BPS-MOTIF: A DATASET FOR REPEATED PATTERN DISCOVERY OF POLYPHONIC SYMBOLIC MUSIC

Yo-Wei Hsiao[1] **Tzu-Yun Hung**[1,2] **Tsung-Ping Chen**[1] **Li Su**[1]
[1]Academia Sinica, Taiwan [2]National Taiwan Normal University, Taiwan
lisu@iis.sinica.edu.tw

ABSTRACT

Intra-opus repeated pattern discovery in polyphonic symbolic music data has challenges in both algorithm design and data annotation. To solve these challenges, we propose BPS-motif, a new symbolic music dataset containing the note-level annotation of motives and occurrences in Beethoven's piano sonatas. The size of the proposed dataset is larger than previous symbolic datasets for repeated pattern discovery. We report the process of dataset annotation, specifically a peer review process and discussion phase to improve the annotation quality. Finally, we propose a motif discovery method which is shown outperforming baseline methods on repeated pattern discovery.

1. INTRODUCTION

Repetition is ubiquitous in music. Computational discovery of repeated patterns in music data has been long discussed in the field of music information retrieval (MIR). Aside from its importance in music analysis [1], the role of repeated pattern discovery has also been noticed in music classification [2,3] and generation [4,5]. The definition of a pattern is multi-fold. Generally speaking, a pattern refers to a group of notes that serves a musically important role and occurs multiple times in a piece of music. Repeated patterns are known by various names, such as motifs, themes, phrases and sections, depending on their specific musical function. The goal of the *repeated pattern discovery* problem is then to find the relevant patterns (depending on the intended task) and all of their occurrences within the provided musical data.

Compared to other music analysis tasks (e.g., harmony analysis) on polyphonic symbolic music data, repeated pattern discovery is relatively less discussed due to mainly two challenges. First, searching for all the possible candidates of repeated patterns is costly and redundant [6]. The computational complexity of the algorithm is high, while the discovered patterns often have little musical significance [7]. Second, repetition is a non-exact attribute of music. A large pattern can be potentially divided into small ones; whether a note group constitutes a meaningful repeated pattern also depends on the subjective views regarding repetition, similarity, and musical importance. As a result, human-annotated datasets that comprehensively identify all the available patterns and all of their occurrences remains in a quite limited scale.

In this paper, we propose a new dataset, BPS-motif, to improve the scalability of music pattern discovery research. The BPS-motif dataset contains the note-level annotation of motives and their occurrences in the first movements of Beethoven's Piano Sonatas (BPS). This is an extension of the many previous musical annotations on BPS, such as the functional harmony, phrase and section annotation provided in the Beethoven Piano Sonata Functional Harmony (BPS-FH) dataset [8]. We are specifically interested in annotating the *motivic* units in the melody parts of each piece of music, which could be complementary to the more *thematic* annotation (e.g., phrases and sections) provided in the BPS-FH dataset. We expect that the proposed dataset can enrich not only multi-task MIR research but also novel computational music analysis tasks.

Besides, as another contribution of this paper, we also propose a simple yet effective algorithm for repeated pattern discovery. Different from previous works which emphasized the equal translations among notes, we emphasize the contextual relationships among short segments of notes. We demonstrate that the proposed algorithm not only outperform several baselines on the BPS-motif dataset, but also on the JKU-PDD dataset [9], the most widely used dataset for the discovery of repeated themes and sections. In other words, the proposed algorithm is competitive for finding both motivic and thematic patterns.

The rest of this paper is organized as follows. Section 2 gives a background introduction and a survey of previous works on the datasets and methods for repeated pattern discovery. In Section 3, we introduce the dataset and our proposed annotation process. In Section 4, we introduce the proposed motif discovery algorithm and demonstrate its evaluation results. Conclusions are made in Section 5.

2. RELATED WORK

2.1 Repeated pattern discovery datasets

The datasets for repeated pattern discovery are built mostly for the interest in computational music analysis research. Complete annotation of repeated patterns should incorporate all the note groups (each note in pitch-onset part) that constitute 1) the patterns of interest and 2) the occurrences of each pattern. Usually, a music piece contains more than one pattern, and each pattern should repeat (i.e., occur more than twice). The occurrences of a pattern may not be the same; one occurrence can be an exact copy of, or a variation from another occurrence that belongs to the same pattern. The music data can be either monophonic or polyphonic, and can be in either symbolic or audio format. The annotation can be either *intra-opus* or *inter-opus* [10]. In the former case, the analysis focuses on how a piece of music is broken down into pattern occurrences by having occurrences of a pattern within one music piece [7, 10]. In the latter case, the analysis focuses on the evolution of common elements in a corpus, by having occurrences of a pattern in different pieces of music. It should be noted that the annotation in inter-opus datasets can be limited to only a small set of patterns, while intra-opus datasets need a comprehensive set of patterns and occurrences and is hard to build; see Table 1 and the discussion below.

