TOWARDS AUTOMATIC PRACTICE LOGGING

K. Siddharth Gururani & R. Michael Winters

Georgia Tech Center for Music Technology
840 McMillan St., Atlanta GA

ABSTRACT

There are many benefits to tracking and logging practice done with musical instruments. Traditional techniques use paper and pencil to record the content, quantity and quality of practice. In this paper, we propose a method to enable this process to occur automatically, with greater precision and ease than manual techniques. After providing additional motivation and context for automatic practice logging (APL), we describe its unique challenges. We then introduce an approach for labeling practice tracks using a form of audio-to-audio matching. Using a subset of 7 hours of practice, 128 tracks, and three pieces, our algorithm performs with 76.6% accuracy.

***Index Terms*—** Automatic Practice Logging, Cover Song Detection, Audio similarity

1. INTRODUCTION

Musicians spend hundreds if not thousands of hours practicing their instruments. Practice can occur in many geolocations, in various sound environments, and potentially with many different types of instruments. While performance might occur linearly, in a short time-span, and with few note-errors, practice is characterized by repetitions, mistakes, various tempi, and extreme fragmentation. These characteristics raise unique challenges to its automatic classification.

However, the landscape for automatic practice logging (APL) is rich. Considering the many thousands of musicians practicing a diversity of instruments on an almost daily basis, the amount of yet unclassified music being generated during practice would easily surpass that of performance and studio recordings. There are also many rewards for APL, including faster, less-intrusive, more precise, and more detailed logging than in manual methods. As a component to a mobile computing application, APL could lead to smarter, more efficient practicing, and provide useful information for instructors.

However, there are many challenges in APL that make the classification process more difficult than other music classification tasks. The system must be able to recognize the piece that is being played even for short excerpts, and must be robust to wrong notes, widely varying tempi, sporadic jumps within a piece, and also recognize when a piece has changed completely.

Despite these challenges, the potential rewards of an APL system are many, and have high-impact for many practicing musicians. In the following paper we present precursors and relevant methods that have been developed in the Music Information Retrieval (MIR) community, and describe the approaches we view as providing the most valuable framework for development. Using the features of Pitch Chroma and MFCCs, we present an algorithm capable of categorizing practice into one of three movements. We then introduce the methods used to create a test dataset of authentic practice from one solo pianist. We target our evaluation using a subset of approximately 7 hours of dedicated practice around one multi-movement work. After presenting our results, we provide some analysis of our errors and possible alternatives for future testing. We conclude by discussing steps for increasing robustness of our system and culminating in a useful practice logging application for many musicians.

2. RELATED WORK

Based upon a literature search of ISMIR proceedings, and to the best knowledge of the authors, the subject of automatic practice logging (APL) is an application space that has not yet been addressed. It does however draw important parallels with other application spaces in MIR, allowing similar frameworks to be used. Perhaps its closest neighbor is the subject of cover song detection, which in turn derives methodologies from audio fingerprinting [1], audio-to-audio or audio-to-score alignment [2] and audio similarity [3]. Other similar areas include subsequence search, performance analysis, and in this case, piano transcription. In this section, techniques of cover song detection are presented and compared with the unique requirements for an APL system.

The cover song detection problem is generally formulated as the following: Given a set of reference tracks and test tracks, look for tracks in the test set that are cover songs of a reference track. A few approaches to solve this problem are as following:

1. Chroma-based features used to train an SVM that classifies a reference/test pair as a reference/cover pair [4].
2. Dynamic time warping (DTW) of harmonic pitch class profiles [5]. The authors make use of con-

cepts from audio-to-audio alignment and subsequence search. DTW matches features extracted from two audio tracks. The cost of alignment is representative of the degree to which a track is a cover of another.

3. A system for large-scale cover song detection [6] where the authors modify a landmark based fingerprinting system [7]. The landmarks in this cover-song detection algorithm are chroma-based instead of frequency-based, as in the original fingerprinting algorithm.

By analogy to cover song detection, repertoire practice consists of fragments of that can be independently identified as belonging to a particular track. It is furthermore common for all the adjacent fragments to originate from the same piece. However, identifying the start and end times of a particular segment computationally is non-trivial, but must be the basis of a subsequence search algorithm (e.g. [8]). The subsequence search algorithm must furthermore be robust practice artifacts such as pauses, various tempi, missed notes, short repetitions, and sporadic jumps. Only as a musician attains mastery of a repertoire piece do the fragments become longer, with fewer mistakes, and with tempi that approach that of recordings. A concert performances of the piece may be the ultimate stage for practice, and closest to usable using a cover song detection algorithm.

