Towards Bridging the Gap between Sheet Music and Audio

Christian Fremerey[1], Meinard Müller[2], Michael Clausen[1]

[1] Universität Bonn, Institut für Informatik III
Römerstr. 164, 53117 Bonn, Germany
fremerey@iai.uni-bonn.de, clausen@iai.uni-bonn.de
[2] Saarland University and MPI Informatik
Campus E1-4, 66123 Saarbrücken, Germany
meinard@mpi-inf.mpg.de

Abstract. Sheet music and audio recordings represent and describe music on different semantic levels. Sheet music describes abstract high-level parameters such as notes, keys, measures, or repeats in a visual form. Because of its explicitness and compactness, most musicologists discuss and analyze the meaning of music on the basis of sheet music. On the contrary, most people enjoy music by listening to audio recordings, which represent music in an acoustic form. In particular, the nuances and subtleties of musical performances, which are generally not written down in the score, make the music come alive. In this paper, we address the problem of bridging the gap between the sheet music domain and the audio domain. In particular, we discuss aspects on music representations, music synchronization, and optical music recognition, while indicating various strategies and open research problems.

Keywords. audio, sheet music, symbolic score, optical music recognition, music synchronization

1 Introduction

The last years have seen increasing efforts in building up large digital music collections, which contain large amounts of textual, visual, and audio data as well as a variety of associated data representations. In particular for Western classical music, three prominent examples of digitally available types of music representations are *sheet music* (available as digital images), *symbolic score data* (e. g., in the MusicXML or the LilyPond format), and *audio recordings* (e. g., given as WAV or MP3). These three classes of representations complement each other describing music on different semantic levels. Sheet music, which in our context denotes a printable form of musical score notation, is used to visually describe a piece of music in a compact and human readable form. This form allows musicians to create a performance and musicologists to study structural, harmonic, or melodic aspects of the music that may not be obvious from mere listening. Symbolic score data can be parsed by computers and can be used

Dagstuhl Seminar Proceedings 09051
Knowledge representation for intelligent music processing
http://drops.dagstuhl.de/opus/volltexte/2009/1965

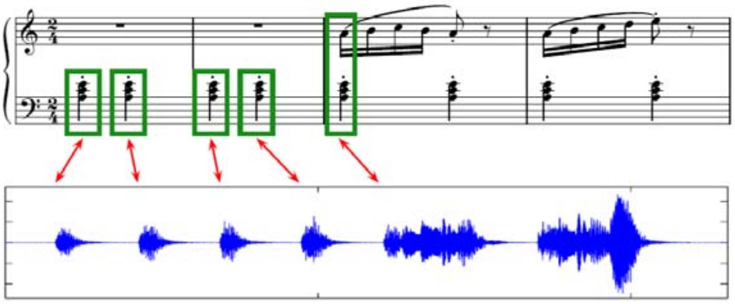

Fig. 1. Illustration of the sheet music-audio synchronization by means of the first four measures of Op. 100, No. 2 by Friedrich Burgmüller. The figure shows a scanned musical score and the waveform of a recording of the same measures. The synchronization is indicated by red bidirectional arrows linking regions (given as pixel coordinates) within the scanned image and physical time positions within the audio recording.

to perform automated analysis tasks and to support larger scale analysis tasks that could not be done manually. Finally, an audio recording encodes the soundwave of an acoustic realization, which allows the listener to playback 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]. In this paper, we discuss the problem of *sheet music-audio synchronization*, which refers to the problem of linking regions (given, e. g., as pixel coordinates) within the given sheet music document to semantically corresponding physical time positions within an audio recording, see Fig. 1. Such linking structures can be used to highlight the current position in the sheet music document during playback of the recording, thus enhancing the listening experience as well as providing the user with tools for intuitive and multimodal music exploration, see Fig. 2. The importance of such a functionality has been emphasized in the literature, see, e. g., [2].

