Audio–Sheet Music Alignment using Soft Bootleg Score Synthesis

Teerapat Jenrungrot* TJ Tsai
Department of Engineering
Harvey Mudd College
mjenrungrot@hmc.edu ttsai@hmc.edu

***Abstract*— Audio–sheet music alignment is the task of finding correspondences between time instants in an audio recording and corresponding pixel locations in sheet music images. We propose a simple, straightforward approach to this problem that requires little or no training data. Our method consists of two main steps. The first step is to convert audio into a crude approximation of a sheet music score, where sudden energy increases in logarithmically-spaced frequency bins are mapped to floating notehead blobs appropriately placed on a staff line system. We generate this crude 'bootleg' score in a soft manner, where the darkness of the notehead blob is proportional to the amount of energy increase. The second step is to align the bootleg score with the sheet music images using a simple variant of dynamic time warping. We present empirical results on the multimodal sheet music dataset, including a comparison to a state-of-the-art system based on deep reinforcement learning. Our method achieves 80% accuracy at an error tolerance of 400 pixels, which is approximately half a line of music.**

I. INTRODUCTION

This paper addresses the problem of audio–sheet music alignment. The goal of this task is to determine the correspondence between each time instant in an audio recording and the corresponding location in the sheet music image. This task is similar to score following but has two significant differences: (1) score following applications typically assume that a symbolic representation of the score is available (e.g. [9–11]), whereas we work with raw sheet music images, and (2) score following is an online task, whereas we consider an offline setting. The International Music Score Library Project[1] has made sheet music available in abundance, but utilizing this resource requires working with raw sheet music images.

Audio–audio alignment is a related problem, and is very well studied. Here, the goal is to find the temporal alignment between two different audio recordings of the same musical piece. The main technique used to solve this problem is dynamic time warping (DTW) [4]. Many works have proposed extensions to DTW that work in an online fashion [5], estimate the alignment at multiple granularities [6], handle jumps and repeats [7], and take advantage of multiple recordings [8].

The audio–sheet music alignment problem is less well studied. There are two main approaches to this problem. The first approach is to use optimal music recognition (OMR) to convert the sheet music into a symbolic representation such as MIDI, and then to align the MIDI and audio using a chroma feature representation [12, 13]. The second more recent approach is to use a multimodal neural network to embed the sheet music and audio in a shared feature space that reflects semantic similarity [2, 3]. The alignment can then be performed using traditional DTW. Another recent work [1] formulates the audio–sheet music alignment problem as a deep reinforcement learning game where the agent has to navigate through the score by adopting its reading speed in response to the currently playing performance. This approach achieves good performance on a synthetic dataset [14] but requires a significant amount of training in a simulated environment. The need for large amounts of annotated training data makes it challenging to train an agent that generalizes well to unseen pieces and audio conditions, especially when the sheet music and audio are not synthetically rendered. For instance, an agent trained on synthetic sheet music may not handle issues like poor scan quality or staff lines that are skewed or warped.

This paper proposes a simple approach for audio–sheet music alignment that requires little or no training data. Previous works on cross-modal alignment have focused either on using OMR to bridge the modality gap or to use large datasets to train a complex multimodal neural network. This work examines the tacit assumption that the raw pixel image space is to be avoided simply because it is very high-dimensional. We show that it is possible to project an audio signal directly into raw pixel space in a way that encodes musical knowledge and has a clear interpretation. Once the audio is projected into the image space, we can estimate the alignment in the image domain using DTW with a simple inner product similarity metric.

The paper is organized as follows: Section II describes the proposed system. Section III explains the experimental setup. Section IV presents empirical results. Section V concludes the work and suggests future directions.

II. SYSTEM DESIGN

There are two inputs to our system: an audio recording and the corresponding sheet music. Similar to other recent works [1–3], we assume that the sheet music is given as a sequence of image strips, where each image strip contains a single line of music. In this work, we focus on piano music, so each strip contains a single grand staff containing an upper staff (for the right hand) and a lower staff (for the left hand). The topmost image in figure 3 shows an example of an

* Final project for E190AP: Music Signal Processing, fall 2018.
[1] https://imslp.org/

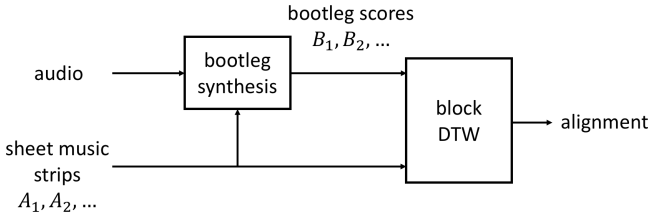

Fig. 1. System Overview. We first convert the audio into a crude approximation of the score, and then align the bootleg scores and sheet music strips with block DTW. Each bootleg score B_i contains the entire piece projected onto the staff line coordinate system of the i^{th} image strip A_i.

