AUTOMATIC MAPPING OF SCANNED SHEET MUSIC TO AUDIO RECORDINGS

Christian Fremerey*, Meinard Müller†, Frank Kurth‡, Michael Clausen*

*Computer Science III
University of Bonn
Bonn, Germany

†Max-Planck-Institut (MPI)
for Informatics
Saarbrücken, Germany

‡Research Establishment for
Applied Science (FGAN)
Wachtberg, Germany

ABSTRACT

Significant digitization efforts have resulted in large multimodal music collections comprising visual (scanned sheet music) as well as acoustic material (audio recordings). In this paper, we present a novel procedure for mapping scanned pages of sheet music to a given collection of audio recordings by identifying musically corresponding audio clips. To this end, both the scanned images as well as the audio recordings are first transformed into a common feature representation using optical music recognition (OMR) and methods from digital signal processing, respectively. Based on this common representation, a direct comparison of the two different types of data is facilitated. This allows for a search of scan-based queries in the audio collection. We report on systematic experiments conducted on the corpus of Beethoven's piano sonatas showing that our mapping procedure works with high precision across the two types of music data in the case that there are no severe OMR errors. The proposed mapping procedure is relevant in a real-world application scenario at the Bavarian State Library for automatically identifying and annotating scanned sheet music by means of already available annotated audio material.

1 INTRODUCTION

The last years have seen increasing efforts in building up large digital music collections. These collections typically contain various types of data ranging from audio data such as CD recordings to image data such as scanned sheet music, thus concerning both the auditorial and the visual modalities. In view of multimodal searching, navigation, and browsing applications across the various types of data, one requires powerful tools that support the process of analyzing, correlating, and annotating the available material. In the case of digitized audio recordings, first services have been established to automate the annotation process by identifying each recording and assigning available metadata such as title, artist, or lyrics. Here, the metadata is drawn from specialized annotation databases provided by commercial services such as Gracenote [6] or DE-PARCON [9].

Opposed to acoustic music data, which is increasingly available in digital formats, most sheet music is still produced and sold in printed form. In the last years, digital music libraries have started to systematically digitize their holdings of sheet music resulting in a large number of scanned raster images. To make the raw image data available to content-based retrieval and browsing, methods for automatically extracting and annotating semantically meaningful entities contained in the scanned documents are needed. In this context, *optical music recognition* (OMR) [3] is a key task. Here, the goal is to convert scanned sheet music into a computer readable symbolic music format such as MIDI or MusicXML [13]. Even though significant progress has been made in the last years, current OMR algorithms are substantially error-prone, resulting in systematic errors that require subsequent correction [2]. Similarly, there is still a high demand for reliable solutions for the more general task of automatic sheet music annotation in the digital library community.

In this paper, we present a novel approach for automatically annotating scanned pages of sheet music with metadata. Our approach is based on a new procedure for *mapping* the scanned sheet music pages to an existing collection of annotated audio recordings. The mapping allows for identifying and subsequently annotating the scans based on the metadata and annotations that are already available for the audio recordings. In particular, as it is the case in the specific application scenario at the Bavarian State Library, we assume the existence of an audio collection containing annotated digitized audio recordings for all pieces to be considered in the sheet music digitization process. The conversion of both the audio recordings (by employing filtering methods) and the scanned images (by employing OMR) to a common feature representation allows for a direct comparison of the two different types of data. Using the feature sequence obtained from a few consecutive staves or an entire page of the scanned sheet music as query, we compute the top match within the documents of the audio database. The top match typically lies within a musically corresponding audio recording, which then allows for identifying the scanned page and for transferring all available audio anno-

