ASSISTIVE ALIGNMENT OF IN-THE-WILD SHEET MUSIC AND PERFORMANCES

Michael Feffer
Carnegie Mellon University

Zachary C. Lipton
Carnegie Mellon University

Chris Donahue
Google Research

ABSTRACT

Sheet music, which contains precise instructions for performers, remains a primary mechanism for communicating musical ideas. While digital scans of sheet music (represented as images) and recordings of performances (represented as audio) are both abundant sources of musical data, there remains a surprising paucity of *aligned data*, mappings between pixels in sheet music and the corresponding timestamps in associated performances. While several existing MIR datasets contain alignments between performances and *structured scores* (formats like MIDI and MusicXML), no current resources align performances with more commonplace raw-image sheet music, possibly due to obstacles like expressive timing and repeat signs that make alignment challenging and time-consuming even for trained musicians. To overcome these obstacles, we developed an interactive system, MeSA , which leverages off-the-shelf measure and beat detection software to aid musicians in quickly producing *measure-level* alignments (ones which map bounding boxes of measures in the sheet music to timestamps in the performance audio). We verified MeSA 's functionality by using it to create a small proof-of-concept dataset, MeSA-13 .[1]

1 Introduction

Sheet music is a canonical format of music representation that has enabled musicians to communicate musical ideas for centuries. In recent years, old and new scores alike are readily available as sheet music that can be purchased online or downloaded for free using resources like the International Music Score Library Project (IMSLP), which in turn hosts over 680K sheet music scans as of November 2022[2] . Unlocking the information contained in sheet music and understanding how sheets map to performances is a nontrivial MIR task. Several released datasets [1–4] and approaches [5–7] have been proposed with related goals. However, these datasets consist of structured scores, such as MusicXML and MIDI, where underlying semantic structure directly maps to musical content. In contrast, sheet music in the form of images (e.g. PDF, PNG) only contain pixels with no hints on how to aggregate them to musical building blocks such as notes and phrases. As a result, models trained on structured scores are incompatible with sheet music. In addition, these existing datasets and approaches are typically limited to solo piano music (only [6] validates their approach with pieces of other instrumentation)—as such, techniques derived from these resources are unlikely to generalize to music performed by different ensembles (e.g., organ, string quartet).

While previous work has produced datasets to link (1) sheet music to structured scores for optimal music recognition (OMR) [8–10] and (2) structured scores to audio performances for transcription [11], to our knowledge, no “end-to-end” dataset exists that links sheet music to performances. A potential explanation for the lack of an end-to-end dataset in light of repositories like IMSLP is the difficulty of aligning sheet music and performances. Namely, we argue that a number of phenomena, including (but not limited to) expressive timing and repeated sections, make annotating slow and difficult, even for experts.

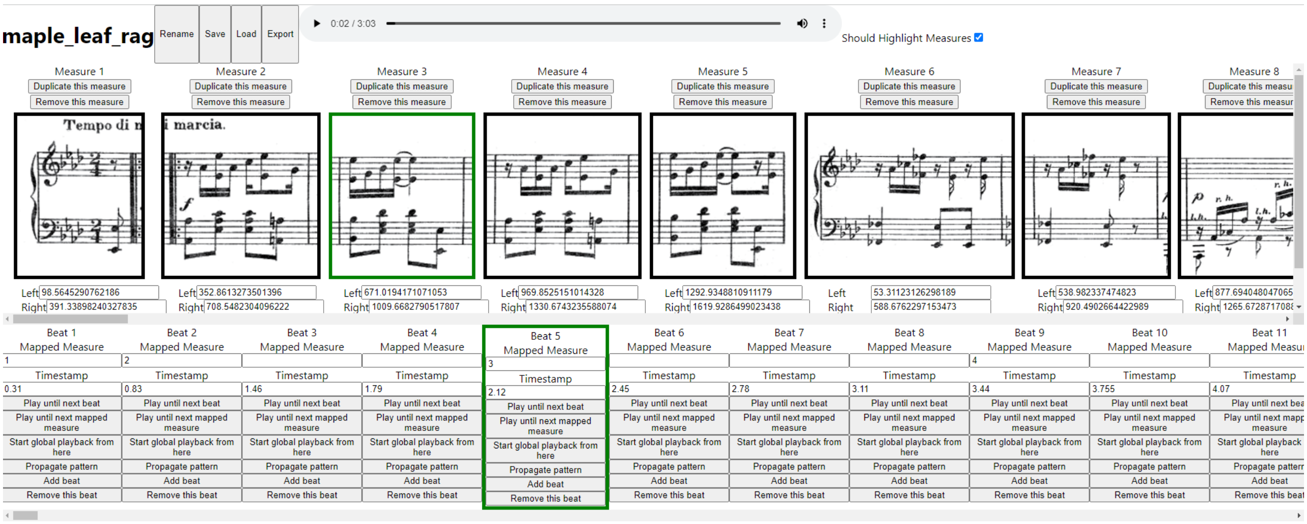
We first note there are a few different levels of granularity at which one could endeavor to align sheet music to performances. Simply pairing sheet music and audio files to align at the piece level is arguably too coarse to be useful for any application beyond recognition of specific pieces. On the other hand, the most granular approach, note-level alignment, would be the most useful, but we posit that producing such alignments would be expensive. Aligning at the line level can fail when both the score and corresponding performance have repeat signs. We therefore advocate for measure-level alignment, which is granular enough to gracefully handle repeats yet not so granular that alignment is infeasible. This being said, real-time alignment can still be difficult due to a performer’s expressive timing and an annotator’s unfamiliarity with a given piece, motivating the need for assistive tools to aid in creating such alignments.

