Music Search Engine from Noisy OMR Data

Sanu Pulimootil Achankunju
Bavarian State Library
Munich, Germany
sanu.pulimootil-achankunju@bsb-muenchen.de

***Abstract*—The music department of the Bavarian State Library (BSB) is one of the internationally leading music libraries. A huge collection of our music scores are only accessible via meta data based search. So we have started a new Music Information Retrieval (MIR) project to enable content wise searching in printed music scores. To this end, we extracted the symbolic note information from the entire works of four famous composers using Optical Music Recognition (OMR) technology, which transforms sheet music or printed scores into a machine readable format. The OMR data are quite noisy containing numerous extraction errors. We have created a music search engine to enable melody search on this noisy data, that can still achieve a very good retrieval quality. We also report on our experiments to test the quality of this search engine with musical themes from an external source.**

***Index Terms*—Optical Music Recognition, Music Information Retrieval, melody search, N-gram**

I. Introduction

Full-text search is now a common concept in many digital libraries due to the advancements in Optical Character Recognition technology. However, that is not the case when dealing with musical documents because OMR technology is far from being perfect [1] [2]. Meta-data based search is the most prominent way of accessing these documents. In order to enhance the search procedure, BSB has started a new project to enable content wise searching in the printed music scores. In the first phase of this project, many OMR software applications were subjectively tested on a small subset of digitized music sheets to find out the best software that works well with our images. In the second phase, sheet music images were converted into symbolic music using the selected software. In the third phase, a web application named "musiconn scoresearch" was developed to enable melody search on this data. Later phases of this project will support features like theme recognition/search, polyphony, rhythm search, etc. In this paper, we deal with a MIR scenario where noisy OMR data form the base of a music search engine.

II. Processing Pipeline

After comparing SharpEye, Audiveris, SmartScore X^2 and Capella Scan, we have decided to use SmartScore X^2 Professional v 10.5.8 for our project as the OMR software. This software was installed on a dedicated server and the sheet music images are sent to it automatically for processing. Complete editions of the works of Ludwig van Beethoven, Georg Friedrich Händel, Franz Schubert and Franz Liszt were converted into symbolic music using this software. As a result, about 40,000 MusicXML files were created from about 44,000 images including non-music sheets like content, introduction and blank pages. Due to the large quantity, manual correction of the noisy OMR data is a difficult task. Each parts of the partwise MusicXML file are processed separately. Polyphony is removed to reduce the complexity of index and search process. Only pitch sequences are considered for the melody search engine. Monophonic pitch sequence is created by considering only the highest value of pitch at a given instance of time, i.e., we take the upper pitch contour of the given polyphonic part. If the part has more than one staff, we extract the monophonic pitch sequence of each individual staff and the same of the given part considering all the staves as a single piece.

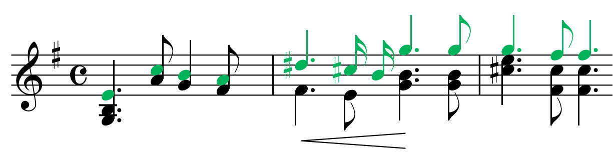

Fig. 1. Sample OMR output from Händel's "Der Messias: Oratorium"

Let us consider the sample OMR output given in Fig. 1. Each element of the pitch sequence consists of the note name and its octave number. Monophonic pitch sequence generated from the OMR data is given as `E4 C5 B4 A4` D$^\sharp$5 C$^\sharp$5 `B4 G5 G5 G5` F$^\sharp$5 F$^\sharp$5 and is highlighted in the figure. In addition to the pitch sequences, we convert each of these also into sequences of intervals. Interval sequence is nothing but the difference of adjacent pitches in the sequence. An advantage of this sequence is its invariance to transposition [3]. OMR procedure can misinterpret the global information such as clef [4] or might detect wrong notes. The interval sequence helps us to get the melodic contour. Interval sequence for the above given example is `8, -1, -2, 6, -2, -2, 8, 0, 0, -1, 0`.

After extracting pitch and interval sequences from the parts of a MusicXML file, they are sent along with their metadata to our search engine for indexing. From these sequences, N-grams of different sizes are indexed in the search engine. N-grams can be considered as a sequential list of n words. The task of our search engine is to find out the documents that have the same N-grams as the query.

This work has been supported by the German Research Foundation (Deutsche Forschungsgemeinschaft) in the grant "Fachinformationsdienst Musikwissenschaft - 1. Fortsetzungsantrag", CE 134/27-2, Jan 17 - Dec 19.