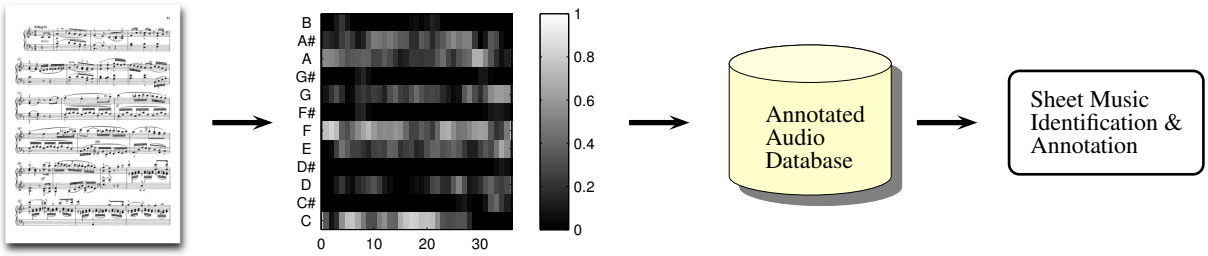

Figure 1. Overview of the mapping procedure for automatic identification and annotation of scanned sheet music using an annotated audio database. The first page of the second movement of Beethoven's piano sonata Op. 2 No. 1 and the resulting scan chromagram are shown.

tations to the scanned sheet music domain. This procedure is described in Sect. 2 and illustrated by Fig. 1. We have tested and analyzed our mapping procedure by means of a real-world application scenario using the corpus of the 32 piano sonatas by Ludwig van Beethoven. In Sect. 3, we discuss the outcome of our experiments showing that the mapping across the two music domains is robust even in the presence of local OMR errors, but suffers in the presence of severe global OMR errors. We also describe a postprocessing procedure that allows for detecting most of the misclassifications and automatically reveals most of the passages within the scanned pages where the severe OMR errors occurred. In Sect. 4, we conclude this paper with prospects on future work and indicate how to improve the identification rate by correcting and compensating for severe OMR errors prior to the mapping stage.

2 MAPPING PROCEDURE

One key strategy of our mapping procedure is to reduce the two different types of music data, the audio recordings as well as the scanned sheet music, to the same type of feature representation, which then allows for a *direct* comparison *across* the two domains. In this context, chroma-based features have turned out to be a powerful mid-level music representation [1, 7, 12]. Here, the *chroma* correspond to the twelve traditional pitch classes of the equal-tempered scale and are commonly indicated by the twelve pitch spelling attributes C, C$^\sharp$, D, . . .,B as used in Western music notation. In the case of audio recordings, normalized chroma-based features indicate the short-time energy distribution among the twelve chroma and closely correlate to the harmonic progression of the underlying piece. Based on signal processing techniques, the transformation of an audio recording into a chroma representation (or chromagram) may be performed either by using short-time Fourier transforms in combination with binning strategies [1] or by employing suitable multirate filter banks [12]. In our implementation, we use a quantized and smoothed version of chroma features, referred to as CENS features, see [12]. The transformation of scanned sheet music into a corresponding chromagram requires several steps, see [11]. First, each scanned page is analyzed using optical music recognition (OMR) [2, 3]. In our system, we use the commercially available SharpEye software [8] to extract musical note parameters (onset times, pitches, durations) along with 2D position parameters as well as bar line information from the scanned image. Assuming a fixed tempo of 100 BPM, the explicit pitch and timing information can be used to derive a chromagram essentially by identifying pitches that belong to the same chroma class. A similar approach has been proposed in [7] for transforming MIDI data into a chroma representation. Fig. 1 shows a resulting scan chromagram obtained for the first page of the second movement of Beethoven's piano sonata Op. 2 No. 1. Note that the tempo assumption (of always choosing 100 BPM) is not a severe restriction since the mapping algorithm to be described next can handle local and global tempo variations anyway.

For the actual scan-audio mapping, we use a matching procedure similar to the one described in [10]. First, in a preprocessing step, all recordings of the given audio database are converted into sequences of CENS vectors. (In our implementation, we use a feature sampling rate of 1 Hz.) While keeping book on document boundaries, all these CENS sequences are concatenated into a single audio feature sequence. Then, each scanned page of sheet music to be identified is also converted into a sequence of CENS features. The resulting scan feature sequence is then compared to the audio feature sequence using subsequence dynamic time warping (DTW). For a detailed account on this variant of DTW we refer to [12]. In our experiments, it turned out that the DTW step sizes $(2, 1), (1, 2), (1, 1)$ (instead of the classical step sizes $(1, 0), (0, 1), (1, 1)$) lead to more robust matching results, and are hence used in the remainder of this paper. As a result of the DTW computation, one obtains a *matching curve*. The ith position of the matching curve contains the costs for matching the scan feature sequence to the most similar subsequence of the audio feature sequence ending at position i. Therefore, the curve's local minima close to zero correspond to audio feature subsequences similar to the scan feature sequence. These subsequences are referred to as *matches*. Because of the book-keeping, doc-

ument numbers and positions of matches within each audio document can be recovered easily. Note that DTW can compensate for possible temporal differences between scan feature sequences and corresponding audio feature subsequences thus also relativizing the above tempo assumption.

