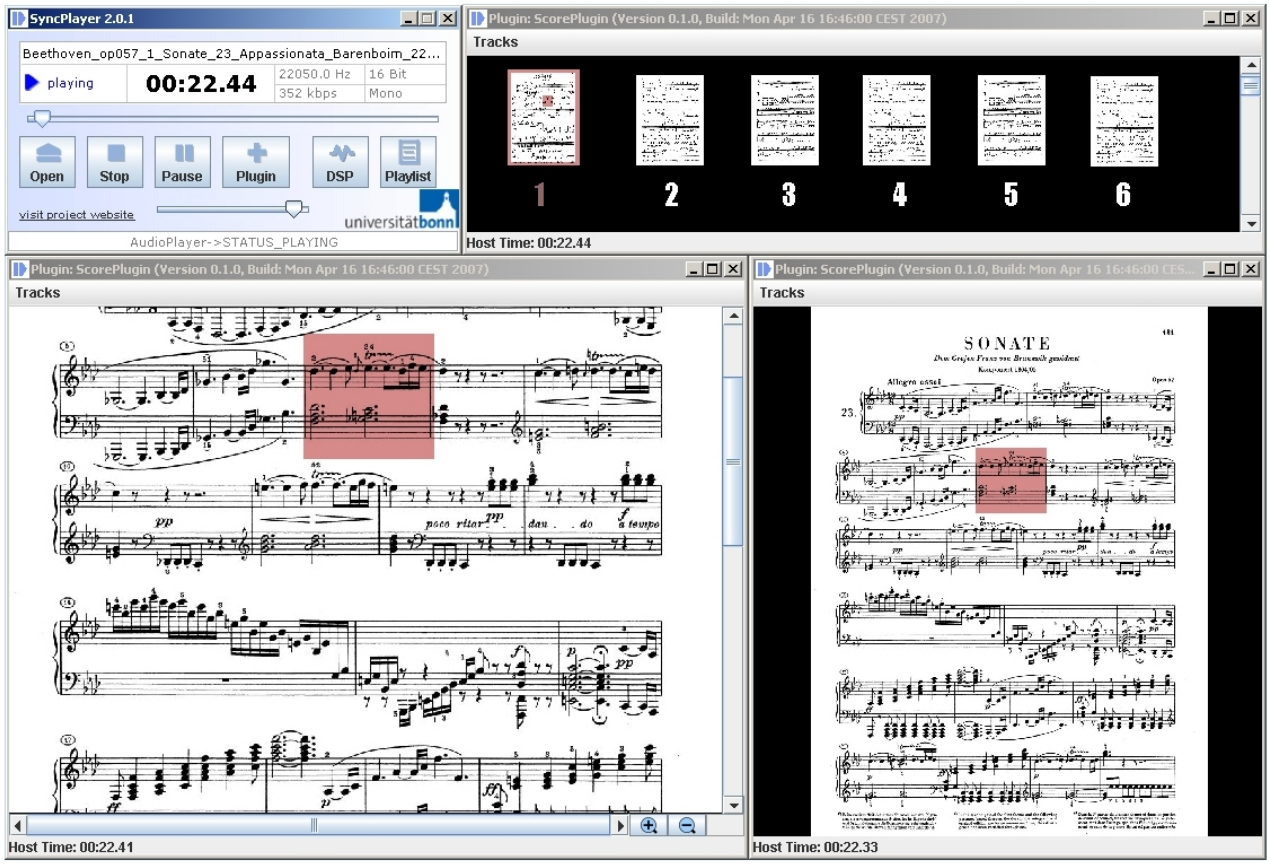

Figure 4. SyncPlayer client and three instances of the SheetMusic plug-in for displaying scanned sheet music. The bar in the scanned score that corresponds to the current playback position is highlighted by a red box. The instances demonstrate three different zoom levels: detailed view (lower left), full page view (lower right) and thumbnail view (upper right).

1. The user selects and opens an audio recording in the SyncPlayer Client.

2. A connection to the SyncPlayer Server is established allowing the server to identify the selected audio recording within the audio database. If available, the score image data and the bar highlighting information on the audio recording are retrieved from its annotation database.

3. The server notifies the client about the availability of synchronized score image data for the selected recording. If available, the client retrieves and displays the available data in the SheetMusic plug-in.

In its basic mode, the *SheetMusic plug-in* displays one page of scanned sheet music as an image (Figure 4, lower left and lower right plug-in windows). The plug-in offers controls for zooming and scrolling within the boundaries of the page. During the playback of a recording in the SyncPlayer, the bar that is currently played is highlighted in the score by a red translucent box. If the current bar is located on a different page or lies outside the visible display area, the plug-in automatically loads and displays the corresponding page and adjusts the scroll settings to ensure that the bar is visible.