Table 1 presents the datasets for both inter-opus and intra-opus pattern discovery. In the Saraga dataset, Srinivasamurthy *et al.* annotated 4,571 temporal occurrences from 1,067 *characteristic melodic phrases*, a musical unit related to the *rāga*, over 170 audio recordings [11]. Krause *et al.* performed large-scale leitmotif classification in audio recordings by annotating the time intervals of 10 leitmotifs in Richard Wagner's four-opera cycle *Der Ring des Nibelungen* and achieve a large scale of occurrence over 16 versions of recordings [12]. [1] In the MTC-ANN dataset, Kranenburg *et al.* categorized 93 patterns in 360 monophonic folk tunes and annotated 1,657 occurrences [13]. Finkensiep *et al.* considered 20 types of *schemata* and annotated 244 events in Mozart's piano sonatas [14]. In the MIREX campaign of *Discovery of Repeated Patterns and Themes*, Collins *et al.* firstly compiled an organized, open-source intra-opus pattern discovery dataset containing 165 occurrences in five pieces and this dataset has been widely discussed in the follow-up research works. It should be noted that, among these datasets, only the JKU-PDD dataset is for intra-opus pattern discovery but its size is smallest among all (only five pieces of music).

Aside from the above-mentioned datasets, it is still worth mentioning the datasets for *pattern matching* [15], such as the Dig That Lick dataset for Jazz music [16] and the Theme Finder for Classical music [17]. These datasets support pattern retrieval tasks with known query, but they neither support pattern discovery research nor provide the annotation of pattern occurrences explicitly.

2.2 Repeated pattern discovery methods

For symbolic music data, there are three major approaches to implementing the repeated pattern discovery algorithms: 1) string-based approach which represents music data as one-dimensional pitch sequence and finds repeated patterns with sub-string matching [18, 19]; 2) geometry-based approach which represents music data as multi-dimensional point sets (usually onset-pitch pairs in two-dimensional space) and retrieves the translatable subsets (see discussion below) as repeated patterns [20–22]; 3) feature-based approach which extracts or learns features from music data, and retrieves patterns with clustering or classification of the features [14, 23–25].

[1] There are in total 38,448 occurrences if counting the 16 versions.

	format	usage	#ps	#ptns	#ocrs
[11]	poly audio	inter	170	1,067	4,571
[12]	poly audio	inter	11	10	2,403
[13]	mono symbolic	inter	360	93	1,657
[14]	poly symbolic	inter	54	20	244
[9]	poly symbolic	intra	5	32	165
Ours	poly symbolic	intra	32	263	4,944

Table 1: Comparison of several open-source musical repeated pattern datasets including the saraga dataset [11], The *Ring* (one performance version) [12], MTC-ANN [13], Schemata [14], JKU-PDD [9], and BPS-motif (ours). The number of pieces (#ps), the number of individual patterns (#ptns), and the number of occurrences (#ocrs) are listed. The data formats can be monophonic (mono) or polyphonic (poly), audio or symbolic. The type of annotation can be inter-opus (inter) or intra-opus (intra).

While the string-based approach falls limited in representing polyphonic music [22], research efforts on pattern discovery have been more emphasized on the geometry-based approach. In the geometry-based approach, we consider a music piece $\mathbf{D}$ with N notes and $\mathbf{d}$ denotes a note. We have $\mathbf{D} := \{\mathbf{d}_i\}_{i=1}^N$, where $\mathbf{d}_i := (o_i, p_i)$ denotes the ith note, and o_i, p_i denote its onset and pitch value, respectively. In the discussion of the structure induction algorithm with translational equivalence classes (SIATEC) [20], two subsets (i.e., two patterns) $\mathbf{m}$ and $\mathbf{n}$ in $\mathbf{D}$ are translatable (denoted as $\mathbf{n} \equiv \mathbf{m}$) if there exists a vector $\mathbf{v}$ such that the translation function $f(\mathbf{d}, \mathbf{v}) : \mathbf{m} \to \mathbf{n}; \mathbf{d} \mapsto \mathbf{d} + \mathbf{v}$ is bijective. All the patterns translatable with respect to $\mathbf{m}$ form a translational equivalence class (TEC) of $\mathbf{m}$ in $\mathbf{D}$, that means

$$\text{TEC}(\mathbf{m}, \mathbf{D}) := \{\mathbf{n} : \mathbf{n} \equiv \mathbf{m}, \mathbf{n} \subseteq \mathbf{D}\} . \quad (1)$$

A *maximal translatable pattern* (MTP) is the largest pattern translatable by a translatable vector $\mathbf{v}$ [20]:

$$\text{MTP}(\mathbf{v}, \mathbf{D}) := \max_{|\mathbf{d}|}\{\mathbf{d} : \mathbf{d} \in \mathbf{D} \text{ and } \mathbf{d} + \mathbf{v} \in \mathbf{D}\} , \quad (2)$$

where $|\mathbf{d}|$ is the number of notes in $\mathbf{d}$. SIATEC is then an algorithm which finds all the TEC of the available MTPs in $\mathbf{D}$. A survey and comparative study can be found in [26].