To this end, we developed the *Measure to Sound Annotator* (MeSA), an interface in the form of a web application that processes sheet music and performances to pre-populate an initial alignment that a musician can audit and further refine. MeSA utilizes off-the-shelf measure detection [12] to specify instrumentation-agnostic measure bounding boxes in a given score as well as off-the-shelf beat tracking [13, 14] to specify beat timestamps in a given

[1] Preview: https://youtu.be/watch?v=WLVYJzgy7nU&list=PL4DZweX7nGV6XEFAES1r8gBC1BHxkFALK
Data: https://github.com/mfeffer/mesa-13

[2] https://imslp.org

Attribution: M. Feffer, C. Donahue, and Z. Lipton, “ Assistive alignment of in-the-wild sheet music and performances ”, in *Extended Abstracts for the Late-Breaking Demo Session of the 23rd Int. Society for Music Information Retrieval Conf.,* Bengaluru, India, 2022.



	Measure 1	Measure 2	Measure 3	Measure 4	Measure 5	Measure 6	Measure 7	Measure 8
Left	98.5645290762186	352.8613273501396	671.0194171071053	969.8525151014328	1292.9348810911179	53.31123126298189	538.982337474823	877.694048047065
Right	391.33898240327835	708.5482304096222	1009.6682790517807	1330.6743235588074	1619.9286499023438	588.6762297153473	920.4902664422989	1265.6728717088

	Beat 1	Beat 2	Beat 3	Beat 4	Beat 5	Beat 6	Beat 7	Beat 8	Beat 9	Beat 10	Beat 11
Mapped Measure	1	2			3				4		
Timestamp	0.31	0.83	1.46	1.79	2.12	2.45	2.78	3.11	3.44	3.755	4.07

Figure 1. Our Measure to Sound Annotator (MeSA) helps users create measure-level alignments between sheet music and performances. Above is a screenshot of MeSA 's frontend while annotating Joplin's "Maple Leaf Rag".

performance. A user is then able to select which beat begins an identified measure, modify detected bounding boxes and beat timestamps, and perform semi-automated alignment of all beats and measures based on sample alignments. We demonstrate functionality by using it to create a small, proof-of-concept dataset.

2 Methods

MeSA consists of two components: a backend server and frontend interface. The backend server processes scores and performance files. The frontend interface allows a user to modify and export a measure-level alignment between a provided score and performance.

Frontend. MeSA 's interactive frontend (Fig. 1) runs on React Typescript [15]. Given a score scan in the form of a PDF and a performance in the form of an audio file (e.g. MP3), MeSA produces heuristic measure bounding boxes with the resulting measure images and estimated beat timestamps. With this information, the user's task is to identify which beats are downbeats and, in turn, the measures in which they occur by recording measure numbers in corresponding text fields. If necessary, a user can modify the measure bounding boxes and beat timestamps using input widgets as desired. Once the user is satisfied with their work, they can use the frontend to export a JSON representation of their alignment.

Backend. MeSA 's backend runs on Python Flask [16] and handles file operations and metadata related to a user's alignment. Most importantly, it specifies bounding boxes for detected measures in a score as well as timestamps of detected beats in a performance. It sends this information to the frontend so that the user can construct the alignment. It also handles checkpointing logic and aggregating alignment information for exporting when requested by the user.

