SIARCT-CFP: IMPROVING PRECISION AND THE DISCOVERY OF INEXACT MUSICAL PATTERNS IN POINT-SET REPRESENTATIONS

Tom Collins, Andreas Arzt, Sebastian Flossmann, and Gerhard Widmer
Department of Computational Perception, Johannes Kepler University Linz
{tom.collins, andreas.arzt, sebastian.flossmann, gerhard.widmer}@jku.at

ABSTRACT

The geometric approach to intra-opus pattern discovery (in which notes are represented as points in pitch-time space in order to discover repeated patterns within a piece of music) shows promise particularly for polyphonic music, but has attracted some criticism because: (1) the approach extends to a limited number of inexact repetition types only; (2) typically geometric pattern discovery algorithms have poor precision, returning many false positives. This paper describes and evaluates a solution to the *inexactness problem* where algorithms for pattern discovery and inexact pattern matching are integrated for the first time. Two complementary solutions are proposed and assessed for the *precision problem*, one involving categorisation (hence reduction) of output patterns, and the second involving a new algorithm that calculates the difference between consecutive point pairs, rather than all point pairs.

1. INTRODUCTION

The discovery of repeated patterns within a piece of music is an activity that manifests itself in a range of disciplines. In music psychology, for example, listeners' emotional responses to a piece exhibit distinctive behaviour at the beginning of repeated sections [11]. In music analysis, an awareness of the locations of motifs, themes, and sections, and their relation to one another, is a prerequisite for writing about the construction of a piece [3]. Last but not least, in music computing, algorithmic pattern discovery can be used to define compressed representations [13] (e.g., the numeric pitch sequence 67, 68, 67, 69, 69, 66, 67, 66, 68, 68 can be encoded as 67, 68, 67, 69, 69, and a translation operation "-1") and can act as a guide for the algorithmic generation of new music [9]. In the interests of supporting these multiple manifestations, it is important that the field of music information retrieval continues to develop and refine algorithms for the discovery of repeated patterns, and continues to evaluate these against each other and human-annotated ground truths.

There are two main representations in use for discovering repeated patterns within a piece of music (hereafter *intra-opus discovery* [8]): (1) *viewpoints* [9] involve encoding multiple aspects of the music as strings of symbols (such as the numeric pitches mentioned above, or durations, intervals between notes, etc.). This approach has been applied mainly to monophonic music; (2) the *geometric approach* [14] involves converting each note to a point in pitch-time space (see the pitch-time pairs in Figures 1A and B). Higher-dimensional spaces are also possible (e.g., including dimensions for duration or staff number). The geometric approach is well-suited to handling polyphonic music, where few attempts have been made to apply viewpoints. This paper focuses on the geometric approach; specifically, ontime and *morphetic pitch number* [14] (C4 = C♮4 = 60, D♭4 = D♮4 = D♯4 = 61, E♭4 = E4 = 62, etc.).

Before getting into more details of related work, it is helpful to distinguish the terms *pattern matching* and *pattern discovery*. Typically in pattern matching, there is a short musical query and a longer piece (or pieces) of music, and the aim is to match the query to more or less exact instances in the piece(s) [2, 17]. In intra-opus pattern discovery there is no query, just a single piece of music, and the requirement to discover motifs, themes, and sections that are repeated within the piece [8, 14]. (One could say that the purpose of a pattern discovery algorithm is to *create* analytically interesting but hitherto unknown queries.) Pattern discovery and pattern matching have been discussed in the same papers [13], but nobody to our knowledge has integrated discovery and *inexact* matching components in one algorithm before. This full integration is one of the contributions of the current work, and the other consists of two complementary methods for improving the precision of pattern discovery algorithms. The paper is organised around describing and evaluating components of a new algorithm called SIARCT-CFP, beginning at the end of the acronym with "FP" for fingerprinting, then "C" for categorisation, and finally SIARCT, which stands for Structure Induction Algorithm for r superdiagonals and Compactness Trawler, which has been defined before [5] and for which a Matlab implementation has been released. [1]

