Mining Melodic Patterns in Large Audio Collections of Indian Art Music

Sankalp Gulati*, Joan Serrà†, Vignesh Ishwar* and Xavier Serra*
*Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
Email: sankalp.gulati@upf.edu, vignesh.ishwar@gmail.com, xavier.serra@upf.edu
†Artificial Intelligence Research Institute (IIIA-CSIC), Bellaterra, Barcelona, Spain
Email: jserra@iiia.csic.es

***Abstract*—Discovery of repeating structures in music is fundamental to its analysis, understanding and interpretation. We present a data-driven approach for the discovery of short-time melodic patterns in large collections of Indian art music. The approach first discovers melodic patterns within an audio recording and subsequently searches for their repetitions in the entire music collection. We compute similarity between melodic patterns using dynamic time warping (DTW). Furthermore, we investigate four different variants of the DTW cost function for rank refinement of the obtained results. The music collection used in this study comprises 1,764 audio recordings with a total duration of 365 hours. Over 13 trillion DTW distance computations are done for the entire dataset. Due to the computational complexity of the task, different lower bounding and early abandoning techniques are applied during DTW distance computation. An evaluation based on expert feedback on a subset of the dataset shows that the discovered melodic patterns are musically relevant. Several musically interesting relationships are discovered, yielding further scope for establishing novel similarity measures based on melodic patterns. The discovered melodic patterns can further be used in challenging computational tasks such as automatic rāga recognition, composition identification and music recommendation**

***Keywords*-Motifs, Pattern discovery, Time series, Melodic analysis, Indian art music**

I. Introduction

Audio music is one of the fastest growing multimedia content in modern days. We need intelligent computational tools to organize big audio music repositories in a way that can enable meaningful navigation, efficient search and discovery, and recommendation. This necessitates establishing relationships between audio recordings based on different types of data, such as the editorial metadata, the audio content and the surrounding context [1]. In this article we focus on music content analysis through the discovery of short-time melodic patterns.

Repeating structures (or patterns) are important information units in data such as text, DNA sequences, images, videos, speech and music [2], [3]. Patterns are exploited in a variety of ways, ranging from signal level tasks such as data-compression [4] to more cognitively complex tasks such as analyzing an art work [5]. In the music domain, the identification of repeating structures in a musical piece is fundamental to its analysis, understanding and interpretation [6], [7].

In music information research (MIR), several approaches have been proposed for analyzing different kinds of repeating structures, including long duration repetitions such as themes, choruses and sections [8], and short duration repetitions such as motifs and riffs [9]. While there exists a number of approaches for motivic discovery in sheet music [10], there are fewer approaches that work on audio music recordings [11]. This can be attributed to the audio-symbolic gap [12], which can be bridged by a reliable automatic transcription system to abstract the audio music content into musically meaningful discrete symbols. There exists a wide scope for developing methodologies for the discovery and analysis of short duration melodic patterns (or motifs) in large audio music collections. In this paper, we address this task for Carnatic music.

Carnatic music is one of the two Indian art music (IAM) traditions with over millions of listeners around the world. Melodies in this music tradition are complex and are based on an intricate melodic framework, the rāga, which has evolved through centuries [13]. Rāgas are largely characterized by their constituent melodic patterns, and hence, discovering melodic patterns is a key to meaningful information retrieval in Carnatic music [14]. When compared to other genres from western popular music, fewer musical instruments and the prominence of melody in IAM reduces the complexity of some signal processing steps such as pitch estimation[1]. However, the main challenges arise primarily due to nuances in the sophisticated rāga framework. Moreover, the improvisatory nature of music results in a higher degree of variability across repetitions of a melodic pattern. These challenges make IAM a unique and difficult repertoire with which to develop computational approaches for melodic pattern discovery from raw audio recordings.

In recent years, many approaches have been proposed for this task, most of them of a supervised nature. Ross et al. [15] detect title phrases of a composition within a concert of Hindustani music. The authors use annotated rhythm cycle boundaries for pattern segmentation.