In the last few years, first methods have been proposed for automatically aligning and matching sheet music and audio material [6, 7]. Here, one possible processing pipeline is to extract musical parameters from the scanned sheet music using optical music recognition (OMR) methods as well as from the audio recordings using signal processing methods. Based on the extracted parameters, one can then use alignment techniques for synchronizing the scanned data and the audio material. One may think of other processing pipelines depending on the types of available data. For example, starting with symbolic score data (e.g., MusicXML, LilyPond), one can generate visual music representations. In this case, there is no need for employing an error-prone OMR step and the synchronization task can be performed on the basis of the explicitly given symbolic information.

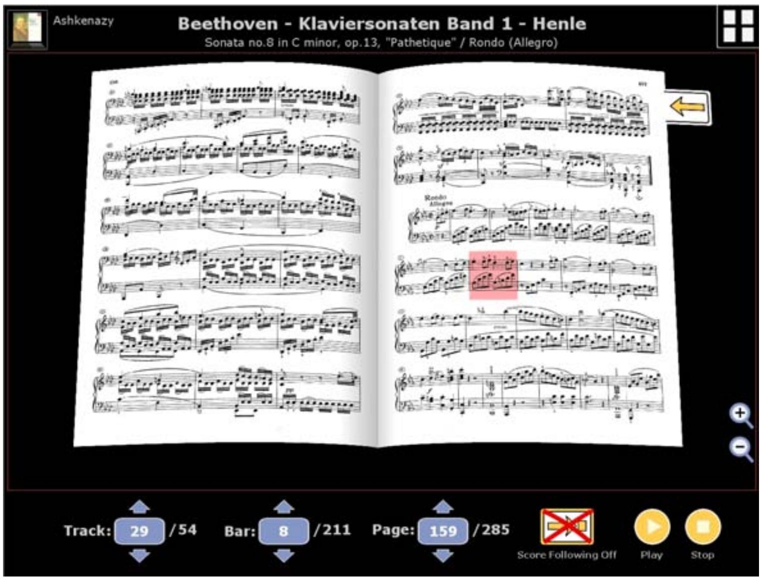
Ashkenazy
Beethoven - Klaviersonaten Band 1 - Henle
Sonata no.8 in C minor, op.13, "Pathetique" / Rondo (Allegro)
Rondo
Allegro
Track: 29 /54
Bar: 8 /211
Page: 159 /285
Score Following Off
Play
Stop

Fig. 2. The Score Viewer interface for multimodal music presentation and navigation (from [5]). Synchronously to audio playback, corresponding musical measures within the sheet music are highlighted.

In the remainder of this paper, we discuss these issues in more detail. In Sect. 2, we summarize some basic properties of various music representations and formats. Then, in Sect. 3, we outline strategies for aligning sheet music and audio recordings as introduced in [7]. In the case of scanned sheet music, the quality of the synchronization results crucially depends on the OMR extraction results. Therefore, in Sect. 4, we address the issue of optical music recognition in more detail. In particular, we analyze the various kinds of recognition errors and indicate how these errors may be dealt with in a subsequent postprocessing step. Finally, in Sect. 5, we give prospects on future work and sketch possible improvements. Further related work is discussed in the respective sections.

2 Music Representations

In this paper, we distinguish between three main classes of music representations: *Audio*, *Symbolic* and *Sheet Music*, see Fig. 3. The entity *Audio* stands for audio recordings as given in formats such as WAV or MP3. The entity *Symbolic* stands for any kind of symbolic representation of a score including MIDI, MusicXML, Humdrum, or LilyPond. Finally, the entity *Sheet Music* stands for visual representations of a score as encoded in TIFF, PDF, or other image format.