2. THE INEXACTNESS PROBLEM

In reviewing the Structure Induction Algorithm (SIA) and other geometric pattern discovery algorithms (see [14] or [7] for details), Lartillot and Toiviainen noted that "this ge-

[1] http://www.tomcollinsresearch.net

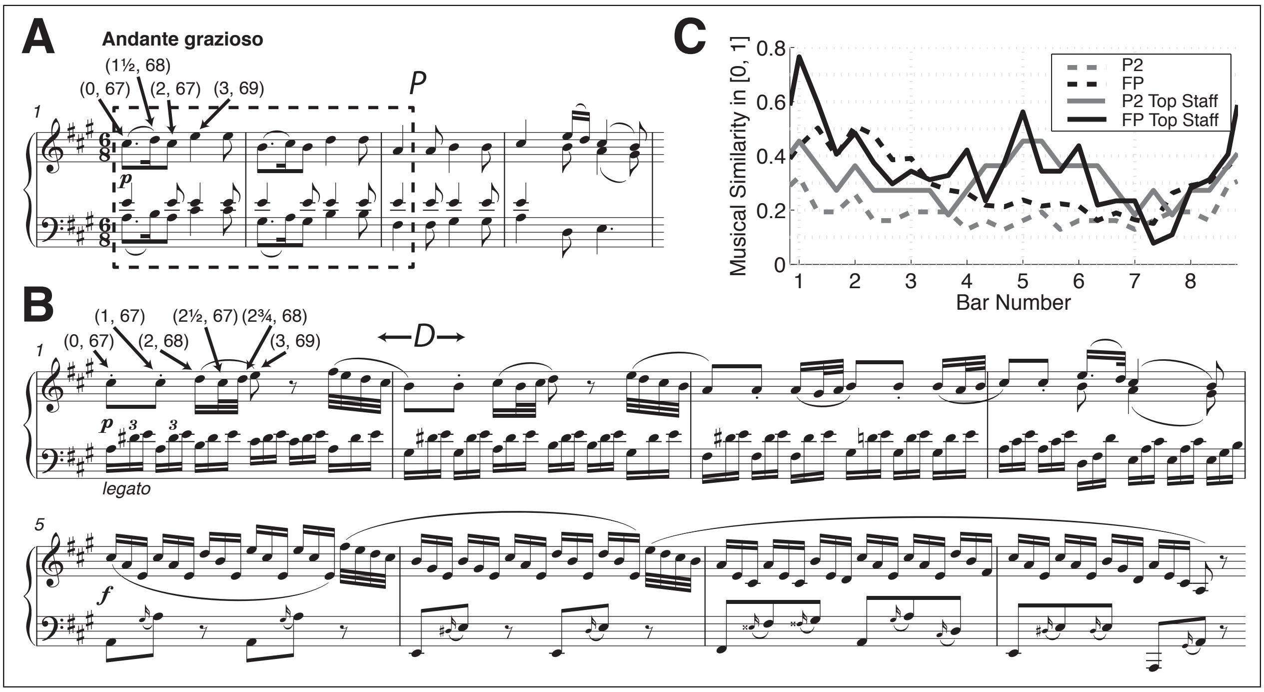

Figure 1. (A) Bars 1-4 of the Theme from the first movement of Piano Sonata no.11 in A major K331 by Wolfgang Amadeus Mozart (1756–1791). Labels give the ontime and morphetic pitch of the indicated note, and the box contains the top-rated pattern output by SIARCT; (B) Bars 1-8 of Variation II from the same movement; (C) Symbolic musical similarity of the pattern in (A) to the passage in (B), for two algorithms applied separately to the full texture and top staff only.

ometrical strategy did not apply to melodic repetitions that presented rhythmic variations" [10, pp. 290-291]. To illustrate this problem we use a theme by Mozart, from "one of the most overanalyzed pieces in the history of music theory" [15, p. 160]. We are not particularly interested in adding to discussions of the structure of the theme itself, rather in the relation of the theme to a subsequent variation. If the passage in Figure 1B were appended to the passage in Figure 1A and SIA applied to the single resulting point set, there would be little in the output to suggest that the first two bars of Figure 1B contain a variation on the bounded pattern P in Figure 1A. The points $\{(0, 67), (3, 69), (6, 66), (9, 68), (12, 65)\}$ would appear in the same output maximal translatable pattern (MTP, [14]), as they occur under the same translation in Figure 1B, but intervening points in the bounded pattern do not.