[1]Compare results across datasets: http://www.music-ir.org/mirex/wiki/2013:MIREX2013_Results

Ishwar et al. [16] propose a two-stage approach and a sparse melody representation for spotting characteristic melodic patterns of a rāga. Rao et al. [14] classify melodic motives in IAM by using exemplar-based matching and propose an approach to learn DTW global constraints for computing melodic similarity. Many of these approaches either use semi-supervised pitch estimation, manually segmented pattern boundaries, a dataset comprising few recordings, or analyze only a limited number of characteristic phrases. Thus, scalability of such approaches is questionable and over-fitting of the approach to a specific dataset is probable.

Computational motivic analysis can yield interesting musical results through a data-driven, unsupervised methodology. This is largely explored in the case of western sheet music. Janssen et al. [9] present an overview and categorization of these approaches based on a taxonomy. Those approaches address various challenges such as melody representation, melody segmentation, melodic similarity and pattern redundancy reduction [10], [17], [18]. In the case of audio music recordings, approaches for motif discovery can benefit from the literature in the domain of time series analysis such as time series representation [19], core pattern discovery methods [20], and search and indexing techniques [21].

In this paper, we present a data-driven unsupervised approach for melodic pattern discovery in large collections of music recordings containing hundreds of millions of pattern candidates. Over 13 trillion distance computations are done in this task. To the best of our knowledge, this is the first time melodic patterns are mined from such a large volume of audio data. We propose a quantitative methodology for parameter selection during the data pre-processing step. In addition, we evaluate four different variants of the DTW cost function for computing melodic similarity. Our approach is robust to different tonic pitches of the lead artist, non-linear timing variations, global tempo changes and added melodic ornaments. As a result, we also discovered several non-intuitive melodic patterns that surprised a professional musician with over 20 years of experience. To facilitate the reproducibility of our work, and in order to incrementally build new tools for the melodic analysis of massive collections, the code and the data used in this study are made available online[2].

II. Method

Our proposed approach consists of four main blocks (Fig. 1). The data processing block (Sec. II-A) generates pitch subsequences from every audio recording in the music collection. The intra-recording pattern discovery block (Sec. II-B) performs an exact pattern discovery by detecting the closest subsequence pairs within an audio recording (referred to as seed patterns). The inter-recording pattern detection block (Sec. II-C) considers each seed pattern as a query and searches for its occurrences in the entire music collection. The rank refinement block (Sec. II-D) reorders a ranked list of search results by recomputing melodic similarity using a more sophisticated similarity measure.

[2]http://compmusic.upf.edu/node/210

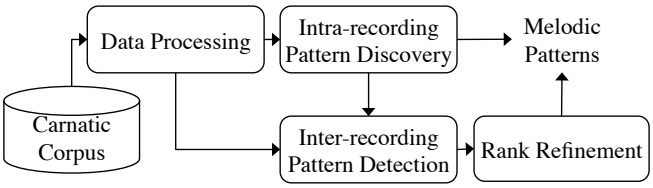

Figure 1. Block diagram of the proposed approach.

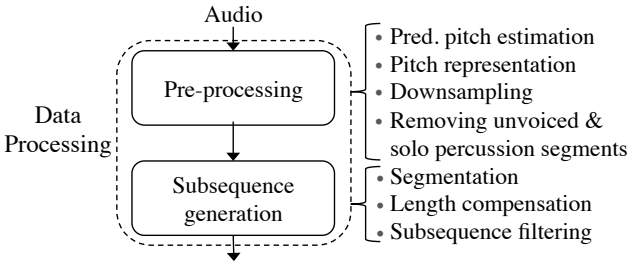

Figure 2. Block diagram of the data processing module.

We choose to perform first an intra-recording pattern discovery because several melodic patterns are repeated within a music piece of Carnatic music. Moreover, the scalability of the computational approaches considered here for discovering patterns at the level of the entire music collection is questionable. To confirm this hypothesis, we conducted an experiment using a state of the art algorithm for time series motif discovery [20], with a trivial modification to extract the top K motifs. Using just 16 hours of audio data, the algorithm could discover only 40 patterns in 24 hours using Euclidean distance. Besides pattern pairs being from the same recording, only a few of the obtained pattern pairs were melodically similar. This brought up the need for a similarity measure that was robust to non-linear timing variations. Scaling these algorithms to over hundreds of hours of audio data and using computationally expensive distance measures is nowadays a challenge.