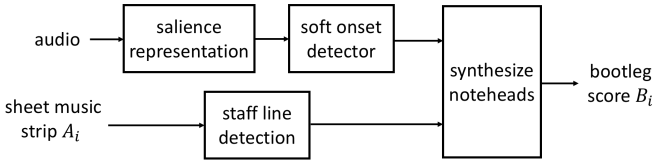

Fig. 2. Overview of bootleg synthesis. From the audio signal, we estimate note onsets for each note. From the sheet music strip, we determine the locations of the staff lines. We then translate note onsets into notehead blobs appropriately placed on the staff line coordinate system. We synthesize the notehead blobs in a soft manner, where the darkness of the notehead blob is proportional to the amount of evidence of an onset.

image strip. Note that the image strips may have different dimensions (particularly the height), and the staff lines may appear at different locations on different image strips.

Figure 1 shows the two keys steps in our proposed approach. The first step is to convert the audio data into a crude approximation of the score, a process which we call bootleg synthesis. The output of bootleg synthesis is a single long image strip containing the entire 'bootleg' score for the entire piece. We use bootleg synthesis to generate N different complete bootleg scores $B_1, B_2, ..., B_N$, where each B_i is the bootleg score projected onto the staff line coordinate system of the i^{th} image strip A_i. In other words, the bootleg scores B_i are all identical except for differences in the staff line locations. The second step is to align the bootleg scores and the original sheet music strips using a variant of DTW which we call block DTW. These two key steps are described in more detail in the following two subsections.

A. Bootleg Synthesis

Figure 2 shows how we generate a bootleg score. There are four substeps, which we describe below.

The first substep is to compute a salience representation for the audio signal. We calculate a log frequency spectrogram with 12 bins per octave spanning from C1 to C8, and then sum the energy of the first eight harmonics for each note. The salience representation is thus a rough indicator of the amount of evidence that a note is being played.

The second substep is to perform soft onset detection. We compute a novelty function $\Delta(n, f)$ for pitch f at frame n according to

$$\Delta(n, f) = max(S(n, f) - S(n-1, f), 0)$$

where $S(n, f)$ is the salience representation. This novelty function captures energy increases in $S(n, f)$. The output of the (soft) onset detector is then given by

$$I(n, f) = \left(\frac{\Delta(n, f)}{M} \right)^2$$

where $M = \max_{n,f} \Delta(n, f)$ is a normalization factor and the squared exponent penalizes lower values. The output of the onset detector is thus a value between 0 and 1 indicating the likelihood of an onset. Note that recent works have used more complex approaches to onset detection based on non-negative matrix factorization and convolutional neural networks [15, 16]. These methods could just as well be used for the onset detection block, but would require a significant amount of labeled training data.

The third substep is staff line detection. Here, the goal is to estimate the vertical pixel position for each of the 10 staff lines in the image strip. We do this by computing the row sum of image pixels, convolving the result with comb filters of various sizes (each containing 5 regularly spaced impulses), and identifying the comb filter that yields the strongest response at two non-overlapping staff locations. This gives us the staff line coordinate system for the upper and lower staves in the image strip.

The fourth substep is synthesizing noteheads. We map note onsets to floating rectangular notehead blobs appropriately placed in the given staff line coordinate system. This mapping is done in a soft manner, where higher onset detection values correspond to darker notehead blobs. Figure 3 shows an example image strip (top) and the corresponding portion of the soft bootleg score generated from an audio recording (middle). The bottommost image of figure 3 shows the corresponding portion of a hard bootleg score assuming perfect (hard) note onset detection, and is included to make the concept of a bootleg score more clear. Note that there is ambiguity when converting from a piano note to its staff line position. For example, G-sharp and A-flat correspond to the same piano note but appear at different staff line positions. To resolve this issue, we simply place noteheads at both possible locations. Furthermore, piano notes in the middle register could appear in the right hand or left hand staves. To handle this ambiguity, we can again place noteheads at both possible locations.