The pattern matching algorithm P2 [17] struggles with rhythmic variation also: for a given pattern P and a larger point set D, it returns all vector-frequency pairs $(\mathbf{w}, m)$ such that $m \geq 1$ points of P occur translated by $\mathbf{w}$ in D. We implemented P2 and used it to match P (from Figure 1A) to partial occurrences in D (Figure 1B). A summary of the output is plotted in Figure 1C, for both full-texture versions of P and D and a restriction to the right hand only (dashed and solid lines respectively). The maximal frequency M for pairs $(\mathbf{w}_1, m_1)_{i \in \{1,2,\ldots,s\}}$ corresponding to each crotchet-note ontime in D is plotted, normalised by the number of points in P, to give a measure of the symbolic musical similarity of P to D over time. While there are local maxima in the grey lines at bars 1, 2, and 5 (in the second case because P2 is transposition-invariant and there is a transposed pattern within P), in general they have a relatively small range, reflecting P2's struggle to distinguish genuine rhythmic variation from less related material.

Subsequent work on geometric pattern matching improves upon P2 in terms of capturing rhythmic variation, by representing durations as line segments [12, 17], by using the Hausdorff metric [16], or by converting to a tonal space representation [1]. A recent *fingerprinting* (FP) approach [2] has the advantage of not relying on durational information, and has options for transposition, time-shift, and scale-factor invariance, as well as tolerance for the amount by which the inter-onset interval of a pair of notes is permitted to differ, compared to a corresponding note pair in the original. The output of FP is a time series $S = S_t : t \in T$, where the set T of successive time points may or may not be uniformly spaced. The magnitude of S_t, called the *matching score*, indicates the extent to which an occurrence of the query begins at time t. In the transposition-invariant version, calculation of the matching score time series begins by creating fingerprint tokens

$$[y_j - y_i,\ x_j - x_i],\ \ t, \tag{1}$$

for locally constrained combinations of successive ontime-pitch pairs $(x_i, y_i), (x_j, y_j)$, in both a query pattern P and the larger point set D. The pair in brackets in (1) is the hash key, and $t = x_i$ is a time stamp. A scatter plot of the time stamps of matching hash keys for P and D can be used to identify regions of high similarity, which appear as approximately diagonal lines. The matching score is calculated by applying an affine transformation to the scatter plot and binning (for details, see [2, 18]).

An implementation of the FP algorithm was used to match exact/inexact occurrences of P from Figure 1A to D in Figure 1B, and the results are plotted in Figure 1C as black lines. It can be seen that FP outperforms P2 at distinguishing the rhythmic variation in bars 1-2 of Figure 1B. The use of locally constrained combinations of ontime-pitch pairs, rather than one candidate translation vector applied to all points in P, is what enables the FP algorithm to find a stronger match than P2.

Progress has been made in geometric pattern *matching* techniques, but Lartillot and Toiviainen's [10] criticism of the *discovery* approach still stands, as nobody to our knowledge has integrated an inexact matching technique within a pattern discovery approach. We do so now, according to the following steps, which define the "FP" part of SIARCT-CFP:

1. Let $P_1, P_2, \ldots, P_M$ be the output of a pattern discovery algorithm, each P_i having at least one translationally exact repetition (two occurrences) in D;

2. For $i = 1, 2, \ldots, M$, run the FP algorithm [2] on P_i and D, returning time points $t_1^{P_i}, t_2^{P_i}, \ldots, t_m^{P_i}$ at which there may be further exact/inexact occurrences of P_i, according to whether the value at $t_j^{P_i}$ is greater than some *similarity threshold* $c \in [0, 1)$.

Underlying this integration of pattern discovery and pattern matching is the following assumption, which we call the *translationally exact once* (TEO) hypothesis:

> If a piece of music contains multiple inexact occurrences of a perceptually salient or analytically interesting pattern, then for some majority subset of the pattern (i.e., a subset containing at least half of the points), there exists at least one translationally exact repetition (i.e., at least two occurrences).