A. Data Processing

1) Pre-processing: The steps involved in the pre-processing block are shown in Fig. 2. A brief description of each of these steps is given below:

a) Predominant Pitch Estimation: We consider melody as the predominant pitch in the audio signal and estimate it using the method proposed by Salamon and Gómez [22]. This method performed very favorably in an international MIR evaluation campaign focusing on a variety of music

genres, including IAM[3]. We use the implementation available in Essentia 2.0 [23], an open-source C++ library for audio analysis and content-based MIR. We use a frame size of 46 ms and a hop size of 4.44 ms. All other parameters are left to their default values. Before pitch estimation, we apply an equal-loudness filter using the default set of parameters. Noticeably, the predominant pitch estimation algorithm also performs voicing detection, which is used in the later part of our data processing methodology to filter unvoiced segments (Fig. 2).

b) Pitch Representation: For the pitch representation to be musically relevant, the pitch values are converted from Hertz to Cents (logarithmic scale). For this conversion we additionally consider the tonic pitch of the lead artist as the reference frequency (i.e., 0 Cent corresponds to the tonic pitch). Thus, our representation becomes independent of the tonic of the lead artist, which allows a meaningful comparison of melodies of two distinct recordings (even if sung by two different artists in different tonic pitches). The tonic of the lead artist is identified automatically using a classification-based multi-pitch approach [24]. We use the implementation of this method available in Essentia with the default set of parameters.

c) Downsampling: In order to reduce the computational cost, we downsample the predominant pitch sequence (Fig. 2). We derive the new sampling rate using the autocorrelation (ACR) of short-time pitch segments generated using a sliding window of 2 s. We compute the ACR of all possible pitch segments in the entire dataset for different lags l, $l \in \{0, 1, \ldots 30\}$, and examine the histogram of normalized ACR values at each lag (Fig. 3). We select the lag at which the third quartile Q3 has an ACR value of 0.8, which corresponds to a sampling rate of 22.22 ms. We informally found that this sampling rate generally preserves melodic nuances and rapid pitch movements while reducing the computational requirements of the task. In the literature, we could not find any reference for this sampling rate of the melody. Thus, our quantitative derivation could be useful for further studies.

d) Solo Percussion Removal: A concert of Carnatic music typically contains a solo percussion section, referred as *Tani Avartana* or *Tani* in short. Its duration typically varies from 2 to 25 min. Since the main percussion instrument in Carnatic music, the *mṛdaṅgaṁ*, has tonal characteristics, the pitch estimation algorithm tracks the pitch of the mṛdaṅgaṁ strokes instead of detecting this section as an unvoiced segment. Hence, we dedicate an extra effort to discard such segments using a classification-based approach (Fig. 2). To feed the classifiers we extracted 13 MFCC coefficients, spectral centroid, spectral flatness and pitch salience (c.f. [25]) from the audio signal using Essentia. We iterated over 23, 46 and 92 ms frame sizes and chose the one which resulted in a better classification accuracy. We set the hop size as half the frame size and all other parameters to their default values. Next, we computed means and variances of these features over 2 s non-overlapping segments. For training, we used a labeled audio music dataset containing 1.5 hours of mixed voice and violin recordings and 1.5 hours of solo percussion recordings. To assess the performance of the extracted features, we performed a leave-one-out cross-validation. We experimented with five different algorithms exploiting diverse classification strategies [26]: decision trees (Tree), K nearest neighbors (KNN), naive Bayes (NB), logistic regression (LR), and support vector machines with a radial basis function kernel (SVM). We used the implementations available in scikit-learn version 0.14.1 [27]. We used the default set of parameters with few exceptions to avoid over-fitting and to compensate for the uneven number of instances per class. We set `min_ samples_split=10` for Tree, `fit_prior=False` for NB, `n_neighbors=5` for KNN, and for LR and SVM `class_weight='auto'`. The combination of the frame size of 46 ms and the SVM classifier yielded the best performance (96% accuracy), with no statistically significant difference to the performance with the Tree (95.5%) and the KNN (95%), for the same frame size. We finally chose KNN because of its low complexity.