3 EXPERIMENTS AND EVALUATION

The basic identification and annotation procedure of a given scanned page of sheet music can be summarized as follows. First, map a given scanned page against the audio database and derive the top match of lowest cost. Then determine the audio recording that contains the top match and transfer all annotations available for the audio recording to the image domain. Note that in the ideal case the top match not only identifies the scanned page but also indicates the time position within the audio recording where the music notated on the page is actually played.

We now show to which extent this approach also works in practice by discussing a real-world application scenario using the musically relevant corpus of Beethoven's piano sonatas. Our test database and some technical details are described in Sect. 3.1. In Sect. 3.2, we discuss a baseline experiment using MIDI versions instead of extracted OMR data. This experiment indicates which identification rates one may expect in the optimal (but unrealistic) case where no OMR extraction errors have occurred. Then, in Sect. 3.3, we describe several experiments performed on the actual OMR data. Here, we also discuss various types of OMR errors that significantly degrade the mapping quality. Finally, in Sect. 3.4, we describe a postprocessing strategy that automatically reveals most of the misclassifications.

3.1 Test Database

Our experiments have been conducted on the basis of the 32 piano sonatas by Ludwig van Beethoven, which play a key role in the evolution of the sonata form and are considered as one of the greatest treasures in the music literature. Because of its outstanding musical significance and the large number of available digitized audio recordings, the automated analysis and organization of the corpus of Beethoven's piano sonatas is highly relevant to musicologists and librarians.

Our audio database consists of a complete recording of the 32 piano sonatas conducted by Daniel Barenboim, comprising 101 audio documents (basically corresponding to the movements) and 11 hours of audio material. Furthermore, we have a scanned version of the corresponding sheet music (Volume 1 & 2, G. Henle Verlag) at our disposal amounting to a total number of 604 digitized pages (3693 two-stave systems). In the following, dealing with piano music, the term *line* is used to denote a two-stave system consisting of a stave for the right and a stave for the left hand. The scanned pages, which are available as 600dpi b/w images in the TIFF format, have been processed by the OMR Engine of SharpEye 2.68 and saved in the MusicXML file format as one file per page. Finally, each of the 604 MusicXML files was transformed into a sequence of CENS vectors (one feature per second) assuming a fixed tempo of 100 BPM. Subsequently, using the extracted OMR information on the notated systems, the CENS sequences were segmented into 3693 subsequences corresponding to the lines.

3.2 Baseline Experiment: MIDI-Audio Mapping

In a baseline experiment, we investigated what identification rates one may expect in the case that there are no severe OMR extraction errors. To this end, we used a complete set of MIDI files for the 32 Beethoven sonatas and randomly generated a large number of MIDI fragments of various lengths, which were used instead of the OMR extraction results. Then, for each of these MIDI fragments we computed a matching curve with respect to the audio database and determined the topmost audio match. Recall that in the identification scenario the objective is to determine the piece of music underlying the respective MIDI fragment by using the audio recordings of the database as an identifier. Therefore, we consider a match as *correct* if it lies within the audio document that corresponds to the same movement as the MIDI document from which the respective query is taken. Otherwise the match is considered as *incorrect*.