If the discovery algorithm outputs such a majority subset, then the matching algorithm may be relied upon to output further exact/inexact occurrences of the pattern.

As a case study, the new algorithm SIARCT-CFP was run on the Nocturne in E major op.62 no.2 by Frédéric Chopin (1810–1849). [2] This is a sensible choice of piece, as it contains multiple variations of the opening theme (c.f. Figures 2B and D for instance). Fourteen patterns were output in total, one of which Q is bounded in Figure 2A, and occurs translated three times (bars 27–28, 58–59, and 60–61). These occurrences are rated as very similar to Q, with normalised matching scores close or equal to 1. The time series output by the FP has mean .264 and standard deviation .173, suggesting that the occurrence in Figure 2C is not distinguishable from other unrelated material. This makes sense, as although the contour and rhythm of the melody are as in Q, the pitch intervals are different (see arrows) and so is the accompaniment. We note, however, that

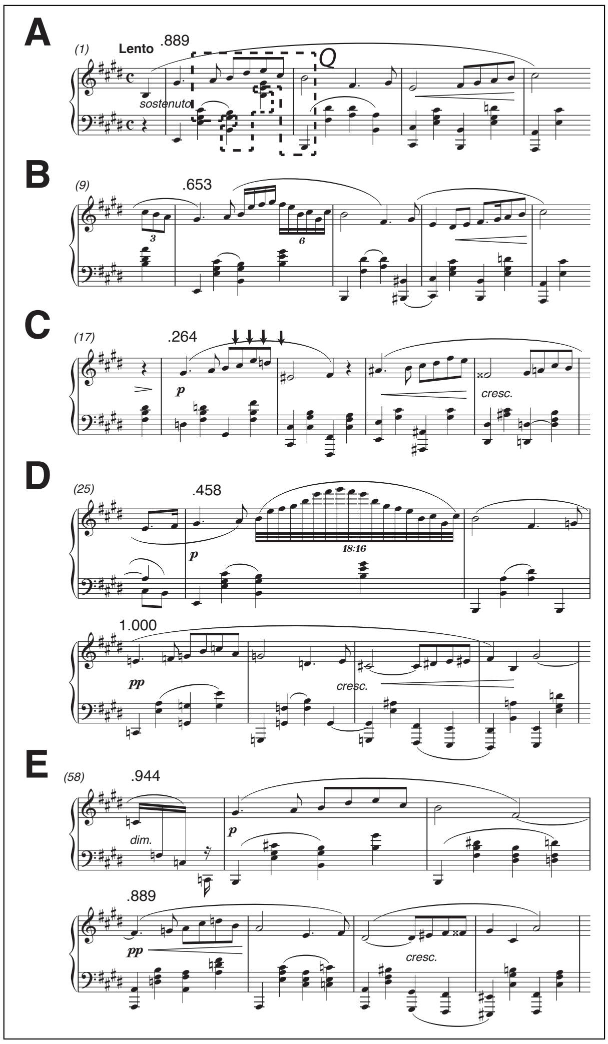

Figure 2. Excerpts from the Nocturne in E major op.62 no.2 by Chopin. Dashed lines in (A) bound a pattern Q discovered by SIARCT, which is used to match other inexact occurrences, with degree of exactness indicated in the figure by numbers in [0, 1]. Pedalling omitted for clarity.

the FP algorithm could be extended further to incorporate contour (up, down, same), as well as other viewpoints [9], because of its use of locally constrained comparisons.

3. THE PRECISION PROBLEM

3.1 Categorisation by Pattern Matching

Now that we have integrated some inexact pattern matching techniques into our pattern discovery approach, it is possible to employ them for the purposes of categorisation, based on the idea that P2 [17] or FP [2] can be used to compare two discovered patterns P_i and P_j in exactly the same way as if $P_i = P$ was a query and $P_j = D$ was a point set (or vice versa, as the measures are symmetric).