3. METHODOLOGY

3.1. Problem Formulation

We separate the automatic practice logging (APL) task into two primary components: recognition of which repertoire piece is being practiced, and where in the piece the practice is occurring. The former would provide a general view into the content of practice while the latter would provide a more detailed view on the evolution of practice within a piece itself.

3.2. Approaches Considered

During the course of the project, a variety of approaches to APL were considered. Because of the diversity of types of practice and the desire to minimize dependence on external libraries of symbolic scores or audio-files, we hoped to find a method that could cluster tracks by similarity.

We began using a fingerprinting based approach that used the landmark-based fingerprinting algorithm mentioned in Sec. 2. We found that it performed well using a small dataset where a very similar practice was present as well, leading to some optimism. However, by time-stretching and compressing the test track, it was discovered that performance was poor for any form of manipulation, as expected from a fingerprinting system which is meant to search for exact recordings.

We then devised an approach based upon an audio-to-audio matching schema where the reference audio file was a complete piece with no mistakes. As an audio-based technique, it does not require a symbolic score, yet had the advantage of being a full, linear recording. Test tracks could then be classified according to which of the three movements they best matched.

4. ALGORITHM OVERVIEW AND DESCRIPTION

We pursued two approaches to identifying the movement: one based upon a heuristic-based subsequence search algorithm, and another that used an onset detection algorithm to perform searches using groups of adjacent notes.

4.1. Heuristic Similarity-Based Algorithm

Our first approach applied a simple heuristic to a pitch chroma based audio similarity algorithm [3] to quantify the similarity of a given test track with the three reference tracks. Fig. 1 is a simple block diagram describing the algorithm.

Fig. 1. Block Diagram for Similarity-Based Algorithm

First, a large window of 2 seconds and hop of 0.5 seconds is chosen for spectrogram calculation. The use of a large window helps to reduce the number of frames for distance computation and therefore reduce the running time of the algorithm. A distance distance matrix is then computed for each of the three (test, reference) pairs using 12 pitch chroma features and the first 13 MFCC features.

We then transform the three distance matrices into to their respective lag-distance matrices, as mentioned in [9] and apply a binary threshold t on the lag-distance matrix. We then apply the image processing technique of erosion and dilation to straight lines on the binary lag-distance matrix. The effect of erosion and dilation is to enhance larger areas of similarity and to attenuate discontinuous or small areas.

Our algorithm then sought to quantify the average length of lines in the lag matrix for each of the three files. After thresholding and erosion and dilation, the perimeter of resulting boundaries was calculated. The reference track for which the average perimeter length was greatest was chosen for the label.

These boundaries in the lag-matrix represent the similarity between a segment in the reference and the test track and provide some compensation for tempo differences. Using the average length of these boundaries can be contrasted with choosing the label based upon the longest boundary or the total amount of similarity. However, the longest boundary would not be robust to cases where practice might involve many short repetitions. Furthermore, the total amount of similarity would not take into account the importance of continu-

ity or length of lines. The average length of lines was therefore chosen as a compromise between these two extremes, and demonstrated better performance in informal testing.

4.2. Onset Segmentation Based Algorithm

A second approach was tested based upon experience annotating the dataset. Although practice is characterized by fragmentation, a trained ear can identify the piece using just a few notes in a row. It was reasoned that by probing a few adjacent notes in row against all possible notes in the three reference tracks, a strong peak would emerge that strongly indicated its belonging to a particular reference track. By using groups of a few adjacent notes spaced evenly across the reference track, the piece could be determined.

To make comparisons between sets of adjacent notes an onset detection algorithm needed to be written. For these purposes, a high-frequency content novelty-based onset detection was chosen [10]. After onsets were detected, sets of ten adjacent onsets were sampled from the test track at regularly spaced intervals. The pitch-class histogram of these ten onsets was then compared to the pitch-class histograms of all sets of ten adjacent onsets in the reference tracks. Fig. 2 is a simple block diagram that describes this method.

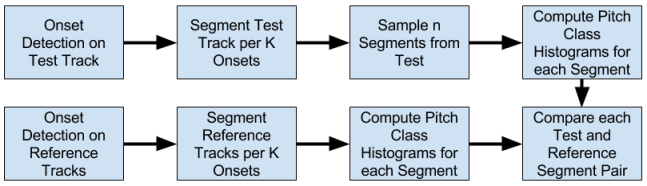

Fig. 2. Block Diagram for Onset Segmentation-Based Algorithm

After segmentation, n segments are tested at regular intervals from the start of the test track to the end. A pitch class histogram is computed for this sample and it is compared with pitch class histograms for every segment in the reference track. The segment is classified as the track for which there is highest similarity. Once all n segments are categorized, the mode of the array was used to to label the test track.