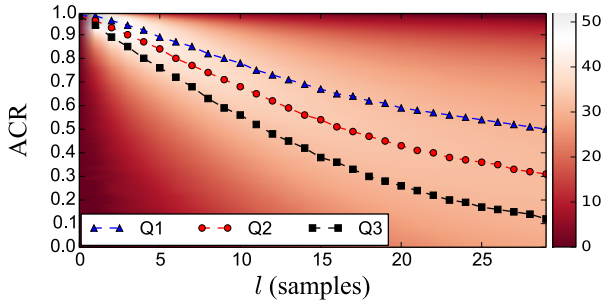

Figure 3. Histograms of ACR values (histogram value is indicated by the colormap on the right; for ease of visualization, we compress the range of the histogram values by taking its fourth root). Q1, Q2 and Q3 denote the three quartile boundaries of the histogram.

2) Subsequence Generation: The steps involved in generating candidate subsequences are as follows:

a) Segmentation: Due to the lack of reliable methods for segmentation of melodic patterns in IAM [28], we generate pitch subsequences by using a sliding window of length W_l with a hop size of one sample (22 ms). Given no quantitative studies investigating the length of the melodic patterns in Carnatic music, we make a choice of $W_l = 2$ s based on recommendations from a few Carnatic musicians. Since unvoiced segments are removed from the pitch sequence at the pre-processing step, a window can include pitch samples separated by more than W_l seconds. To handle these cases, we use the time stamps of the first

[3]http://nema.lis.illinois.edu/nema_out/mirex2011/results/ame/indian08/summary.html

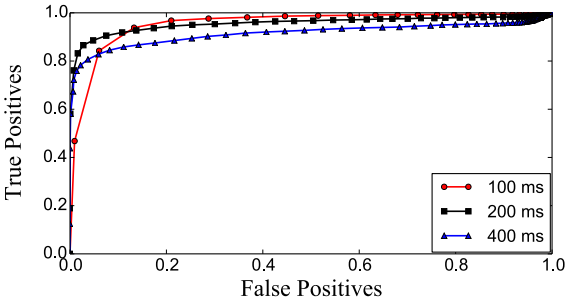

Figure 4. ROC curves for 'flat' and 'non-flat' region classification for different values of window length (W_{std}) used for selecting an optimal value of standard deviation S_i.

sample (T_1) and the last sample (T_2) in a window. We filter out all subsequences for which $T_2 - T_1 > W_l + \Phi$. We select $\Phi = 0.5\,\text{s}$ to account for the short pauses during a phrase rendition. This value was empirically set to differentiate between inter- and intra-phrase pauses.

b) Subsequence Filtering: A subsequence may contain a segment of the pitch contour corresponding to a single musical note, where the pitch values are nearly constant. Such musically uninteresting patterns are discarded in a filtering stage (Fig. 2). The criterion for discarding such subsequences is summarized below:

$$\beta = \sum_{i=0}^{W_n} \Theta\left(S_i \geq T_{\text{std}}\right),$$

where β is the flatness measure of a subsequence, W_n denotes its number of samples, $\Theta(z)$ is a Heaviside step function yielding $\Theta(\text{true}) = 1$ and $\Theta(\text{false}) = 0$, and S_i is the standard deviation at the i-th sample of a subsequence, computed using a window of length W_{std} centered at sample i. In order to determine the optimal values of W_{std} and T_{std}, we manually labeled a number of regions in pitch contour as 'flat' and 'non-flat' for 4 excerpts in our database. We iterated over different parameter values and analyzed the resultant ROC curve (Fig. 4). Doing so, we found that $W_{\text{std}} = 200\,\text{ms}$ resulted in the best performance and that the knee of the curve corresponded to $T_{\text{std}} = 45\,\text{Cents}$. Having a value of β for each subsequence, we finally filter out the ones for which $\beta \leq \gamma W_n$, using $\gamma = 0.8$. The latter was set by visual inspection.

After the data processing step, we retain around 17.5 million pattern candidates for our entire dataset. If no subsequence filtering is applied, a sampling rate of 225 Hz for pitch sequence amounts to nearly 300 million pattern candidates for a database as big as ours.

B. Intra-recording Pattern Discovery

We perform an exact pattern discovery by computing the similarity between every possible subsequence pair obtained within an audio recording. We regard the top $N = 25$ closest subsequence pairs in each recording as seed patterns. We omit overlapping subsequences in order to avoid trivial matches and additionally constrain the top N seed pattern pairs to be mutually non-overlapping. Due to this constraint for some recordings we obtain less than 25 pattern pairs. In total, for all the recordings, nearly 1.4 trillion DTW distance computations are done to obtain 79,172 seed patterns.