The second "C" in SIARCT-CFP stands for a **c**ategorisation process, which will be described now. The purpose of categorisation is to reduce an overwhelming amount of information (e.g., output patterns) to a more manageable number of exemplars. Here *categorisation* does not mean

[2] The first part of the algorithm, SIAR, ran with parameter $r = 1$. Second, the compactness trawler (CT) ran with compactness threshold $a = 4/5$, cardinality threshold 10, and lexicographic region type [7]. Third, the categorising and fingerprinting (CFP) ran with similarity threshold $c = 1/2$.

classifying patterns into an accepted/interesting category versus a rejected/uninteresting category; rather it means grouping similar patterns and representing each group with one exemplar pattern. Our motivation for categorising the output of SIARCT is to improve its precision: while the precision and recall of pattern discovery algorithms has been shown to benefit from compactness trawling, the precision is still quite poor [7]. For example, SIARCT outputs 76 patterns when run on Chopin's op.62 no.2, which can be reduced to fourteen patterns by using the following categorisation process:

1. Let $P_1, P_2, \ldots, P_M$ be the output of a pattern discovery algorithm, sorted descending by a rating of perceived pattern importance [6], or some other ordering. Let $J = \{1, 2, \ldots, M\}$ index the patterns that are uncategorised currently;
2. For the most important uncategorised pattern, index $i = \min(J)$, calculate the maximum normalised matching scores $s(P_i, P_j)$ for each $j \in J$, $j \neq i$;
3. For each similarity score $s(P_i, P_j)$ that is greater than some specifiable similarity threshold $c \in [0, 1)$, place pattern P_j in the category for which P_i is the exemplar, and remove j from J;
4. Repeat steps 2 and 3 until either J has one element k, in which case define P_k to be an exemplar with category membership P_k, or otherwise J is empty;
5. For the purposes of algorithm evaluation, return only the exemplars $P_{i(1)}, P_{i(2)}, \ldots, P_{i(m)}$.

Depending on the choice of c, $m \ll M$. The categorisation process can be visualised with two similarity matrices (Figure 3). The matrix in Figure 3A contains the maximum normalised matching scores for each pair of 76 output patterns for Chopin's op.62 no.2, ordered as in step 1 above. The matrix in Figure 3B is a permutation of 3A, showing the categorised patterns ($c = .5$) in their fourteen categories, bounded by white squares. The fourth square from top-left in Figure 3B represents the category for which Q in Figure 2A is the exemplar. The fivefold ($5.43 \approx 76/14$) reduction in output achieved by pattern-matching categorisation may well improve precision: as discussed, the theme annotated in Figure 2A survives the categorisation process, and so do all of the repetitions in this piece lasting four or more bars (results not shown). Pattern-matching categorisation also constitutes a novel and interesting use of the FP algorithm [2]. It should be noted that choosing too low a value for c could lead to over-reduction and filtering out of analytically interesting patterns. For instance, the first two squares in Figure 3B show considerable variegation, suggesting that some interesting subcategories may be overlooked.

3.2 Consecutive Points and Conjugate Patterns

The final novel contribution of this paper is to evaluate the SIARCT pattern discovery algorithm [5] against a collection of music containing repeated sections, and to compare its performance (especially precision) to SIA [14] and SIAR [5]. SIA outputs thousands of patterns for Chopin's op.62 no.2 (and other pieces of music [7]), so it is necessary to develop a more parsimonious pattern discovery algorithm for use as input to the categorisation and fingerprinting components described above (e.g., SIARCT outputs only 76 patterns for Chopin's op.62 no.2).