Testing this method however did not lead to results significantly above chance. Based upon our investigation, this performance was due to the onset detection algorithm. In future work, it would be useful to use an onset detection algorithm better suited for piano.

5. EVALUATION

5.1. Dataset

As a companion to this work, we present a dataset containing approximately 250 hours of practice by one musician on solo piano. The recordings were generated using a stereo, two-channel H4N microphone placed upon the piano stand or placed on a microphone stand and pointed at the harp. To automatically discard breaks in practice and long silences, the recorder was programmed to automatically turn on with SPL above a threshold, and to automatically turn off with SPL below a threshold that continues for four seconds. For data reduction, the recordings were automatically converted to MP3s using the H4Ns MP3 96kbps recording option.

The recordings provide a sampling of the variety of types of practice a solo musician might undertake, including sight-reading, technique, improvisation, ensemble, and repertoire work. Within these, repertoire work provides a particularly interesting subset. Repertoire practice is marked by repeated work of a few pieces, with small improvements over time. Tracking and logging this practice can be used by instructors and performers to evaluate the content, quality, and quantity of practice. This information can be used to assess improvement and gain insight into the time spent practicing.

5.2. Ground Truth & Methodology

To evaluate our algorithm, a ground-truth set of tracks needed to be annotated. From the original 250 hour dataset lasting one year, we narrowed our search to a more manageable 72 hours of practice that occurred during 45 days prior to a studio recording of Prokofiev's *Piano Sonata No. 4*. As is characteristic of a preparatory period, there is a high likelihood of playing this piece, facilitating annotation. Within this section, 128 recordings were labeled according to which of the three movements within the Sonata they corresponded. These 128 tracks comprised approximately 7 hours of practice.

We evaluated the performance of our algorithm using the accuracy of the categorization compensated by the chance level, which we define as

$$\text{Compensated Accuracy} = \frac{\text{\# of correct detections}}{(\text{\# of files}) \cdot (\text{\# of pieces})}.$$

Without compensating for the number of pieces, our average accuracy might stay the same, even as we added more possible pieces. However, this result should be avoided as a viable performance metric. If we added more reference pieces and still performed at the same accuracy level, our algorithm would clearly have improved. In the compensated accuracy function, this accuracy increases as chance level decreases. For misclassifications, the value also serves to mark how far below chance the algorithm performed.

5.3. Results

These compensated accuracy levels are displayed in Tab. 1. In the compensated technique, our true positives are roughly twice above chance level. All of the misclassifications are well below chance level. The average accuracy is 76.6%, which translates to a compensated accuracy of 2.3.

Table 1. Compensated Confusion Matrix (Chance = 1)

	Movt. 1	Movt. 2	Movt. 3
Movt. 1	2.2	0.6	0.3
Movt. 2	0.3	2.4	0.3
Movt. 3	0.5	0.2	2.3

The performance of our algorithm was found to be highly correlated with length of track. Tracks below 30 seconds long had approximately a 50% likelihood of being misclassified, whereas with more than 30 seconds, the algorithm performs well. These results are displayed in Fig. 3.

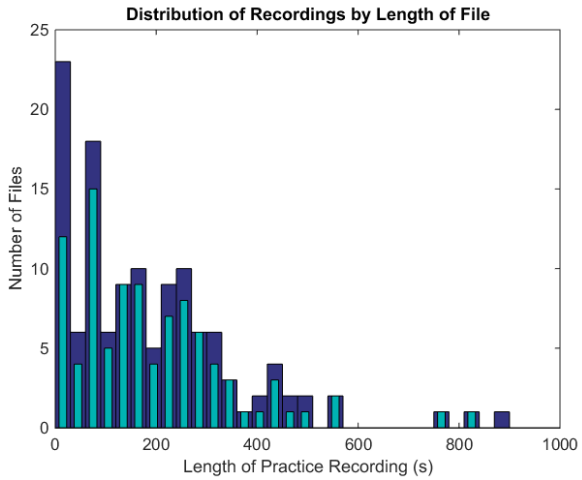

Fig. 3. Figure displaying the relationship of true positives to total files.

In our algorithm, there are two parameters that can be varied after the lag-distance matrix has been classified: the threshold of binarization t, and the length of the erosion/dilation line l. In order to determine desirable values for each, we used grid search in the parameter space for 25 files, revealing a blob of good performance with a peak around $l = 9$ and $t = 15$. These results are displayed in Fig. 4.

5.4. Discussion & Future Work

Through analysis of our results, it was discovered that for tracks below 30 seconds, our algorithm performs at roughly 50% accuracy, significantly less than for longer tracks. The reason for this might be that for longer tracks, there are more possibilities to have longer lines in the lag-distance matrix. However, for some of our shortest tracks (e.g. 6 seconds), the segment is very identifiable by ear as originating from a particular movement. In the future, a good approach would be to design the algorithm first around tracks below 30 seconds.