In particular, we investigated the dependence of the number of correct audio matches subject to the length L (given in seconds) of the MIDI query. To this end, we randomly generated 1000 MIDI queries for each of the seven parameters $L \in \{10, 20, 30, \ldots, 70\}$. Each of the queries lies within a single MIDI file and therefore has a unique correct assignment to one of the 101 movements. The second column of Table 1 shows the number of correct matches. As an example, consider the case $L = 10$, where 823 of the 1000 matches were correct. Note that the number of correct matches increases significantly with the query length. For example, for $L = 40$ only 3 of the 1000 queries were misclassified. To give a more detailed picture of the matching quality, Table 1 additionally provides various cost and confidence values. The third, fourth, and fifth column show the average cost values, the standard deviations, and the maximal cost values for the correct top matches. For example, in the case $L = 10$, the average cost value (standard deviation/maximal cost value) for the 823 correct matches is 0.059 (0.024/0.223). The latter cost values are with respect to a range from 0 (no costs) to 1 (maximum costs). Increasing L leads to slightly higher cost values stabilizing around the value 0.07 even for long queries.

Similarly, the sixth, seventh, and eighth columns of Table 1 show the corresponding values for the incorrect top matches. For example, in the case $L = 10$, the average cost of the 177 incorrect top matches is 0.084 with a standard de-

Length	#(Cor.)	Cost (correct)			Cost (incorrect)			Gap
(in sec.)	(in %)	av.	std.	max.	av.	std.	max.	av.
10	82.3	0.059	0.024	0.223	0.084	0.034	0.207	0.044
20	96.7	0.068	0.026	0.206	0.102	0.031	0.196	0.070
30	99.2	0.070	0.024	0.189	0.139	0.040	0.214	0.093
40	99.7	0.071	0.024	0.218	0.177	0.027	0.198	0.106
50	99.9	0.072	0.023	0.204	0.117	0.000	0.117	0.118
60	99.9	0.071	0.021	0.193	0.159	0.000	0.159	0.128
70	99.9	0.071	0.022	0.196	0.229	0.000	0.229	0.135

Table 1. Results for the baseline experiment of mapping MIDI fragments of various lengths $L \in \{10, 20, \ldots, 70\}$ (given in seconds) to the audio database. Each line shows the length L, the percentage of correct matches for the 1000 MIDI fragments of the respective length, the average values (av.), the standard deviations (std.), and the maximum values (max.) of the correct matches and incorrect matches, and the average confidence gap.

Length	#(Cor.)	Cost (correct)			Cost (incorrect)			Gap
(in sec.)	(in %)	av.	std.	max.	av.	std.	max.	av.
10	48.4	0.080	0.033	0.198	0.104	0.040	0.247	0.034
20	67.9	0.103	0.039	0.261	0.147	0.051	0.285	0.050
30	78.4	0.114	0.044	0.292	0.173	0.049	0.317	0.062
40	84.9	0.120	0.043	0.356	0.192	0.051	0.340	0.072
50	87.1	0.132	0.043	0.305	0.208	0.051	0.367	0.080
60	87.0	0.143	0.050	0.304	0.232	0.044	0.356	0.080
70	87.1	0.153	0.052	0.316	0.247	0.049	0.373	0.078

Table 2. Experimental results mapping OMR fragments of various lengths (given in seconds) to the audio database. For each length parameter $L \in \{10, 20, \ldots, 70\}$ we randomly generated 1000 OMR chroma subsequences, each corresponding to a subpart of exactly one of the scanned pages. The table has the same interpretation as Table 1.

viation of 0.034. Note that in the case of incorrect matches, when increasing the query length, the average cost increases at a much higher rate than in the case of correct matches.

We also investigated how well the correct matches were separated by successive matches that do not lie in the respective correct audio document. To this end, we computed for each query the minimal cost value of a restricted matching curve, where the correct audio document had been removed. Then, for all correctly identified queries, we computed the difference of this minimal value and the cost of the correct top match. This difference value, which we refer to as *confidence gap*, indicates the identification reliability based on the top match. The average confidence value is shown in the last column of Table 1. For example, in the case $L = 10$ the average confidence gap amounts to the value 0.044. Increasing L leads to a significant increase of the confidence gap up to the value of 0.135 for $L = 70$. In conclusion, one may say that one obtains very good identification rates (with an error rate of less than 1%) for MIDI fragments of at least 30 seconds of duration.