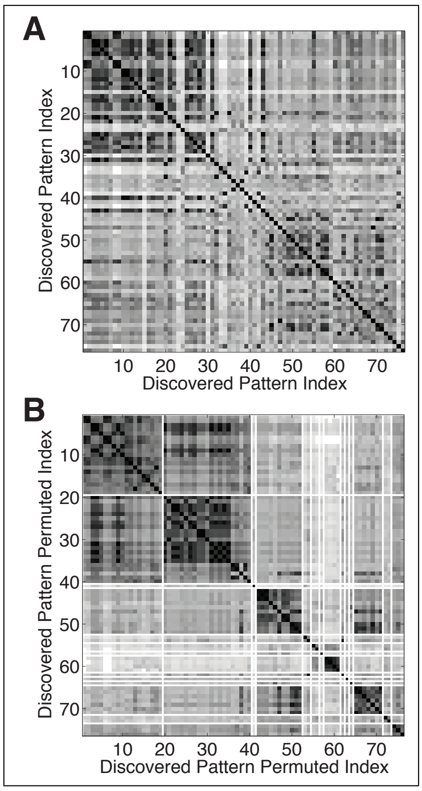

Figure 3. (A) Pairwise symbolic musical similarities (ranging from white for dissimilar to black for identical) for 76 patterns discovered by SIARCT in Chopin's op.62 no.2, ordered by a rating formula for perceived salience; (B) Permutation of the above matrix, with white lines indicating the results of categorising into fourteen groups.

It has long been thought that in order to discover repeated patterns within a geometric representation D of a piece, it is necessary to calculate the difference between each pair of n points ($n[n-1]/2$ calculations in total), as in SIA [14]. Unlike SIA, the first step of SIARCT is to calculate the difference between consecutive pairs of points only ($n - 1$ calculations). Some exhaustive pairwise comparisons are still made in the second step, but for small, non-overlapping subsets of D, meaning that the total number of difference calculations performed by SIARCT is far less than $n[n-1]/2$, in all but one degenerate case. [3] The third step of SIARCT makes use of a concept known as *conjugate patterns* [5]: if a pattern containing l points occurs m times in a point set, then there exists in the same point set a pattern consisting of m points that occurs l times. The fourth step calculates MTPs for each vector in a list L. As a consequence of manipulating conjugate patterns, the vectors corresponding to repeated sections should be at or near

[3] Please see [5] for the algorithmic details.

the top of L. So for this step we could: (1) distribute each MTP calculation to parallel processors, and/or; (2) output MTPs dynamically for the user to browse, whilst calculation of the remaining MTPs continues. The main claim is that SIARCT will have much smaller output than SIA, with minimal or no negative impact on its performance as measured by precision, recall, and robust versions of these metrics [4]. The compactness trawler (CT) part of SIARCT is exactly as in [7], so is not addressed again here.

SIA, SIAR, and SIARCT were run on point-set representations of movements by Ludwig van Beethoven (1770–1827) and Chopin listed in Figure 4A. SIARCT ran with compactness threshold $a = 1$, and points threshold $b = 50$. This means that only patterns containing 50 points or more were returned, and they had to have maximal compactness of 1. The parameter values make sense in terms of trying to discover repeated sections. To make the evaluation fair, we also filtered the results of SIA and SIAR, returning only those patterns that contained 50 points or more. In the results, these versions of SIA and SIAR are referred to as SIA (50+) and SIAR (50+).

3.3 Evaluation Results

Figure 4B shows the log of the total number of patterns output by each algorithm for each movement/piece. It supports the claim that SIAR has a much smaller output than SIA. It is difficult to see from Figure 4B, but the same observation applies to the filtered versions of each algorithm, SIAR (50+) and SIA (50+). The number of patterns output by SIARCT is several orders of magnitude less than that of any other algorithm. Figure 4C and Figure 4E show that compared with SIA's performance, SIAR is not negatively impacted by restricting calculations to consecutive pairs of points. The establishment precision and establishment recall for SIAR and SIA are comparable across all pieces.

Overall, the most effective algorithm is SIARCT (see Figure 4C and Figure 4E). For half of the pieces, it discovers all ground truth patterns exactly (Figure 4F). When SIARCT fails to discover a ground truth pattern exactly, often this is due to a difference between the repeated section as written in the score, and the repeated pattern as heard in a performance. For instance, in the fourth movement of Beethoven's op.7, bars 65–70 are marked as a repeated section, and this is included in the ground truth. The repeated notes extend beyond these bars in both directions, however, creating a longer repeated pattern in a performance. SIARCT discovers the latter, *performed* pattern, which reduces exact precision and recall. The more robust *establishment* metrics are not much reduced (e.g., see Figure 4E), and arguably discovering the performed pattern is preferable from a music-perceptual point of view.