The second approach we attempted was based upon onset detection, but because it performed at chance level, it was not explored further. This result is problematic, and points

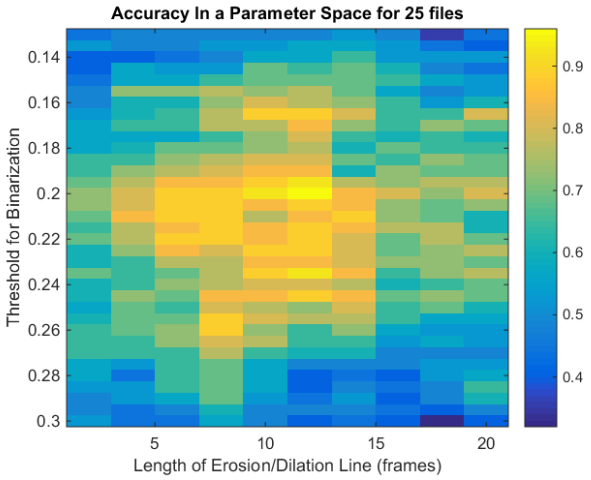

Fig. 4. A figure displaying the relationship between threshold of binarization and length of the erode/dilate line.

to a weakness in our implementation. In future work, the onset/note-based approaches should be reconsidered.

Our algorithm currently makes use of only two features: MFCCs and Pitch Chroma. We foresee that adding additional features related to temporal, intensity, or more specific spectral features such as spectral centroid, spread and kurtosis might provide increased performance. Additional features and feature weighting may be key to robust alignment [11].

One of the benefits of using audio-to-audio matching is that it could be expanded to be used without an external dataset. Clustering could occur based only upon previously recorded tracks from the performer. In the future, it would be interesting to explore this clustering further.

As discussed in Sec. 5.1, additional methods might break practice into more general components such as ensemble work, technique, sight-reading and improvisation. Within this categorization, repertoire practice would provide a sizeable subset.

6. CONCLUSION

We present the first steps toward an approach to automatic practice logging. Using concepts and approaches from audio similarity and the features of pitch chroma and MFCCs, we are able to classify practice tracks into one of three movements significantly above chance level. By adding more features, and making further modifications to our algorithm, we expect that performance would increase further.

The dataset we are using contains 250 hours of practice recordings, which provide much fodder for future investigation. By continuing to pursue APL, we hope that our work might eventually provide a robust practice logging application that can be used by musicians everywhere.

7. REFERENCES

[1] Pedro Cano, Eloi Batlle, Ton Kalker, and Jaap Haitsma, "A review of audio fingerprinting," *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, vol. 41, no. 3, pp. 271–84, 2005.

[2] Alexander Lerch, *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*, chapter 7, pp. 139–50, Wiley-IEEE Press, 2012.

[3] Ning Hu, Roger B. Dannenberg, and George Tzanetakis, "Polyphonic audio matching and alignment for music retrieval," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 185–8.

[4] Suman Ravuri and Daniel P. W. Ellis, "Cover song detection: from high scores to general classification," in *Proceedings of the IEEE International Converence on Acoustics Speech and Signal Processing*, 2010, pp. 65–8.

[5] Joan Serra, Emilia Gómez, Perfecto Herrera, and Xavier Serra, "Chroma binary similarity and local alignment applied to cover song identification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 6, pp. 1138–51, 2008.

[6] Thierry Bertin-Mahieux and Daniel P. W. Ellis, "Large-scale cover song recognition using hashed chroma landmarks," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2011, pp. 117–20.

[7] Avery Li-Chun Wang, "An industrial strength audio search algorithm.," in *Proceedings of the 4th International Conference on Music Information Retrieval*, 2003, pp. 7–13.

[8] AnYuan Guo and Hava Siegelmann, "Time-warped longest common subsequence algorithm for music retrieval," in *Proceedings of the 5th International Conference on Music Information Retrieval*, 2004, pp. 258–61.

[9] Jonathan Foote, "Automatic audio segmentation using a measure of audio novelty," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2000, pp. 452–5.

[10] Juan Pablo Bello, Laurent Daudet, Samer Abdallah, Chris Duxbury, Mike Davies, and Mark B Sandler, "A tutorial on onset detection in music signals," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 1035–47, 2005.

[11] Holger Kirchhoff and Alexander Lerch, "Evaluation of features for audio-to-audio alignment," *Journal of New Music Research*, vol. 40, no. 1, pp. 27–41, 2011.