Actually the boundaries between these classes are not as sharp as the illustration in Fig. 3 might suggest. In fact, the clustering of music representations

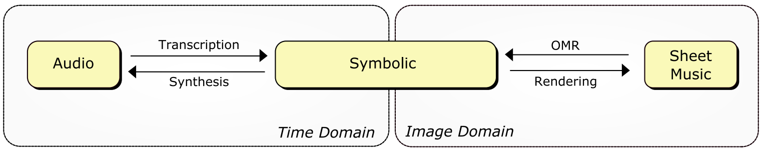

Fig. 3. Illustration of three classes of music representation used in this paper: Audio, Symbolic and Sheet Music.

into classes is a matter of choosing a definition of where to draw the lines between classes. The term "symbolic" is not very descriptive by itself. Actually any kind of digital data uses symbols to represent abstract entities of some data model. In this paper, we use the term *symbolic* to refer to any data format that explicitly represents musical entities.

The musical entities may range from timed note events as is the case in MIDI files to graphical shapes with attached musical meaning as is the case in the SCORE engraving system. Examples for symbols that do not represent musical entities are audio samples in audio files or pixels in bitmap image files. Also the graphical shapes in vector graphics representations of scores are not considered to be musical entities as long as they do not additionally specify the abstract musical entity represented by that shape. Certainly, there is a wide range of what to consider as symbolic music, see [8] for a detailed discussion. However, our rough and intuitive classification should suffice for the discussion to follow. Some examples for data formats for symbolic score data are MusicXML, LilyPond, Humdrum, NIFF, MIDI and SCORE. Example formats for audio data are WAV and MP3. Example formats for sheet music documents are BMP, TIFF and PDF.

Note that audio data is considered to live in the time domain, i.e., positions and regions are measured in units of time. On the contrary, sheet music is considered to live in the image domain, i.e., positions and regions are specified in the two-dimensional Euclidean space. Symbolic score data can contain both time domain and image domain information. Actually, it is in the symbolic realm where the transition between the image domain and time domain is made.

Having specified the meaning of music representation classes, we now identify transformations between these classes. In Fig. 3, these transformations are indicated by arrows. The transformation of sheet music documents into symbolic score data is commonly referred to as *optical music recognition* (OMR). This includes, for example, the recognition of musical symbols from scans of printed sheet music pages. The reverse transformation, i.e., the creation of a sheet music image from symbolic score data, is called *rendering*. Depending on how detailed geometric information is given in the symbolic score data, this step may or may not require the application of typesetting and engraving rules. The task of creating symbolic score data from an audio recording is called *transcription*. A transformation in the reverse direction is called *synthesis*. Note that, in general, these transformations are not required to preserve all information and therefore may not be reversible.

In many music libraries, scores are mostly available in form of printed sheet music. To be able to process the scores with computers, one has to digitize the prints through scanning and employ optical music recognition (OMR) to extract symbolic score data from the images. In recent years, the digitization of documents in libraries has been a major topic in the library communities. Most libraries have started to digitize their content in one way or another. However, most of the generated data is not freely available due to copyright and other restrictions. In the last years, an increasing number of freely available repositories of digital sheet music have come into existence. For example, the Mutopia [9] and IMSLP [10] project aim at generating and supplying such data. In view of academic research, the availability of music data without any copyright restrictions is of crucial importance. A data format for which no data is available or for which the data is legally protected is of very limited value for most researchers.

3 Sheet Music-Audio Synchronization

The goal of sheet music-audio synchronization is to link regions within the two-dimensional image domain of sheet music documents to semantically corresponding temporal regions in audio recordings, see Fig. 1. One may choose one of several options for the granularity of the regions that are to be linked, for example, pages, lines, bars or notes. Depending on the choice of granularity, there are different strategies to perform the synchronization task. For example, in case of a page-wise synchronization the manual creation of links between the pages of sheet music and corresponding temporal regions of audio recordings may still be acceptable. However, aiming at finer grained resolutions, the complexity for creating the links increases significantly, hence requiring automated synchronization methods.