4. DISCUSSION AND FUTURE WORK

This paper identifies two valid reasons why the geometric approach to intra-opus pattern discovery has attracted some criticism—namely (1) the approach extends to a limited number of inexact repetition types only, and (2) typically geometric pattern discovery algorithms are imprecise, returning many false positives results. A new algorithm called SIARCT-CFP has been described and evaluated component-wise, in an attempt to address these criticisms. It is the first geometric pattern discovery algorithm to fully integrate an inexact pattern matching component (the fingerprinting algorithm of [2]), and this matching component was shown to be effective for retrieving inexact occurrences of themes in pieces by Mozart and Chopin. The comparison of the FP algorithm [2] to a baseline pattern matching algorithm P2 [17] demonstrated that the former was superior for a particular example. In general it may be preferable to have two or more pattern matchers at one's disposal, however, as the number of variation techniques is large, and trying to account for them all with one algorithm will likely produce false positive matches.

The precision metrics were of particular interest to us in the comparative evaluation of SIARCT [5], SIAR [5], and SIA [14], as we claimed that SIARCT could achieve levels of precision comparable to SIA and SIAR, without harming recall. This claim was supported by the evaluation results, although in future work it will be necessary to see if similar results are achieved for ground truths containing shorter patterns than repeated sections.

Our *translationally exact once* (TEO) hypothesis (see Section 2) was borne out in the case study of Chopin's op.62 no.2, where Q (Figure 2A) occurred exactly under translation (bars 27–28, 58–59, and 60–61), and its contents were sufficient for use as a query to retrieve less exact versions such as in bars 9 (Figure 2B) and 25 (Figure 2D). For the case study of the Theme section and Variation II from Mozart's K331, SIARCT was able to discover perceptually salient patterns such as P in Figure 1A, which recurs in bars 5-7 of the Theme section (not shown). As the TEO hypothesis holds in both cases, future work should focus on finding counterexample pieces, as this will help to refine and improve our underlying assumptions and ensuing algorithms. Future work will also attempt to show users/developers the differences between themes and partial matches, and to identify variation techniques (triplets, minore, etc.) automatically.

5. ACKNOWLEDGEMENTS

This paper benefited from the use of Kern Scores, and helpful discussions with David Meredith. We would like to thank four anonymous reviewers for their comments. This work is supported by the Austrian Science Fund (FWF), grants Z159 and TRP 109.

6. REFERENCES

[1] T. Ahonen, K. Lemström, and S. Linkola: "Compression-based similarity measures in symbolic, polyphonic music," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 91–96, 2011.

[2] A. Arzt, S. Böck, and G. Widmer: "Fast identification of piece and score position via symbolic fingerprint-