The plug-in implements several zoom level strategies, each one offering a different tradeoff between detail and overview. The *detailed view* shows only a local part of the current page surrounding the highlighted bar. The scrolling functionality is restricted to the page boundaries (Figure 4, lower left window). The *full page view* fixes the zoom settings to always displaying one full page to give a better overview on where the current playback position is located on the current page (Figure 4, lower right window). An overview on the entire set of pages belonging to a piece of music is given in the *thumbnail view*. This view displays several pages at a time, highlighting the currently active page by a red frame (Figure 4 upper right). Additionally, the page numbers are displayed.

The SheetMusic plug-in not only has visualization capabilities, but also offers the user several ways to navigate within the score and audio representations. By clicking on a target bar, the user may request the plug-in to set the playback of the audio recording to the starting time of that particular measure. When the user selects a page in the thumbnail view, the player jumps to the playback position

corresponding to the start of the page.

In the process of developing and testing the SheetMusic plug-in, we experimented with other styles of highlighting suitable score regions. A sliding window, which moves across the score in a note-by-note fashion, is problematic in the current version because small (e.g., note-level) inaccuracies in the synchronization are likely to confuse the user. Furthermore, as noted before, musicians usually tend to seize entire note groups and musical structures rather than following individual notes. As another visualization strategy, besides highlighting the currently active measure, several subsequent measures could be visualized to allow a certain look-ahead. In future work, several other visualization strategies have to be implemented and evaluated on the basis of comprehensive user studies.

5 CONCLUSIONS

In this paper, we presented an automated procedure for scan-audio synchronization, which consists of aligning scanned sheet music with corresponding CD audio recordings. In the alignment, spatial regions of the sheet music are linked to musically corresponding temporal regions within the audio recording. In our approach, we used chroma-based features as a mid-level representation for encoding both the audio recording and the scanned sheet music, which allowed us to perform the synchronization step using standard DTW techniques. The use of this mid-level representation allows us to exploit both the robustness of chroma-based features to compensate for local OMR errors and the robustness of the global DTW to compensate more global staff-level OMR errors. The evaluation of the proposed methods is carried out on a real-world corpus of digitized sheet music and audio recordings currently digitized at the Bavarian State Library.

For presenting the results of a scan-audio synchronization to a user, we implemented a SheetMusic plug-in for the existing SyncPlayer framework. The plug-in offers several different modes for displaying scanned score images along with audio playback and allows to synchronously highlight (and hence track) the current playback position in the scanned score and to automatically turn pages.

As a next step, a more formal and exhaustive evaluation of the system should be conducted, especially regarding the impact of different classes and rates of OMR errors on the synchronization results. However, because the focus of this paper lies on the overall concept and interaction of components, such an evaluation is outside this paper's scope. A formal evaluation is planned to be a topic in our future work.

Acknowledgement

This work was supported in part by Deutsche Forschungsgemeinschaft (DFG) under grant 554975 (1) Oldenburg BIB48 OLof 01-02. We thank the project partners at BSB for providing the digitized material used within this research project.

6 REFERENCES

[1] V. ARIFI, M. CLAUSEN, F. KURTH, AND M. MÜLLER, *Synchronization of music data in score-, MIDI- and PCM-format*, Computing in Musicology, 13 (2004).

[2] M. A. BARTSCH AND G. H. WAKEFIELD, *Audio thumbnailing of popular music using chroma-based representations*, IEEE Trans. on Multimedia, 7 (2005), pp. 96–104.

[3] D. BYRD AND M. SCHINDELE, *Prospects for improving OMR with multiple recognizers*, in Proc. ISMIR, Victoria, Canada, 2006, pp. 41–46.

[4] J. W. DUNN, D. BYRD, M. NOTESS, J. RILEY, AND R. SCHERLE, *Variations2: Retrieving and using music in an academic setting*, Special Issue, Commun. ACM, 49 (2006), pp. 53–48.

[5] N. HU, R. DANNENBERG, AND G. TZANETAKIS, *Polyphonic audio matching and alignment for music retrieval*, in Proc. IEEE WASPAA, New Paltz, NY, October 2003.

[6] F. KURTH, M. MÜLLER, D. DAMM, C. FREMEREY, A. RIBBROCK, AND M. CLAUSEN, *SyncPlayer—an advanced system for content-based audio access*, in Proc. ISMIR, London, GB, 2005.

[7] M. MÜLLER, H. MATTES, AND F. KURTH, *An efficient multiscale approach to audio synchronization*, in Proc. ISMIR, Victoria, Canada, 2006, pp. 192–197.

[8] T. STEENWEG AND U. STEFFENS, *Probado – nontextual digital libraries put into practice*, ERCIM News, Special Theme: European Digital Library (2006), pp. 47–48.

[9] SYNCPLAYER, *An advanced system for multmodal music access*. Website, January 2007. `http://www-mmdb.iai.uni-bonn.de/projects/syncplayer`.