The respective strategy for achieving a sheet music-audio synchronization very much depends on the type of the given input data, see Fig. 4. In the following, we consider two scenarios that are of great practical importance. In the first scenario, one is given an audio recording and a sheet music document (e.g., a scan). Using optical music recognition, symbolic score data is generated from the sheet music. In the second scenario, one is given an audio recording and symbolic score data (e.g., in the LilyPond format). In this case, a sheet music image has to be rendered from the symbolic score data.

In both scenarios, the connection between the audio data and the symbolic data is realized through the same mechanisms. The basic idea is to transform both the symbolic score data as well as the corresponding audio recording into a common mid-level representation, which can then be synchronized based on standard alignment techniques such as dynamic time warping [4]. In the synchronization context, chroma-based music features have turned out to be a powerful mid-level music representation [11, 3, 4].

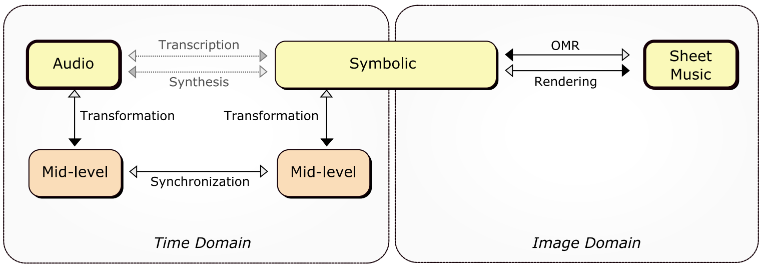

Fig. 4. Illustration of data types and data transformations relevant for sheet music-audio synchronization. Double-headed arrows with one solid and one non-solid head indicate a (possibly lossy and error-prone) data transformation towards the direction of the solid head that implicates a two-way mapping from regions in one data type to regions in the other data type. For example in the case of OMR, one can derive symbolic score data from a sheet music document, implicating a mapping between symbols in the score data and corresponding locations or regions in the sheet music document. The double-headed arrow with two non-solid heads for the synchronization indicates a two-way mapping without any data transformation. The arrows for transcription and synthesis are greyed out, because they are not in the focus of this work.

Note that in both scenarios, one based on OMR and the other based on rendering, one requires an explicit mapping between the musical objects given by the symbolic representation and the 2D coordinates of their depicted counterparts in the image representation. Using this mapping and the synchronization result, a correspondence between spatial regions in the sheet music and temporal regions in the audio recording can be derived.

The quality of the resulting synchronization depends on several factors. In particular, differences between the audio and score representation may have a crucial impact on the final synchronization result. Here, such differences may be due to extraction errors in the OMR step. Furthermore, the actual interpretation may deviate from the notated score. In the following, we present of list of typical classes of differences, each class having a different impact on the final synchronization result.

- **Differences in structure (repeats, jumps):** The score and audio representations might disagree on the level of their global structure. For example, the score might contain a section that is not played in the audio recording or the audio recording might contain an extra repeat that is not present or not recognized in the score. Differences in structure may violate the boundary and monotonicity assumptions made in dynamic time warping, see [4]. Such differences may be handled in a preprocessing step or by partial matching strategies [12].
- **Local dissimilarities:** A typical example for local dissimilarities are musical events in the audio and sheet music representations with deviating pitch or duration. Problematic are also note ambiguities in the score such

as arpeggios, trills, grace notes, or other ornaments. Generally, differences of this class tend to have little impact on the synchronization result as long as they stay local and are enclosed by sections without mismatches and errors.

- **Significant differences in tempo:** For computing a mid-level representation from a symbolic music representation, one needs to decide on the tempo to use in the tranformation from musical onset times like beats and bars to physical onset times like seconds and milliseconds. Since tempo directives of music notation are often not output by OMR systems, the tempo then has to be guessed or estimated. For classical music, the tempo can vary over a wide range from about 25 to 200 BPM (beats per minute). Differences between the estimated tempo and the actual tempo of the audio recording are usually handled by the DTW-based alignment strategy. However, dynamic time warping starts to loose flexibility and accuracy when the tempo differences become too large.
- **Differences in loudness and timbre:** Such differences usually have little impact on the synchronization when using normalized chroma features as mid-level representation. Such features show a high degree of robustness to variations in timbre and articulation [11, 4]. Furthermore, normalizing the features makes them invariant to dynamic variations. However, there may be cases where changes in timbre may cause effects in the frequency spectrum that can not be handled by chroma features. In particular, in the presence of percussive elements and in the case of non-tonal elements other feature types are needed.