One more point is worth mentioning. When converting the audio into a bootleg score, we need to specify how much time corresponds to a single pixel column. To avoid extreme warping in the DTW, we adaptively select this parameter to ensure that the bootleg representation is approximately the same length as the image strips concatenated end-to-end.

At the end of this process (Figure 2), we have a single bootleg score representation of the entire performance (B_i). Because each sheet image strip A_i may have a different size and a different staff line coordinate system, we generate one bootleg score B_i for each image strip A_i. In other words, B_i is the bootleg score representation projected onto the staff

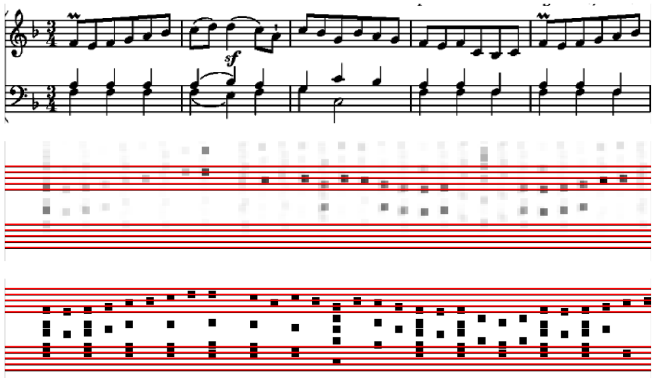

Fig. 3. An image strip (top) and the corresponding portion of a soft bootleg score generated from audio (middle). The bottommost image is a hard bootleg score generated from a note onset oracle, and is included merely for explanatory purposes.

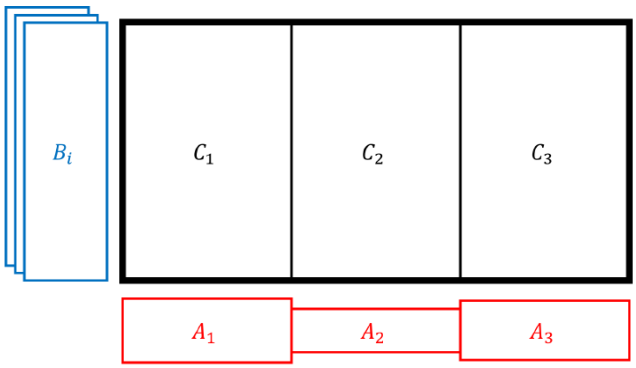

Fig. 4. Overview of block DTW. Each sheet music strip A_i is compared to its corresponding bootleg score B_i to generate a cost matrix block C_i. The cost matrix blocks are concatenated to form a global cost matrix, and DTW is used to estimate the global alignment. Note that each B_i is the bootleg score of the entire piece projected onto the staff line coordinate system of strip A_i.

line coordinate system of image strip A_i. Note that all of the bootleg score representations will have the same length (i.e. number of pixel columns), but each B_i will have a unique height that matches the height of image strip A_i.

B. Block DTW

We align the sheet music strips and bootleg scores using a simple variant of DTW. Figure 4 shows a graphical depiction of this process. In this example, the sheet music contains three image strips A_1, A_2, and A_3. We first calculate the cost matrix C_i between each image strip A_i and its corresponding bootleg score B_i. Because A_i and B_i share the same staff line coordinate system, we can compute their similarity with a simple negative inner product. The $(k,l)^{th}$ element of C_i thus indicates the amount of overlapping black pixels in the k^{th} pixel column of B_i and the l^{th} pixel column of A_i. We assemble the constituent cost matrices C_i into a single global cost matrix (represented as a bold black rectangle in Figure 4), and then apply DTW with step transitions $\{(1,1),(1,2),(2,1)\}$ and corresponding weights $\{2,3,3\}$. The lowest cost path through the global cost matrix is the estimated alignment between the audio recording and the sheet music.

III. EXPERIMENTAL SETUP

We will describe the experimental setup in three parts: the baseline systems, the dataset, and the evaluation metric.