3.3 OMR-Audio Mapping

Next, we describe a similar experiment, now using the (potentially flawed) OMR extraction results instead of the "clean" MIDI data. For each of the 604 scanned pages, we computed a CENS feature sequence as explained in Sect. 2. Then, from these sequences, we randomly generated 1000 subsequences of length L for each of the length parameters $L \in \{10, 20, \ldots, 70\}$. Table 2 summarizes the OMR-audio mapping results. Obviously, the identification rate drops significantly compared to the pure MIDI case. For example, in the case $L = 10$ only 484 out of the 1000 OMR query fragments appear as top match in the correct audio document (opposed to the 823 correct matches in the MIDI case). The identification rate increases to roughly 87% for OMR feature sequences that correspond to a duration of 50 seconds and above. A comparison with Table 1 shows that, in the OMR case, the average costs of the correct matches are much higher than the ones in the MIDI case. Furthermore, the confidence gap is much smaller.

All these numbers indicate that the OMR-audio mapping procedure significantly suffers from the artifacts that are mainly caused by OMR extraction errors. In particular, a manual investigation of samples of the OMR extraction results revealed that there are two prominent types of OMR errors that significantly degrade the quality of the CENS feature sequences. First, for roughly 7% of the lines (two-stave systems) the key signature was extracted incorrectly. In particular, one or even more accidentals notated at the beginning of each stave were missing. Such an error generally distorts the CENS subsequence for an entire line, since a missing accidental causes all notes of a specific pitch class to be shifted upwards or downwards by one semitone, which may significantly corrupt the chroma distribution. Second, in almost 5% of the measures there were some note or beam extraction errors that resulted in inconsistencies with respect to the notated time signature. In such cases, the conversion tool of our OMR software, which transforms the OMR extraction parameters into a MusicXML file, simply discards all voices within those measures that reveal such inconsistencies. This also results in a significant corruption of the chroma distribution. Obviously, the automated detection and correction of such OMR extraction errors would overcome these problems resulting in significantly improved identification rates. These issues are left for future work and are further discussed in Sect. 4.

We continue the analysis of our OMR-audio mapping procedure based on the raw OMR material. Instead of using randomly chosen OMR fragments of a specific duration, we now investigate the mapping quality based on musical units such as pages or lines. Using entire pages in the OMR-audio mapping leads to an identification rate of roughly 82.5%. The average length of the corresponding CENS sequences

Lines	Length (in sec.)	$k=1$ (in %)	$k=2$ (in %)	$k=5$ (in %)	$k=10$ (in %)	$k=20$ (in %)	$k=50$ (in %)
1	9.133	44.57	52.97	65.77	76.20	84.59	92.26
3	27.099	71.30	76.66	83.62	88.06	92.45	96.13
5	45.053	77.04	81.23	86.41	90.12	93.37	96.86
7	62.995	77.74	81.83	86.84	90.85	93.83	96.89

Table 3. Identification rates depending on the number of lines used in the OMR-audio mapping. The columns indicate the recall percentage (out of 3693 mappings, respectively) of the correct audio document within the top k matches.

amounts to 55 seconds yielding robust mappings if there are no severe OMR errors. Another problem that often leads to misclassifications is that a single scanned page may refer to more than one pieces of music. In particular for our Beethoven corpus, a single page may contain both the end and the beginning of two consecutive movements. To overcome this problem, one may use single lines in the mapping process instead of entire pages. This also yields the advantage of having several identifiers per page. On the downside, the average length of the CENS sequences corresponding to the lines lies below a duration of 10 seconds yielding an identification rate of only 44.57%, see Table 3. To improve the identification rate of the line-based mapping strategy, we query each line in the context of ℓ preceding and ℓ subsequent lines. In other words, instead of using a single line we use a block of $2\ell+1$ subsequent lines with the reference line positioned in the middle. Here, we assume that all pages belonging to one movement are in the correct order, hence allowing us to consider blocks of lines ranging across two consecutive pages. To systematically investigate the identification rate depending on the number of lines used in the OMR-audio mapping, for each of the 3693 lines of our scanned Beethoven material, we generated CENS query sequences corresponding to 1, 3, 5, and 7 lines. Table 3 shows both the resulting identification rates based on the top match ($k=1$) and the recall values for the correct audio document for the top k matches with $k \in \{1,2,5,10,20,50\}$. For example, using three lines, the top match ($k=1$) was correct in 71.30% of the 3693 OMR-audio mappings. Considering the top 5 matches ($k=5$), at least one of these matches was correct in 83.62% of the mappings.