Besides the differences between score and audio data as mentioned above, the accuracy of the synchronization also crucially depends on the properties of the chosen mid-level feature representations. Here, the feature resolution and, even more important, the features' ability for capturing local characteristics of the underlying music is of foremost importance. For example, chroma features do well in the case of harmony-based music based on the equal-tempered scale when the chroma distribution does not stay constant over a long period. However, in situations where the chroma distribution stays mainly constant, e.g., when having the same chord been played over and over again, the chroma features may seriously fail.

4 Optical Music Recognition

As already pointed out in Sect. 2, the term optical music recognition (OMR) is commonly used to refer to transformations from the sheet music domain to the symbolic score domain. Actually, there is a wide spectrum of transformations depending on the kind of input and output data. For example, starting with a vector graphics image of a score, the task of assigning labels for musical entities such as `note head`, `stem`, `treble clef`, `staff line` to each of the shapes could already be considered as OMR. However, OMR usually covers a lot more than that. For example, when starting with a bitmap image, one crucial subtask of

OMR consists in grouping the pixels to musically meaningful shapes and relating these shapes to musical entities.

Another question is how *far* the transformation extends into the symbolic realm: Where does the task of OMR end? Many OMR systems stop after having identified most of the symbols and having interpreted their basic meaning and relations. Symbols that are not recognized are often ignored. Furthermore, higher level semantics are often neglected. For example, an OMR system might recognize repeat symbols, but not necessarily does it also interpret their musical meaning (i.e., the implied jumps and repeats to be considered when playing the piece). As another example, consider text-based information contained in sheet music. Most OMR systems are able to recognize text and even to distinguish between text that is meant to be song lyrics and text that it not meant to be song lyrics. The systems usually do not reveal the meaning of the text elements, e.g., specifying them as title heading, section name, tempo directive, jump directive (dacapo, fine, etc.) or part name. Thus, higher-level semantic information on tempo, repeats, jumps (dacapo, alternative endings), voicing, or link-ups of staff systems over several consecutive pages are not output by OMR systems.

The various OMR systems may differ significantly with respect to their output. For example, an OMR system might output a sequence of abstract musical symbols and relations without giving any information about the positions and shapes of these symbols in the original image. Due to the loss of such information, it is impossible to reverse the transformation. However, recall that a kind of reversibility is required in the sheet music-audio synchronization scenarios described above. Several commercial and non-commercial OMR software systems exist.[1] Three of the more popular commercial systems that operate on common Western classical music are SharpEye, SmartScore, and PhotoScore. Two of the more prominent examples of free OMR systems are Gamera and Audiveris. However, a drawback of using Gamera for OMR is that it cannot be used out-of-the-box because it requires training to be performed on the data to be recognized. Audiveris currently is not competitive in terms of recognition rates compared to the commercial products. Evaluating and comparing the general performance of OMR systems is a non-trivial task. Some methods for lower-level comparison and evaluation have been proposed in the literature [13–16]. In general, the quality of OMR results strongly depends on the quality of the input image data and the complexity of the underlying scores. More complex scores tend to result in more OMR extraction errors. Here, typical OMR errors are non-recognized symbols, missing accidentals (also in key signatures), missing beams, missing or extra barlines, wrong time signatures, and splitting of grand staffs.