In the feature-based approach, machine learning techniques are usually applied; features are processed by clustering for the pattern discovery task (when a query is not given), and by classification for the pattern matching task (when a query is given) [15]. For example, in [23], agglomerative clustering over the wavelet transform of the pitch sequence data was used for pattern discovery in melodies. In [14], music schema recognition was performed by extracting the schema candidates using a skip-gram model and then a binary classification on the rhythm and pitch features over the candidates. It is also noted that the feature-based approach has also been widely discussed in the repeated pattern discovery of audio. In [24], Nuttall *et al.* adopted matrix profile, a time-series-based motif discovery method [27], on the predominant pitch contours to extract the characteristic melodic phrases from audio [11]. Krause *et al.* utilized recurrent neural networks (RNN) to classify over 30,000 leitmotifs over differnt performances of *Der Ring des Nibelungen* [25].

3. DATASET

3.1 Overview

The BPS-motif dataset contains the annotation of motives in the first movements of Beethoven's 32 piano sonatas. An annotation unit contains a group of motif notes and the corresponding motif label. The motif labels are sorted in alphabetical order: the motif that occurs first in the music piece is labeled as A, the secondly occurred motif is labeled as B, the thirdly occurred one is C, and so on. The group of notes which are the jth occurrence of the motif A in $\mathbf{D}$ is annotated as $\mathbf{m}_{A,j}$, $\mathbf{m}_{A,j} \subset \mathbf{D}$, $j \in \mathbb{Z}_{\geq 0}$. All the occurrences of this motif are annotated with A. Further information, such as the start time and end time of each motif occurrence, and the non-motif notes (i.e., the notes which do not belong to any motif) can be directly derived. The dataset is available at: `https://github.com/Wiilly07/Beethoven_motif`.

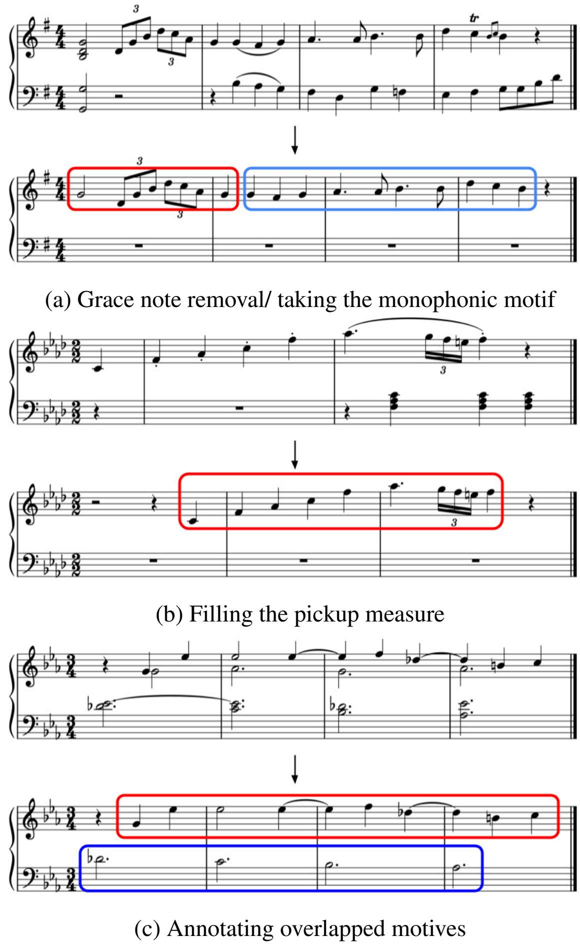
Over the 32 music pieces, we labeled 263 distinct motives with 4,944 occurrences in total (see Table 1). These occurrences contain 36,652 notes, which is 28.87% of the total number of 126,943 notes. For each piece of music, the number of motives ranges from 2 to 13 (average 8.22 motives), and the number of occurrences ranges from 41 to 290 (average 154.5 occurrences). On average, a motif contains 7.41 notes and spans 5.30 crochet beats. The pitch ranges of the motives are mostly within two octaves.

To facilitate the annotation process, we only consider the repeated patterns in melodic notes; that means, all the annotated motives are constrained to be a monophonic note sequence. For example, in Figure 1a, although the first beat of the first measure contains three notes (i.e., B3, D4, G4), only G4 is included in the the annotated motif `A0`. However, there can be multiple motives which are fully or partly overlapped in time; see the demonstration in Figure 1c (the red and blue boxes represent two overlapped motives).

3.2 Data format

We basically followed the data format adopted in the BPS-FH dataset. First, all the articulation symbols and grace notes were omitted (see Figure 1a). Second, pickup was filled when needed (see Figure 1b and the following discussion). Repeat signs are also unfolded when needed.