3.4 Postprocessing

We now show how the additional information of considering the k top matches (instead of considering only the top match) can be used to detect most of the incorrect identifications. The only assumption we use is that the scanned pages that correspond to a specific movement are given as a sequence of consecutive pages, i. e., pages of different movements are not interleaved. We explain our postprocessing procedure by means of our Beethoven scenario using $Q = 3693$ OMR queries each consisting of 7 subsequent lines and considering the $k=5$ top matches. Recall that the objective is to map each of the queries to one of the $P=101$ audio documents (representing the pieces or movements). We construct a $P \times Q$ *mapping matrix* M, where the rows correspond to the pieces and the columns to the queries. Then an entry $M(p,q)$, $1 \leq p \leq P$, $1 \leq q \leq Q$, is non-zero if and only if the p^{th} audio document appears among the top k matches for the q^{th} query. In this case $M(p,q)$ is set to $1-c$, where $c \in [0,1]$ denotes the cost of the corresponding match. In case there are several matches for the entry (p,q) among the top k matches, we define c to be the minimal cost value over these matches. Note that $M(p,q)$ expresses a kind of confidence that the q^{th} query belongs the p^{th} piece. Furthermore, M indicates the kind of confusion that occurred in the identification procedure. Fig. 2 shows the mapping matrix for the Beethoven scenario.

For our Beethoven corpus, both the audio recordings and the scanned pages are sorted with respect to increasing opus and movement numbers. Therefore, a correct mapping of all queries corresponds to a diagonal staircase-like structure in M. In the following, we do not assume that the scanned pages are given in the same order (on the piece and movement level) as the audio recordings, since this assumption is often violated in real-world digitization applications. For example, many music books contain a more or less unsorted mixture of various pieces and movements. Therefore, we only make the assumption that the pages that correspond to a specific audio document (referring to a specific movement) are given in the correct order. Then, in case of a correct identification of the OMR queries, the matrix M reveals a structure of horizontal line segments, where each such segment corresponds to an audio document.

In the following, a tuple (p,q) is referred to as *positive* if the entry $M(p,q)$ is non-zero. Furthermore, a positive tuple (p,q) is referred to as *true positive* if the q^{th} query semantically corresponds to the p^{th} audio document, otherwise (p,q) is called *false positive*. Now, the idea is that positive tuples included in long horizontal line segments within M are likely to be true, whereas isolated positive tuples are likely to be false. Intuitively, our procedure classifies the positive tuples by looking for groups of tuples included in long horizontal line segments (these tuples are classified as true) and discards isolated positives tuples (these tuples are classified as false). Due to space limitations, we do not give technical details and refer to Fig. 2 for an illustration.

We have applied this postprocessing procedure to the Beethoven scenario using $Q = 3693$ queries each consisting of 7 subsequent lines and considering the top match only. As a result, 78.42% of the queries were mapped correctly and 17.17% of the queries were not mapped (by discarding false positives). The remaining 4.41% are incorrect mappings. Note that the result of this type of postprocessing is the detection rather than the correction of incorrect identifi-

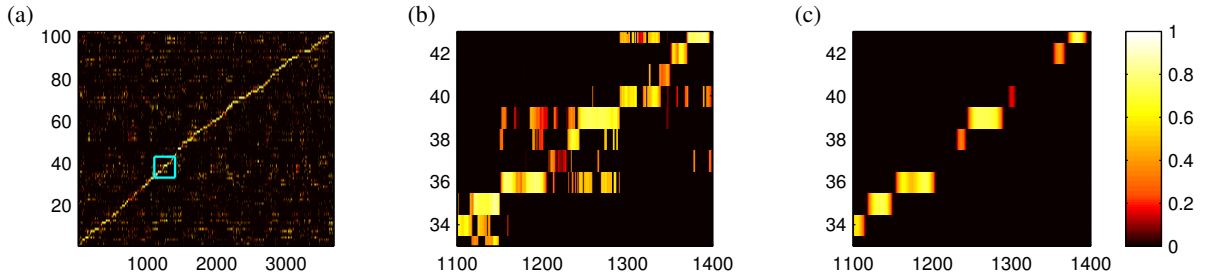

Figure 2. **(a)** Mapping matrix M for the Beethoven scenario. The rows correspond to the audio documents ($P = 101$) and the columns to the OMR queries ($Q = 3693$). **(b)** Enlargement of the marked region of M. **(c)** The same region after applying the postprocessing procedure.

cations. Having identified incorrect mappings allows to both further improve the identification process and to automatically reveal passages within the sheet music where severe OMR errors have occurred.