Assuming some extra constraints on the type of sheet music, one may use specialized rules to identify and auto-correct some of these errors. For example, in case of piano music, different key signatures in the left-hand and right-hand staff

[1] Online lists can be found at `http://www.informatics.indiana.edu/donbyrd/OMRSystemsTable.html` and `http://www.music-notation.info/en/compmus/omr.html`

are typically caused by OMR errors. As it turns out, it is much more likely that accidentals are missed by OMR rather than that extra accidentals are added. Therefore, a good strategy to handle different key signatures in the left and right hand is to choose the one with more accidentals. Another example are key signatures that change with the start of a new system without being announced at the end of the previous system. Such a situation most likely indicates that either the recognized key signatures are inconsistent or the announcement has been missed out. In the second case, some unused space is left at the end of the respective staff line, which in turn is very unlikely for professionally typeset scores. Using such heuristics, certain OMR errors may be easily fixed in a postprocessing step.

Even without using very strict constraints on the type of sheet music, the use of higher-level semantic knowledge can help to identify and correct problems in the OMR data, especially for scores with more than one staff per system. For example, mismatching time signatures are most likely caused by recognition errors. Bars inside which the accumulated duration disagrees with the time signature, reveal problems in the recognized durations of notes and rest, the division into voices or the active time signature. As another example, one can exploit the fact that the horizontal positions of notes and rests are strong indicators for the relative onset times. The horizontal spacing of notes and rests is strongly correlated with their durations. Furthermore, having several voices, the accumulated durations within each voice should coincide at every position where more than one voice has an onset. This information can be used to locate errors in recognized note durations, and in many situations, the information is sufficient for inferring corrections of note durations or time signatures.

Most OMR programs seem to make only very little use of the strong interdependencies between the musical entities of sheet music. One reason for that might be that the interdependencies are very rich and complex spreading over several semantic levels. For example, the decision if a set of black pixels that lie on a vertical line on top of a staff represent a barline or a note stem depends on other decisions such as the existence of a nearby note head. This, in turn, depends on already having recognized the note head, which again may depend on several other high-level and low-level decisions. Most of the common OMR approaches follows a fixed processing pipeline to get from lower level entities to higher level entities. Therefore, high-level entities do not influence the decisions on the lower levels. To address this shortcoming, first mechanisms were introduced in [17] that allow higher-level steps to give feedback to lower-level steps.

In general, to get from a sheet music image to a high-level symbolic score representation, a lot of decisions have to be made. Some of the decisions involve the abstraction from sets of pixels to shapes representing musical entities. Others are about relationships between musical entities, about their functions or about the formation of even higher-level entities. The following list gives some examples of decisions that might have been made:

- Pixelset P_1 represents a line ℓ_1.
- Line ℓ_1 represents a staff line.

- Line ℓ_1 represents the top line of a 5-line staff f_1.
- Staffs f_1 through f_3 form a system s_1.
- Pixelset P_2 represents a treble clef.
- Horizontal region r_1 of a system s_1 represents a bar b_1.
- Bars b_1 through b_{165} form a track t_1.
- Track t_1 has three parts p_1, p_2 and p_3 (e.g. *Violin*, *Cello* and *Piano*).
- Set of notes and rests N_1 in bar b_1 make up a voice v_1.
- Voice v_1 belongs to part p_1.
- Pixelset P_3 represents a letter x_1.
- Set of letters X_1 forms a text y_1.
- Text y_1 represents a title heading h_1.

As can be seen from the examples, the decisions about the involved entities are strongly interdependent. Using these interdependencies, a human reader is able to quickly infer these decisions with a very high degree of certainty. But when seen in an isolated fashion, as it is often done in optical music recognition systems, most of these decisions can only be made with a low degree of certainty. Therefore, in order to improve optical music recognition, one has to avoid making hard decisions before exploiting the interdependencies. Furthermore, music notation for Western classical music obeys a rich set of high-level rules and practices that is of key importance for inferring decisions in OMR with a very high degree of certainty, and therfore should be incorporated into OMR systems.