III. Search engine Interface

We have also created a user friendly front end for the search engine. A screenshot of this web interface is shown in Fig. 2. Query input to the search interface can be given using a virtual piano. A minimum of three notes should be given as a query and it can be any notes from `C2` to `C7`. As a visual feedback,

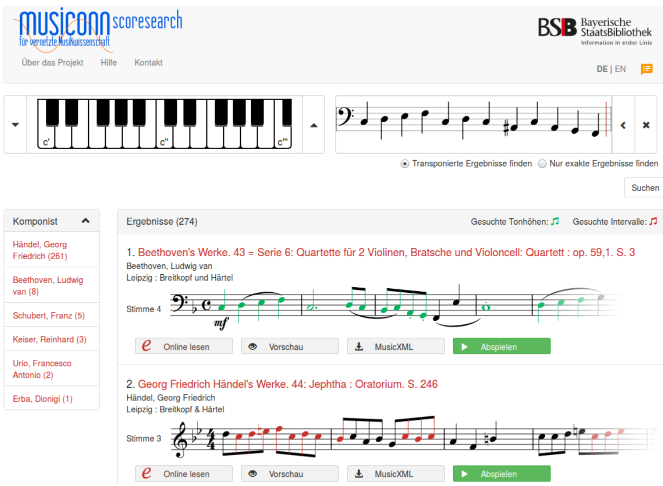

Fig. 2. Screenshot of the Music Search Engine

the notes given through the piano will appear on the editable musical staff. Notes can also be given through the keyboard shortcut keys and it will be inserted into the current cursor position that can be changed by clicking the staff. An audio feedback is also given as each note is pressed. Higher or lower octaves can be selected using buttons to the right and left of the piano. The default option will search the absolute pitches and intervals together. There is also an option to search only the pitches. A search on an average takes only about 4-5 seconds.

The search results will appear with its metadata. Search result displays some measures where the notes are found. Pitches or intervals detected in these measures are highlighted in different colours as shown in Fig. 2. There are options to open the exact page of the original document, to see the whole page rendered from MusicXML data with all the search results highlighted, to download the particular MusicXML file and to listen to the search results. Faceted navigation is also enabled to refine the search results.

IV. Retrieval Experiment and Result

In order to evaluate the retrieval quality, we use MIDI files from the "Electronic Dictionary of Musical Themes"[1]. It contains roughly about 10000 monophonic musical themes of instrumental western classical music. For our evaluation purposes, we considered only about 1200 of them from the works of same composers mentioned in II. Using the metadata, ground truth information were made for 200 random files and pitch data were extracted from these files. Random queries were made from these data and were sent to the search engine. Each query contains about 6 to 12 notes, not necessarily from the beginning of a theme. In a normal search engine scenario, a user is only interested in the first few matches. So in our experiment, for each query we consider the top K matches or the first K results of the search engine for some number $K \in \mathbb{N}$. A search is considered successful if the top K matches contain the relevant document. In our retrieval process, we

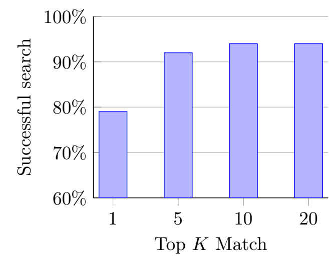

Fig. 3. Top K match for retrieval experiment

consider the percentage of successful cases. Fig. 3 shows the retrieval results for $K \in \{1, 5, 10, 20\}$. Considering the top match ($K = 1$), the search engine successfully retrieved the documents for 79% of the queries. There has been a substantial increase in the top five match to 92%. This increase can be explained by the fact that queries containing low number of notes can retrieve similar pieces and hence the ground truth document will be some where on the top positions of the retrieved list. Only when we become more specific about a theme, it will appear in the top match. Considering $K = 10$, one obtains 94% successful match which indicates that the information retrieval quality is quite good. There is no further increase in the quality of the retrieved list as $K = 20$. If the OMR errors are comparatively smaller, then the document will already be in the top matches of the retrieved list. An inspection on some of the failed queries showed that the original document contains fatal OMR errors. In such cases, obviously the retrieval will not be successful.

V. Conclusion

In this paper we have presented a new melody search engine that searches the noisy OMR data. We evaluated the retrieval quality of the search engine using external sources. The result shows that the retrieval quality is good even though the score inputs were corrupted by the OMR process.

[1] www.multimedialibrary.com/barlow/all_barlow.asp (accessed 08.07.2014)

References

[1] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[2] D. Bainbridge and T. Bell, "The challenge of optical music recognition," *Computers and the Humanities*, vol. 35, no. 2, pp. 95–121, 2001.

[3] S. Doraisamy and R. Stefan, "A Polyphonic Music Retrieval System Using N-Grams," in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR-2004)*, 2004.

[4] S. Balke, S. P. Achankunju, and M. Müller, "Matching Musical Themes based on noisy OCR and OMR input," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2015, pp. 703–707.