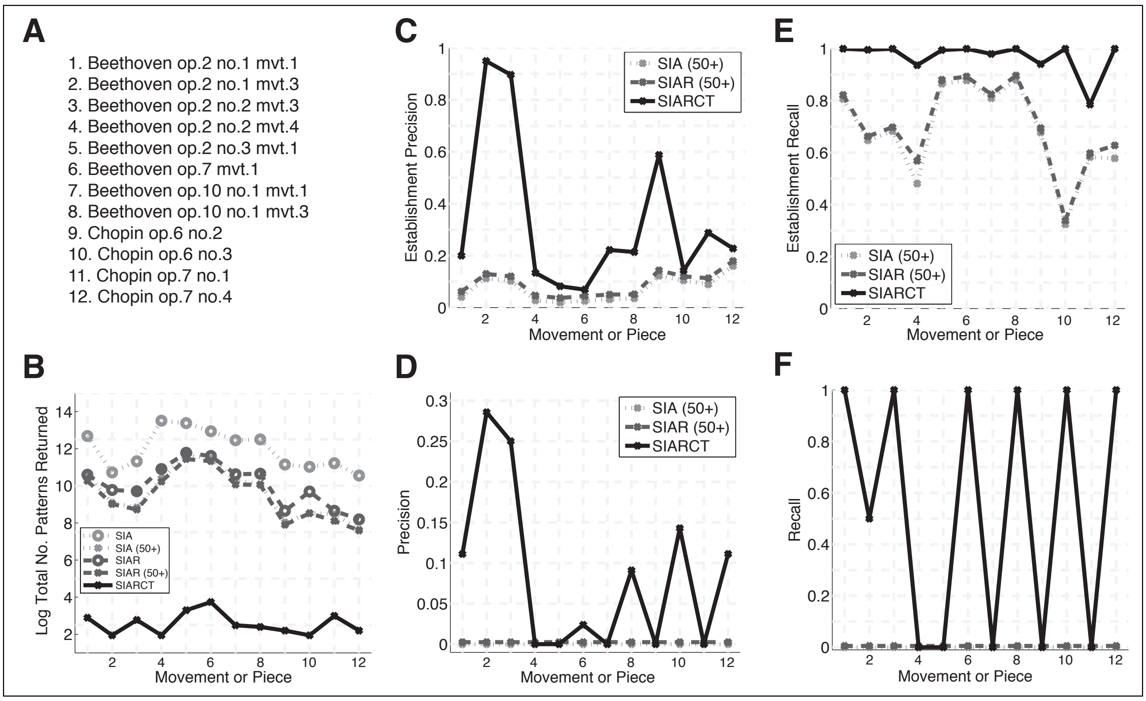

Figure 4. Evaluation metrics for three algorithms, run on eight movements by Beethoven and four pieces by Chopin.

ing," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 433–438, 2012.

[3] I. Bent and A. Pople: "Analysis," *The new Grove dictionary of music and musicians*, Macmillan, London, 2001.

[4] T. Collins: "Discovery of repeated themes and sections," Retrieved 4th May 2013, from http://www.music-ir.org/mirex/wiki/2013:Discovery_of_Repeated_Themes_%26_Sections

[5] T. Collins: *Improved methods for pattern discovery in music, with applications in automated stylistic composition*, PhD thesis, Faculty of Mathematics, Computing and Technology, The Open University, 2011.

[6] T. Collins, R. Laney, A. Willis, and P. Garthwaite: "Modeling pattern importance in Chopin's mazurkas," *Music Perception*, Vol. 28, No. 4, pp. 387–414, 2011.

[7] T. Collins, J. Thurlow, R. Laney, A. Willis, and P. Garthwaite: "A Comparative Evaluation of Algorithms for Discovering Translational Patterns in Baroque Keyboard Works," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 3–8, 2010.

[8] D. Conklin and C. Anagnostopoulou: "Representation and discovery of multiple viewpoint patterns," *Proceedings of the International Computer Music Conference*, pp. 479–485, 2001.

[9] D. Conklin and I. Witten: "Multiple viewpoint systems for music prediction," *Journal of New Music Research*, Vol. 24, No. 1, pp. 51-73, 1995.

[10] O. Lartillot and P. Toiviainen: "Motivic matching strategies for automated pattern extraction," *Musicae Scientiae*, Discussion Forum 4A, pp. 281–314, 2007.

[11] S. Livingstone, C. Palmer, and E. Schubert: "Emotional response to musical repetition," *Emotion*, Vol. 12, No. 3, pp. 552–567, 2012.

[12] A. Lubiw and L. Tanur: "Pattern matching in polyphonic music as a weighted geometric translation problem," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 154–161, 2004.

[13] D. Meredith: "Point-set algorithms for pattern discovery and pattern matching in music," *Proceedings of the Dagstuhl Seminar on Content-Based Retrieval*, 2006.

[14] D. Meredith, K. Lemström, and G. 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.

[15] E. Narmour: "Some major theoretical problems concerning the concept of hierarchy in the analysis of tonal music," *Music Perception*, Vol. 1, No. 1, pp. 129–199, 1983.

[16] C. Romming and E. Selfridge-Field: "Algorithms for polyphonic music retrieval: the Hausdorff metric and geometric hashing," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 457–462, 2007.

[17] E. Ukkonen, K. Lemström, and V. Mäkinen: "Geometric algorithms for transposition invariant content-based music retrieval," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 193–199, 2003.

[18] A. Wang: "An industrial strength audio search algorithm," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 7–13, 2003.