We take a crotchet beat as the unit time step (i.e., the duration of a crotchet is 1 in our note annotation and is 1 second in MIDI) to represent the data. Two types of timestamps are recorded. The *score time* takes the pickup measure as negative while the *MIDI time* fills the pickup measure and defines the beginning of the measure as 0. For example, in Figure 1b, the score time of the C4 note at the beginning is -1 while the MIDI time is 3. Both the score time and the MIDI time unfolds the repeat signs so the timestamps increase monotonically. Similarly, at the measure level, the *score measure number* is the measure number counted on the score sheet (the pickup measure is measure 0, with repeat signs), while the *MIDI measure number* takes the pickup measure (if there is) as measure 1 and unfolds the repeat signs. Two types of pitch number are recorded: the MIDI pitch (in MIDI number) and the morphetic pitch number [28].



(a) Grace note removal/ taking the monophonic motif

(b) Filling the pickup measure

(c) Annotating overlapped motives

Figure 1: Examples of annotated motives. From (a) to (c), the three demonstrated excerpts are from Beethoven's Piano Sonata No. 20, No. 1, and No. 5, respectively. The notes bounded by a colored box form a motif occurrence.

For each piece of music, we provide annotation data in different formats for users to retrieve the motif events in different ways. The file formats include:

1. A multi-track MIDI file that records the motif notes. Temporally overlapped motives are recorded in different tracks. There are at most four tracks in our annotation of this dataset.

2. A list of all the notes. Each note has the labels of 1) onset time (in score time), 2) MIDI pitch number, 3) morphetic pitch number, 4) note duration (in

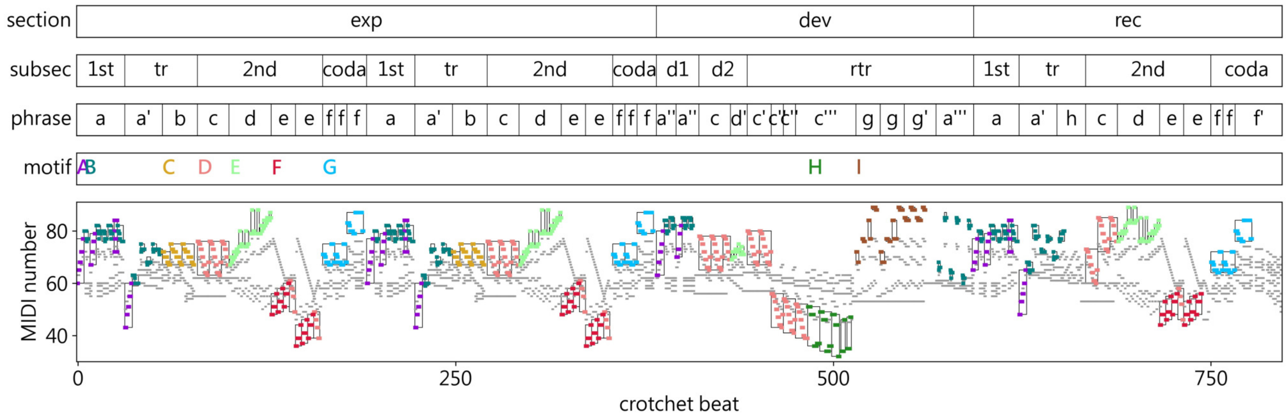


Figure 2: Motives and occurrences labeled in Beethoven's Piano Sonata No.1 in F minor. From top to bottom shows the annotation of section intervals, subsection intervals, phrase intervals, the time when a new motif occurs (with motif labels), and the piano roll of the music piece marked with motif and non-motif notes. In the bottom subfigure, different motives are specified by different colors. Motif occurrences are marked with a black bounding box. Non-motif notes are in gray color.

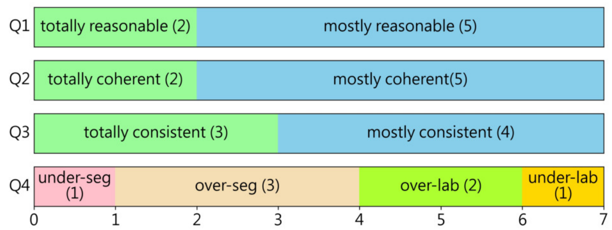


Figure 3: Assessment results (Q1 to Q4) regarding the data annotation from the seven reviewers. The results were collected before the discussion phase.

crotchet beats), 5) staff number (integers from zero for the top staff), 6) MIDI measure number, and 7) motif (e.g., a note is annotated as A if it is part of A). The notes without motif labels are non-motif notes.

3. Individual note lists of each motif occurrence. These lists are provided for users to better retrieve each occurrence. The labels in these lists are the same as the ones in the list of all notes.

4. A list describing the properties of motif occurrences. Each motif occurrence has the labels of 1) the start time and end time (in both score time and MIDI time), 2) the duration of the occurrence, 3) the measure number where the motif start (in both score measure and MIDI measure), 4) the "start beat" of the motif start, and 5) time signature.