Rather than identifying incorrect mappings, one may also increase the number of correct identifications. For this, certain tuples are specified as true positives by "filling" small gaps within horizontal line segments. Thus, OMR queries are assigned to a specific audio document if neighboring OMR queries are consistently assigned to the same audio document. Using $k = 3$ in our example increases the number of correct identifications to 86.70% (instead of 77.74% without postprocessing). Note that there is a natural trade-off between eliminating the incorrect identifications and boosting the correct identifications.

4 CONCLUSIONS

In this paper, we have introduced the problem of mapping sheet music to audio recordings. Based on an automated mapping procedure, we have presented a novel approach for automatically identifying scanned pages of sheet music by means of a given audio collection. Such a procedure, which constitutes an important component in the digitization and annotation process of multimodal music material, is needed for building up the Probado music repository [4] currently set up at Bavarian State Library in Munich, Germany. This music repository, which contains digitized sheet music and audio data for a large collection of classical and romantic piano sonatas (Haydn, Mozart, Beethoven, Schubert, Schumann, Chopin, Liszt, Brahms) as well as German 19th centuries piano songs, is continuously expanded requiring automated procedures for music processing and annotation.

As our experiments show, the proposed procedure for mapping scanned sheet music and audio material works well in the case that there are no severe OMR extraction errors. Our postprocessing procedure allows for automatically revealing most of the critical passages containing these OMR errors. In the future, we will use various heuristics to correct typical OMR errors prior to the mapping step. For example, in the case of piano music, different key signatures for the left and right hand staves can be assumed to be invalid and easily corrected by considering neighboring stave lines. Furthermore, similar to the strategy suggested in [2], one can simultaneously employ various OMR extraction results obtained from different OMR software packages to stabilize the mapping result. Based on these strategies, we expect to achieve a significant improvement of the identification rates reaching the ones reported in our MIDI baseline experiment.

5 REFERENCES

[1] Bartsch, M.A., Wakefield, G.H.: Audio thumbnailing of popular music using chroma-based representations. IEEE Trans. on Multimedia **7**(1), pp. 96–104 (2005).

[2] Byrd, D., Schindele, M.: Prospects for improving OMR with multiple recognizers. Proc. ISMIR, Victoria, CA (2006).

[3] Choudhury, G., DiLauro, T., Droettboom, M., Fujinaga, I., Harrington, B., MacMillan, K.: Optical music recognition system within a large-scale digitization project. Proc. ISMIR, Plymouth, MA, US. (2000).

[4] Diet, J., Kurth, F.: The Probado Music Repository at the Bavarian State Library. Proc. ISMIR, Vienna, AT (2007).

[5] 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**(8), pp. 53–48 (2006).

[6] Gracenote: `http://www.gracenote.com` (2008)

[7] Hu, N., Dannenberg, R., Tzanetakis, G.: Polyphonic audio matching and alignment for music retrieval. Proc. IEEE WASPAA, New Paltz, US (2003).

[8] Jones, G.: SharpEye Music Reader, `http://www.visiv.co.uk` (2008)

[9] Krajewski, E.: DE-PARCON software technology, `http://www.deparcon.de` (2008)

[10] Kurth, F., Müller, M.: Efficient Index-based Audio Matching. IEEE Trans. on Audio, Speech, and Language Processing **16**(2), pp. 382–395 (2008).

[11] Kurth, F., Müller, M., Fremerey, C., Chang, Y., Clausen, M.: Automated Synchronization of Scanned Sheet Music with Audio Recordings. Proc. ISMIR, Vienna, AT (2007).

[12] Müller, M.: Information Retrieval for Music and Motion. Springer (2007).

[13] Recordare LLC: Music XML, `http://www.recordare.com/xml.html` (2008).