5 Conclusions and Future Work

In this paper, we have discussed various strategies on how to bridge the gap between various music representations by automatically finding semantically meaningful correspondences between various instances of the same piece of music. In particular, when dealing with sheet music, one depends on recognition procedures to extract musical entities from the image data. Besides the accuracy of the OMR output, the quality of the synchronization results depends on other factors such as differences in structure, significant differences in tempo, as well as the choice of suitable mid-level features. Sheet music-audio synchronization as described in this paper is implemented in the PROBADO project as one aspect of organizing and presenting music in a digital library [5, 18].

We conclude this paper with a selection of open research questions that should be pursued in future work. How can structural differences (caused by repeats, jumps, cuts, etc.) in the various music representations be handled? How can sheet music books be automatically segmented into meaningful units such as sonatas and movements? Which temporal accuracy (e.g., page-wise, bar-wise, note-wise) is needed for practical application? How can interdependencies between entities of music notation be exploited in the OMR process? How can visual and auditory modalities be combined in user interfaces? Which kind of user studies can be conducted for identifying the users needs?

6 Acknowledgements

We would like to express our gratitude to Donald Byrd, Ian Knopke, and Laurent Pugin for stimulating discussion and feedback.

References

1. Arifi, V., Clausen, M., Kurth, F., Müller, M.: Synchronization of music data in score-, MIDI- and PCM-format. Computing in Musicology **13** (2004)
2. Dunn, J.W., Byrd, D., Notess, M., Riley, J., Scherle, R.: Variations2: Retrieving and using music in an academic setting. Special Issue, Commun. ACM **49** (2006) 53–48
3. Hu, N., Dannenberg, R., Tzanetakis, G.: Polyphonic audio matching and alignment for music retrieval. In: Proc. IEEE WASPAA, New Paltz, NY. (2003)
4. Müller, M.: Information Retrieval for Music and Motion. Springer (2007)
5. Kurth, F., Damm, D., Fremerey, C., Müller, M., Clausen, M.: A framework for managing multimodal digitized music collections. In: Proceedings of the 12th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2008). (2008)
6. Fremerey, C., Müller, M., Kurth, F., Clausen, M.: Automatic mapping of scanned sheet music to audio recordings. In: Proc. ISMIR, Philadelphia, USA. (2008)
7. Kurth, F., Müller, M., Fremerey, C., Chang, Y., Clausen, M.: Automated synchronization of scanned sheet music with audio recordings. In: Proc. ISMIR, Vienna, Austria. (2007) 261–266
8. Selfridge-Field, E., ed.: Beyond MIDI: The Handbook of Musical Codes. MIT Press, Cambridge, MA, USA (1997)
9. Mutopia Project: Music free to download, print out, perform and distribute. `http://www.mutopiaproject.org` (2006)
10. International Music Score Library Project: International music score library project (IMSLP) portal. `http://imslp.org/` (2009)
11. Bartsch, M.A., Wakefield, G.H.: Audio thumbnailing of popular music using chroma-based representations. IEEE Trans. on Multimedia **7** (2005) 96–104
12. Müller, M., Appelt, D.: Path-constrained partial music synchronization. In: Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2008). (2008)
13. Droettboom, M., Fujinaga, I.: Symbol-level groundtruthing environment for OMR. In: Proc. ISMIR, Barcelona, Spain. (2004)
14. Byrd, D., Schindele, M.: Prospects for improving OMR with multiple recognizers. In: Proc. ISMIR, Victoria, Canada. (2006) 41–46
15. Ian, K., Byrd, D.: Towards musicdiff: A foundation for improved optical music recognition using multiple recognizers, Vienna, Austria (2007) 123–126
16. Bellini, P., Bruno, I., Nesi, P.: Assessing optical music recognition tools. Computer Music Journal **31** (2007) 68–93
17. McPherson, J.: Coordinating Knowledge to Improve Optical Music Recognition. PhD thesis, The University of Waikato (2006)
18. Blümel, I., Diet, J., Krottmaier, H.: Integrating multimedia repositories into the PROBADO framework. In: Proc. ICDIM. (2008)