We also provide the PDF scoresheets with the annotator's manual annotations and notes. These scoresheets are for reference only because they are the raw annotations and may not be the same as the annotation of our final version; see Section 3.3 for more details about the annotation process. The note lists and the score time are compatible with the BPS-FH dataset, therefore annotation of more thematic units (e.g., theme, sub-section and phrase) can be retrieved from the BPS-FH dataset. To better see our annotation result, Figure 2 illustrates the hierarchical musical structures with motives of Beethoven's Piano Sonata No. 1, combining the section, subsection and phrase labels in BPS-FH, and the motif labels in BPS-motif. [2]

3.3 Annotation process

There are a few challenges in the data annotation process. First, as mentioned, identifying of musical motifs and their repetitions or variations in a piece of music is not straightforward. Ambiguity arises from multiple factors. For example, some repeated patterns may not be considered as valid musical motifs, a motif may not always be the smallest unit of a repeated pattern, and the similarity or difference between two such sequences can also be subject to human interpretation. Besides, while experienced musicians can read the scoresheet and mark the motives directly by hand on it, converting such hand-drawing annotations into database formats still requires lots of efforts.

Our proposed approach to build the BPF-motif dataset incorporates three parts: annotation, review, and score typing. First, two annotators (the first and the second authors) manually annotate the motives on the scoresheet. Each piece is annotated by one annotator. Then, we invite external reviewers to review annotated scoresheets. Also, the reviewer helps us digitize the manual annotation. In the review process, we design a review form to let the reviewers assess the overall quality of annotation and also provide their suggested annotation if they hold different opinions. The review form contains the following questions:

1. (Q1) Are the annotations reasonable? (3: totally reasonable; 2: mostly reasonable; 1: unreasonable)

2. (Q2) Are the annotations coherent with your opinion? That means, if you were the annotator, will you

[2] It should be noted that there are still some annotation inconsistency between the BPS-FH and BPS-motif datasets. For example, in Figure 2, the phrase c is constructed only with the motif D, while the phrase c''' is constructed only with the motif H. This means that while the annotator of BPS-FH considered c''' as simply a variation of c, the annotator of BPS-motif considered them being different (and are thereon constructed with different motives).

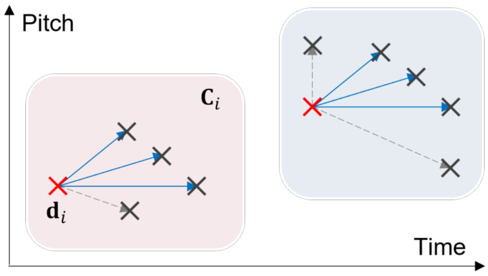


Figure 4: Two segments (light gray and light purple regions) and their common structure. The crosses indicate notes, and the set of vectors in a segment represents its structure. Blue vectors denote the common structure which exists in both segments. The dashed gray arrows represents non-motivic notes within the two segments.

also have the same annotations as ours? (3: totally coherent; 2: mostly coherent; 1: incoherent)

3. (Q3) Are the annotations consistent (i.e., did we hold consistent criteria annotating the data)? (3: totally consistent; 2: mostly consistent; 1: inconsistent)
4. (Q4) In which way your opinions are different from ours? (a: we took multiple motives into one (*undersegmentation*); b: we divided a motif into many (*oversegmentation*); c: we took some non-motif patterns as motives (*overlabeling*); d: we omitted some musically important motives (*underlabeling*)) Choose one even you totally agree to our annotation.
5. (Q5) If you hold different opinions on our annotation and think we should revise them, leave your comments explicitly. Your comments can be, for example, "the motif E in Sonata No. x should be further divided into F and G" (describe what F and G are); "the motif H in Sonata No. y can be considered as a variation of B and should be merged," etc.

Seven reviewers were invited to review the annotations. The reviewers are all from composition background and are good at using computer scorewriters. Each reviewer was assigned from 3 to 7 pieces (according to the length of the music piece) for review, then they answered the above questions and provided their suggested annotations on a co-edited document. During the review and discussion phase, the reviewers also need to convert the manual annotation on the score into the symbolic form using the scorewriter MuseScore. This confirms that they had carefully read the annotation, and also speed up the process of building the dataset. After the reviewers typed the scores of the annotated motives, we can directly convert it to MIDI and the final annotation data.

The reviewer's assessment results are shown in Fig. 3. From Q1 to Q3, it is shown that no reviewer reported our annotation as unreasonable, incoherent to their thoughts, or self-inconsistent. However, over half of the reviewers did point out a few annotation they considered problematic. We discussed with the reviewers regarding those issues and revised them such that all the annotations are acceptable for the reviewer. The result of Q4 shows that reviewers tend to say our annotations are oversegmented. This however fits our needs because doing this provides extra flexibility to the dataset; researchers who are interested in longer repeated patterns can simply merge our annotations. On the other hand, it is hard to retrieve short motivic patterns from undersegmented annotation.