We compare our system to two other baseline systems. The first baseline is global linear interpolation. Here, we simply assume that there is a global linear correspondence between the audio recording and the panorama strip containing the sheet music strips concatenated end-to-end. The second baseline system is the recently proposed approach based on deep reinforcement learning [1]. We also evaluate a variant of our proposed system where we use a log frequency spectrogram in place of the salience representation (top left block of Figure 2). The log frequency spectrogram can be interpreted as a degenerate case of the salience representation when we only consider a single harmonic (i.e. the fundamental frequency).

We evaluate performance on the multimodal sheet music dataset (MSMD) [14]. MSMD contains 479 classical piano pieces from 53 composers. For each piece, the dataset provides the sheet music images, pixel locations of all noteheads and staff lines, three different MIDI performances at slightly different tempos, and the ground truth correspondence between pixel locations of noteheads and timestamps in the MIDI files. The dataset also provides four different piano soundfonts for synthesizing audio from MIDI. Note that the sheet music and MIDI performance are both synthetically rendered from LilyPond. Because the reinforcement learning baseline system [1] trains its network on a subset of the 479 pieces, we decided to evaluate all systems on the same test split containing 100 pieces. This test split contains 131 sheet music pages and 29 321 notehead-timestamp pairs. We evaluated performance of all systems using only one soundfont.

We use two different evaluation metrics. The first metric is error rate at a fixed error tolerance. The error rate indicates the percentage of notehead-timestamp pairs that are within a given allowable error tolerance. By considering a range of error tolerance values, we can plot the tradeoff between error rate and error tolerance. The second metric is the mean ($\overline{|e_i|}$) and standard deviation ($std(|e_i|)$) of absolute errors in predicted note onset locations.

IV. RESULTS

Figure 5 and Table I compare the performance of all four systems. There are three things to notice about these results. First, the global linear baseline performs very poorly. This simply establishes that the problem is not trivial. Second, the deep reinforcement learning system ('Dorfer') performs best by far and achieves very low error rates. For reference, a single sheet music strip is approximately 800 pixels in length, so this system achieves reliable alignment to within a fourth of a music strip. This shows what can (currently) be achieved in scenarios where a large amount of labeled training data

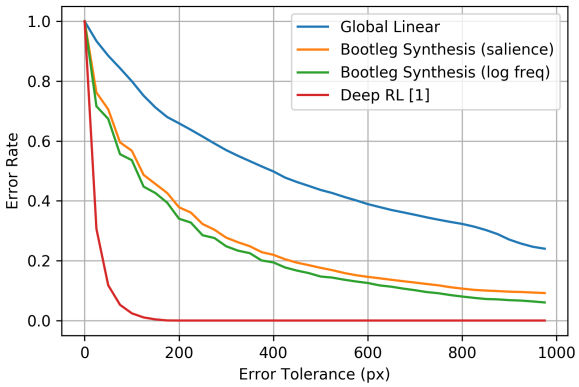

Fig. 5. Error tolerance curves on the multimodal sheet music dataset.

System	$\overline{\lvert e_i \rvert}$	$std(\lvert e_i \rvert)$
Global Linear	623 px	632 px
Bootleg Synth (salience)	330 px	540 px
Bootleg Synth (log freq)	249 px	368 px
Deep RL [1]	23 px	26 px

TABLE I

MEAN AND STANDARD DEVIATION OF ABSOLUTE ERROR

is available. Third, the bootleg systems perform much worse than deep reinforcement learning but much better than global linear interpolation. This shows what can be achieved in scenarios where training data is not available, perhaps to bootstrap a system that requires training labels.

V. CONCLUSIONS

We propose a method for audio–sheet music alignment that does not require any training data. This method uses musical domain knowledge to convert an audio recording into a crude approximation of a sheet music score, a process which we call bootleg score synthesis. The bootleg score contains floating notehead blobs that are appropriately placed on a staff line coordinate system, where the darkness of the notehead blob is proportional to the amount of evidence of a note onset. We then align the bootleg representation and the sheet music in the image domain using a variant of dynamic time warping. We evaluate the proposed system on the multimodal sheet music dataset. Our results indicate that the proposed approach works moderately well on piano music. However, the state-of-the-art system based on deep reinforcement learning outperforms our system by a significant margin. As a trade-off, the deep reinforcement learning approach requires a significant amount of labeled training data, which is only available on synthetically rendered data. We note that our proposed system does not involve any training and can be adapted very easily to handle other kinds of music involving other instruments (e.g. string quartet).