Ideal Workflow. A user creates a new project and uploads a score PDF and performance audio file pertaining to a musical work. There are no repeats or anacruses, and the performance has a steady, regular tempo. The backend detects measures and beats perfectly, so the user does not make any changes to heuristic bounding boxes and beat timestamps in the frontend. Upon receiving these results, the user indicates which beats correspond to the first and second downbeats, and then they use the system to propagate this alignment pattern and automatically label the rest of the downbeats with their corresponding measures. Lastly, the user exports the alignment as a JSON file.

Handling Real-World Issues. In practice, heuristic bounding boxes and timestamps may be inaccurate for a number of reasons (e.g. low-quality score caused measure detection issues, expressive timing in performance caused errors with beat tracking, etc.). The user can modify these estimates via text input elements until they are satisfied. Additionally, propagating an alignment pattern will fail if a piece contains repeats, but one can fix this by indicating the first and second downbeats of the repeated section and then propagating again. An anacrusis can similarly be remedied by propagating based on the second and third downbeats after mapping the first to the pick-up.

3 Results

We paid 4 musicians 20 dollars an hour for 5 hours to use MeSA to create alignments. The resulting alignment rate was approximately one alignment per annotator hour. This resulted in MeSA-13 , a proof-of-concept aligned dataset of 13 pieces. We plan to collect a larger dataset, though more refinement to our tool will be needed to further reduce costs.

4 References

[1] M. Dorfer, J. Hajič Jr, A. Arzt, H. Frostel, and G. Widmer, “Learning audio–sheet music correspondences for cross-modal retrieval and piece identification,” *Transactions of the International Society for Music Information Retrieval*, vol. 1, no. 1, 2018.

[2] F. Foscarin, A. McLeod, P. Rigaux, F. Jacquemard, and M. Sakai, “Asap: A dataset of aligned scores and performances for piano transcription,” p. 8.

[3] Q. Kong, B. Li, J. Chen, and Y. Wang, “Giantmidi-piano: A large-scale midi dataset for classical piano music,” no. arXiv:2010.07061, Apr 2022, arXiv:2010.07061 [cs, eess]. [Online]. Available: http://arxiv.org/abs/2010.07061

[4] L. Liu, V. Morfi, and E. Benetos, “Joint multi-pitch detection and score transcription for polyphonic piano music,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun 2021, p. 281–285.

[5] M. Dorfer, A. Arzt, and G. Widmer, “Learning audio-sheet music correspondences for score identification and offline alignment,” 2017.

[6] C. Fremerey, M. Clausen, and S. Ewert, “Sheet music-audio identification.” in *ISMIR*, 2009.

[7] D. Yang, A. Goutam, K. Ji, and T. Tsai, “Large-scale multimodal piano music identification using marketplace fingerprinting,” *Algorithms*, vol. 15, no. 5, p. 146, 2022.

[8] T. Tsai, D. Yang, M. Shan, T. Tanprasert, and T. Jenrungrot, “Using cell phone pictures of sheet music to retrieve midi passages,” *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3115–3127, 2020.

[9] J. Hajič jr. and P. Pecina, “The MUSCIMA++ Dataset for Handwritten Optical Music Recognition,” in *14th International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan, November 13 - 15, 2017*, Dept. of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University. New York, USA: IEEE Computer Society, 2017, pp. 39–46.

[10] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, “CVC-MUSCIMA: a ground truth of handwritten music score images for writer identification and staff removal,” *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 3, pp. 243–251, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10032-011-0168-2

[11] J. Thickstun, Z. Harchaoui, and S. M. Kakade, “Learning features of music from scratch,” in *International Conference on Learning Representations (ICLR)*, 2017.

[12] S. Waloschek, A. Hadjakos, and A. Pacha, “Identification and cross-document alignment of measures in music score images.” in *ISMIR*, 2019, pp. 137–143.

[13] S. Böck, F. Krebs, and G. Widmer, “Joint beat and downbeat tracking with recurrent neural networks.” in *ISMIR*. New York City, 2016, pp. 255–261.

[14] S. Böck, F. Korzeniowski, J. Schlüter, F. Krebs, and G. Widmer, “madmom: a new Python Audio and Music Signal Processing Library,” in *Proceedings of the 24th ACM International Conference on Multimedia*, Amsterdam, The Netherlands, 10 2016, pp. 1174–1178.

[15] F. Zammetti, *Modern Full-Stack Development: Using TypeScript, React, Node. js, Webpack, and Docker*. Springer, 2020.

[16] M. Grinberg, *Flask web development: developing web applications with python*. " O’Reilly Media, Inc.", 2018.