Algorithm 1 Find Common Structure

1: **function** COMMON STRUCTURE($\mathbf{D}, \Delta t$)
2: $\quad \mathbf{S} \leftarrow \emptyset$
3: $\quad$ **for** $i \leftarrow 1$ to $N-1$ **do**
4: $\quad\quad \mathbf{C}_i \leftarrow \emptyset$
5: $\quad\quad$ **for** $j \leftarrow i+1$ to N **do**
6: $\quad\quad\quad$ **if** $\mathrm{o}_j - \mathrm{o}_i < \Delta t$ **then**
7: $\quad\quad\quad\quad$ add $\mathbf{d}_j$ to $\mathbf{C}_i$
8: $\quad\quad\quad$ **end if**
9: $\quad\quad$ **end for**
10: $\quad\quad \mathbf{S}_i \leftarrow \{\mathbf{d}_j - \mathbf{d}_i,\ \mathbf{d}_j \in \mathbf{C}_i\}$
11: $\quad\quad$ add $\mathbf{S}_i$ to $\mathbf{S}$
12: $\quad$ **end for**
13:
14: $\quad \mathbf{M} \leftarrow \emptyset$
15: $\quad$ **for** $i \leftarrow 1$ to $N-1$ **do**
16: $\quad\quad \hat{\mathbf{S}} \leftarrow \emptyset$
17: $\quad\quad$ **for** $j \leftarrow i+1$ to N **do**
18: $\quad\quad\quad$ add $\{\mathbf{S}_i \cap \mathbf{S}_j\}$ to $\hat{\mathbf{S}}$
19: $\quad\quad$ **end for**
20: $\quad\quad$ add MOST_COMMON($\hat{\mathbf{S}}$) to $\mathbf{M}$
21: $\quad$ **end for**
22: $\quad$ **return** $\mathbf{M}$
23: **end function**

4. MOTIF DISCOVERY

4.1 Algorithm

We regard a motif as a short pattern recurring with little change in its structure. In other words, the relative positions of the notes in a motif will be almost fixed. We therefore find motifs by detecting *common structures* in short musical segments. The idea of the proposed algorithm is presented in Figure 4 and Algorithm 1. Formally, let Δt denote a threshold of time interval, and $\mathbf{D} := \{\mathbf{d}_i\}_{i=1}^N$ a musical piece composed of N notes sorted in ascending order, with $\mathbf{d}_i = (\mathrm{o}_i, \mathrm{p}_i)$ being a two-dimensional vector indicating the onset and pitch number of the ith note. For $\mathbf{d}_i$, we first aggregate its context $\mathbf{C}_i$ and create a segment $\mathbf{S}_i$. The derived segments are then compared pairwisely to obtain common structures. By representing a segment as a set of *vectors* (see Figure 4 and Line 3–12 in Algorithm 1), the common structure of any two segments (i.e., the blue arrows in Figure 4) can be obtained by collecting vectors which exist in both segments (Line 18).

As the pairwise comparisons between segments (Line 15-19) will result in various types of common structures, we retrieve a representative pattern and all its occurrences by finding the "most common" structure (i.e., the com-

Algorithm	P_{est}	R_{est}	F_{est}	P_{occ}	R_{occ}	F_{occ}	P_{thr}	R_{thr}	F_{thr}	Runtime
SIATEC	0.1804	0.6444	0.2803	0.2102	0.2771	**0.2235**	0.0408	**0.2994**	0.0713	**28.5082**
COSIATEC	0.2118	0.4557	0.2863	**0.2769**	0.1282	0.1548	0.0489	0.1601	0.0743	208.4119
SIATECCompress	0.2136	0.4326	0.2835	0.1430	0.1121	0.1103	0.0579	0.1703	0.0856	636.6930
Proposed	**0.5709**	**0.8339**	**0.6733**	0.1491	**0.4174**	0.2002	**0.1222**	0.2644	**0.1646**	119.5330

(a) Motif discovery on the proposed dataset

Algorithm	P_{est}	R_{est}	F_{est}	P_{occ}	R_{occ}	F_{occ}	P_{thr}	R_{thr}	F_{thr}	Runtime
SIATEC	0.1238	0.4630	0.1920	**0.5248**	0.3970	**0.4437**	0.0706	**0.4006**	0.1176	**1.5099**
COSIATEC	0.1140	0.2530	0.1491	0.1305	0.0870	0.1044	0.0740	0.2042	0.1027	6.0167
SIATECCompress	0.1807	0.2849	0.2181	0.1778	0.0889	0.1185	**0.1117**	0.2202	0.1470	34.6371
Proposed	**0.2649**	**0.5002**	**0.3406**	0.4208	**0.5105**	0.3948	0.1096	0.3003	**0.1561**	4.0546

(b) Repeated pattern discovery on the JKU-PDD dataset

Table 2: Evaluation of pattern discovery algorithms. The subscripts *est*, *occ*, and *thr* indicate the *establishment*, *occurrence*, and *three-layer* measurements, respectively. The averaged runtime is in minutes.

mon structure that occurs the most times) in $\hat{\mathbf{S}}$ with the MOST_COMMON operation (Line 20). Finally, motifs are acquired by filtering out non-motivic patterns in $\mathbf{M}$ heuristically. In this work, we set $\Delta t = 12$ crotchet beats.