There are several avenues of future work we plan to explore. Because this project was focused on applying the concepts covered in the Music Signal Processing course, our approach was necessarily limited to non-deep learning approaches. However, the bootleg synthesis approach could be easily integrated with deep learning methods to improve performance. In particular, the topmost branch of Figure 2 could be replaced with a state-of-the-art method for automatic piano transcription such as [17]. We also plan to do a more careful analysis of system errors to identify areas of improvement.

REFERENCES

[1] Matthias Dorfer, Florian Henkel, and Gerhard Widmer. "Learning to Listen, Read, and Follow: Score Following as a Reinforcement Learning Game", *in Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, 2018, pp. 784-791.

[2] Matthias Dorfer, Andreas Arzt, and Gerhard Widmer. "Towards score following in sheet music images", *in Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 789-795.

[3] Matthias Dorfer, Andreas Arzt, and Gerhard Widmer. "Learning audio-sheet music correspondences for score identification and offline alignment", *in Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*, 2017, pp. 115-122.

[4] Roger B. Danenberg and Ning Hu. "Polyphonic audio matching for score following and intelligent audio editors," *in Proceedings of the International Computer Music Conference (ICMC)*, 2003, pp. 27-34.

[5] Robert Macrae and Simon Dixon. "Accurate Real-time Windowed Time Warping," *in Proceedings of the International Society for Music Information Retrieval Conference,* 2010, pp. 423-428.

[6] Meinard Müller, Henning Mattes, and Frank Kurth. "An Efficient Multiscale Approach to Audio Synchronization," *in Proceedings of the International Society for Music Information Retrieval Conference,* 2006, pp. 192-197.

[7] Christian Fremerey, Meinard Müller, and Michael Clausen. "Handling Repeats and Jumps in Score-Performance Synchronization," *in Proceedings of the International Society for Music Information Retrieval Conference,* 2010, pp. 243-248.

[8] Siying Wang, Sebastian Ewert, and Simon Dixon. "Robust and Efficient Joint Alignment of Multiple Musical Performances," *in IEEE/ACM Transactions on Audio, Speech, and Language Processing,* 24(11): 2132-2145, 2016.

[9] Arshia Cont. "A coupled duration-focused architecture for realtime music to score alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):837-846, 2009.

[10] Christopher Raphael. "Music Plus One and machine learning," *in Proceedings of the International Conference on Machine Learning (ICML)*, 2010.

[11] Matthew Prockup, David Grunberg, Alex Hrybyk, and Youngmoo E. Kim. "Orchestral performance companion: Using real-time audio to score alignment," *IEEE Multimedia*, 20(2):52-60, 2013

[12] Frank Kurth, Meinard Müller, Christian Fremerey, Yoon-Ha Chang, and Michael Clausen. "Automated synchronization of scanned sheet music with audio recordings," *in Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2007, pp. 261-266.

[13] Verena Thomas, Christian Fremerey, Meinard Müller, and Michael Clausen, "Linking sheet music and audio—challenges and new approaches," *in Multimodal Music Processing*, vol. 3, pp. 1-22, 2012.

[14] Matthias Dorfer, Jan Hajič jr., Andreas Arzt, Harald Frostel, and Gerhard Widmer. "Learning Audio-Sheet Music Correspondences for Cross-Modal Retrieval and Piece Identification," *Transactions of the International Society for Music Information Retrieval*, issue 1, 2018.

[15] Tian Cheng, Matthias Mauch, Emmanouil Benetos, Simon Dixon, et al. "An attack/decay model for piano transcription," *in Proceedings of the International Society for Music Information Retrieval Conference,* 2017, pp. 584-590.

[16] Qi Wang, Ruohua Zhou, and Yonghong Yan. "A two-stage approach to note-level transcription of a specific piano," *in Applied Sciences,* 7(9):901, 2017.

[17] Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse Engel, Sageev Oore, and Douglas Eck. "Onsets and Frames: Dual-Objective Piano Transcription," *in Proceedings of the International Society for Music Information Retrieval Conference,* 2018, pp. 50-57.