The proposed algorithm differs from the SIA family in two aspects. First, the SIA family aggregates notes of a pattern by detecting equal *translations* among notes, while our algorithm finds patterns by identifying common structures, or *contextual relationships*, among small segments. Second, the SIA family computes maximal translatable patterns (MTP) and subsequently find their occurrences, whereas our algorithm establishes a small pattern and all its occurrences at the same time. Our approach is promising in that the contextual comparisons between segments help identify motifs which are small and recurring. The code of the proposed algorithm is available at `https://github.com/Tsung-Ping/motif_discovery`.

4.2 Evaluation

We evaluate the motif discovery algorithm on the proposed dataset (with an averaged number of 3937 notes per piece) as well as the JKU-PDD dataset (1284 notes in average) [9] using standard metrics for pattern discovery. The *establishment measurement* (est) shows the capability of an algorithm to recognize patterns rather than to find all occurrences of a pattern. The *occurrence measurement* (occ), on the contrary, emphasizes the ability to find all occurrences of a pattern. The *three-layer measurement* (thr) is a comprehensive evaluation combining aspects of both the establishment and occurrence measurements. Each of the three measurements are specified in terms of precision, recall, and F1 score. [3] The *averaged runtime* on each dataset will also be measured to give a rough sketch of the time complexity. We compare the proposed algorithm with three methods from the SIA family, i.e., SIATEC [20], COSIATEC [21], and SIATECCompress [21]. [4] All algorithms were implemented in Python programming language.

The evaluation results are summarized in Table 2. Generally, our algorithm performs consistently across datasets despite that the two datasets are composed of distinct types of musical patterns, i.e., *motivic* (the proposed) versus *thematic* (the JKU-PDD), which differ with each other mainly in the size. Our algorithm is superior to the baselines in all the three establishment measures, indicating that our method can identify more existences of the ground-truth patterns than the other algorithms. Besides, our algorithm is competent in the other two measurements, with at least one best performance in each measurement. Specifically, the R_{occ} measure shows that the patterns retrieved by our algorithm are more complete (i.e., discovering more occurrences of a pattern) with respect to the ground-truth patterns, and the F_{thr} measure suggests that our method has better capability to recognize salient patterns in music, especially the motivic ones. Finally, the runtime measurement indicates that our algorithm can achieve a better performance on the pattern discovery tasks at a moderate computational cost, which is 4.2 (resp. 2.7) times slower than the SIATEC on the proposed (reps. JKU-PDD) dataset.

5. CONCLUSION

We have demonstrated a dataset for repeated pattern discovery of polyphonic symbolic data and a motif discovery algorithm. Our data annotation clearly demonstrates the hierarchical structure of music. The proposed motif discovery algorithm has been shown outperforming the baseline methods on various repeated pattern discovery problems. These findings suggest a direction for developing repeated pattern discovery algorithms, and also evoke further investigation on music structure analysis, novelty analysis, and repeated pattern discovery algorithms.

[3] For more detailed definitions of the three evaluation measurements, refer to `https://www.music-ir.org/mirex/wiki/2017:Discovery_of_Repeated_Themes_\%26_Sections`.

[4] For the three baseline algorithms, we use the implementations available at `https://github.com/wsgan001/repeated_pattern_discovery`.

6. ACKNOWLEDGEMENT

The contribution of each author is as follows. Yo-Wei Hsiao performed data annotation and compiled the whole dataset. Tzu-Yun Hung performed data annotation and the data review process. Tsung-Ping Chen developed the motif discovery algorithm. Lastly, Li Su contributed in project supervision and paper writing. Also, the authors would like to thank Li-Rong Huang, Hsing-Chen Lin, Yu-Fang Hsu, Po-Hsuan Huang, Joseph-On-King Lau, Pei-Ling Kuo, and Chia-Han Li from the Department of Music, National Taiwan Normal University, for their efforts in reviewing and digitizing our data annotation.

7. REFERENCES

[1] D. Meredith, Ed., *Computational music analysis*. Springer Cham, 2016.

[2] P. Boot, A. Volk, and W. B. de Haas, "Evaluating the role of repeated patterns in folk song classification and compression," *Journal of New Music Research*, vol. 45, no. 3, pp. 223–238, 2016.

[3] C. Louboutin and D. Meredith, "Using general-purpose compression algorithms for music analysis," *Journal of New Music Research*, vol. 45, no. 1, pp. 1–16, 2016.

[4] Y.-J. Shih, S.-L. Wu, F. Zalkow, M. Müller, and Y.-H. Yang, "Theme transformer: Symbolic music generation with theme-conditioned transformer," *IEEE Transactions on Multimedia*, March 2022.

[5] Z. Hu, X. Ma, Y. Liu, G. Chen, and Y. Liu, "The beauty of repetition in machine composition scenarios," in *Proceedings of the 30th ACM International Conference on Multimedia*, Lisboa, Portugala, 2022, pp. 1223–1231.

[6] D. Meredith, "RECURSIA-RRT: Recursive translatable point-set pattern discovery with removal of redundant translators," in *International Workshops of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD) 2019*, Würzburg, Germany, 2019, pp. 485–493.

[7] O. Björklund, "Siatec-c: Computationally efficient repeated pattern discovery in polyphonic music," in *Proceedings of the 23rd International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022, pp. 59–66.

[8] T.-P. Chen and L. Su, "Functional harmony recognition of symbolic music data with multi-task recurrent neural networks," in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 90–97.

[9] T. Collins. 2013:Discovery of Repeated Themes & Sections. [Online]. Available: https://www.music-ir.org/mirex/wiki/2013:Discovery_of_Repeated_Themes_%26_Sections

[10] D. Conklin and C. Anagnostopoulou, "Representation and discovery of multiple viewpoint patterns," in *Proceedings of the 2001 International Computer Music Conference (ICMC)*, Havana, Cuba, 2001, pp. 479–485.

[11] A. Srinivasamurthy, S. Gulati, R. C. Repetto, and X. Serra, "Saraga: Open datasets for research on indian art music," *Empirical Musicology Review*, vol. 16, no. 1, pp. 85–98, December 2021.

[12] M. Krause, F. Zalkow, J. Zalkow, C. Weiß, and M. Müller, "Classifying leitmotifs in recordings of operas by Richard Wagner," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Montréal, Canada, 2020, pp. 473–480.

[13] P. van Kranenburg, B. Janssen, and A. Volk, "The meertens tune collections: The annotated corpus (mtc-ann) versions 1.1 and 2.0.1," *Meertens Online Reports*, vol. 2016, no. 1, 2016.

[14] C. Finkensiep, K. Déguernel, M. Neuwirth, and M. Rohrmeier, "Voice-leading schema recognition using rhythm and pitch features," in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR)*, Montreal, Canada, 2020, pp. 520–526.

[15] B. Janssen, W. B. De Haas, A. Volk, and P. Van Kranenburg, "Finding repeated patterns in music: State of knowledge, challenges, perspectives," in *Sound, Music, and Motion: 10th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Marseille, France, 2014, pp. 277–297.

[16] K. Frieler, F. Höger, M. Pfleiderer, and S. Dixon, "Two web applications for exploring melodic patterns in jazz solos," in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 777–783.

[17] D. Huron. Theme finder. [Online]. Available: http://www.themefinder.org/

[18] J.-L. Hsu, A. L. Chen, and C.-C. Liu, "Efficient repeating pattern finding in music databases," in *Proceedings of the 7th international conference on Information and knowledge management*, Maryland, USA, 1998, pp. 281–288.

[19] E. Cambouropoulos, M. Crochemore, C. Iliopoulos, L. Mouchard, and Y. Pinzon, "Algorithms for computing approximate repetitions in musical sequences," *International Journal of Computer Mathematics*, vol. 79, no. 11, pp. 1135–1148, 2002.

[20] D. Meredith, K. Lemström, and G. A. Wiggins, "Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music," *Journal of New Music Research*, vol. 31, no. 4, pp. 321–345, 2002.

[21] D. Meredith, "COSIATEC and SIATECCompress: Pattern discovery by geometric compression," in *Music Information Retrieval Evaluation eXchange (MIREX)*, Curitiba, Brazil, 2013.

[22] ——, "Point-set algorithms for pattern discovery and pattern matching in music," in *Dagstuhl Seminar Proceedings on Content-Based Retrieval*, Schloss Dagstuhl, Germany, 2006.

[23] G. Velarde, D. Meredith, and T. Weyde, "A wavelet-based approach to pattern discovery in melodies," D. Meredith, Ed. Springer Cham, 2016, pp. 303–333.

[24] T. Nuttall, G. Plaja, L. Pearson, and X. Serra, "The matrix profile for motif discovery in audio-an example application in carnatic music," in *Proceedings of the 15th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Online, 2021, pp. 109–118.

[25] M. Krause, F. Zalkow, J. Zalkow, C. Weiß, and M. Müller, "Classifying leitmotifs in recordings of operas by richard wagner," in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR)*, Montreal, Canada, 2020, pp. 473–480.

[26] I. Ren, A. Volk, W. Swierstra, and R. C. Veltkamp, "A computational evaluation of musical pattern discovery algorithms," *arXiv preprint arXiv:2010.12325*, 2020.

[27] C.-C. M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, H. A. Dau, D. F. Silva, A. Mueen, and E. Keogh, "Matrix profile i: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets," in *2016 IEEE 16th international conference on data mining (ICDM)*, Barcelona, Spain, 2016, pp. 1317–1322.

[28] D. Meredith, "Computing pitch names in tonal music: a comparative analysis of pitch spelling algorithms," Ph.D. dissertation, St. Anne's College, University of Oxford, 2007.