1) Melodic Similarity: We compute melodic similarity between two subsequences using a DTW-based distance measure [29]. We use a step condition of $\{(1,0),(1,1),(0,1)\}$ and the squared Euclidean distance as the cost function. We do not use any penalty for insertion and deletion. These choices are made in order to allow lower bounding (see below). In addition, we apply the Sakoe-Chiba global constraint with the band width set to 10% of the pattern length. This constraint may be sufficiently large for accounting time warpings in melodic repetitions in Carnatic music.

2) Lower Bounding DTW: To make DTW distance computations tractable for such a large number of subsequences we apply cascaded lower bounds [21]. In particular, we use FL (first-last) lower bound and LB_Keogh bound for both query to reference and reference to query matching. Besides, we apply early abandoning, both during the computation of lower bounds as well as during the DTW distance computation [21].

3) Pattern Length Compensation: Along with the local non-linear time warpings, the overall length of a melodic pattern may also vary across repetitions. For example, a melodic pattern of length 2 s might be sung in 2.2 s in a different position in the song. We handle this by using multiple time scaled versions of a subsequence in the distance computation. This technique is also referred to as local DTW and is shown to have tighter lower bounds [30]. It should be noted that typically such issues are addressed by using a subsequence variant of the DTW distance measure. However, the lower bounding techniques we used during the DTW distance computation do not work for the subsequence variant of the DTW.

For every subsequence, we generate five subsequences by uniformly time scaling it by a factor of $\alpha \in I_{\text{intp}} = \{0.9, 0.95, 1, 1.05, 1.1\}$, such that the length of the resulting subsequences is W_l. We use cubic interpolation for uniformly time scaling a subsequence. Since these 5 interpolation factors increase the computational cost by a factor of 25, we assume that the distance between a subsequence pair $X_{1.0}$ and $Y_{1.05}$ is very close to the distance between the pair $X_{1.05}$ and $Y_{1.1}$ (the sub-index denotes the interpolation factor α). Following this rationale, we can avoid the distance computation between 16 of the 25 combinations without a significant compromise on accuracy.

C. Inter-recording Pattern Detection

We consider every seed pattern as a query (79,172 in number) and perform an exhaustive search over all the subsequences obtained from the entire audio music collection (nearly 17.5 million in number). For every seed pattern we store top $M = 200$ closest matches (referred to as search patterns). Here also for every subsequence we consider 5 uniformly scaled subsequences in the distance computation. For inter-recording pattern detection also use the same similarity measure and lower bounding techniques as used in intra-recording pattern discovery block (Sec. II-B). In total, nearly 12.4 trillion DTW distance computations are done in this step.

D. Rank Refinement

The lower bounds we use for speeding up distance computations are not valid for any variant of DTW. However, once the top matches are found, nothing prevents us from reordering the ranked list using any variant of DTW, as we do not need to apply lower bounds in this reduced search space. In this step, we select a DTW step condition of $\{(1,2),(1,1),(2,1)\}$ to avoid some pathological warpings of the path. Furthermore, we investigate four different distance measures d_i, $i = 1,\ldots 4$, used in the computation of the DTW cost matrix as described below.

$$d_1 = \delta \;; \quad d_3 = \begin{cases} \delta - 25, & \text{if } \delta > 25 \\ 0, & \text{otherwise} \end{cases}$$
$$d_2 = \delta^2 \;; \quad d_4 = \begin{cases} (\delta - \phi)^{1.5} + \varphi, & \text{if } \delta > 100 \\ d_3, & \text{otherwise} \end{cases} \tag{1}$$

where $\delta = |p_1 - p_2|$ is the city block distance between two pitch values and all numeric values are in Cents. We set $\phi = 99.555$ and $\varphi = 74.70$ to maintain point and slope continuity. The formulation for the different d_i is inspired by our own experience and some of the approaches we find in the literature [14], [16]. We denote the four variants of the rank refinement method by V_i, $i = 1\ldots 4$.

III. Evaluation

A. Music Collection

The data used in this article comprises 365 hours of music, containing 1,764 audio recordings covering diverse forms in Carnatic music. This dataset is a subset of the carefully compiled Carnatic music corpus of the CompMusic project [31], [32]. The selected musical material is diverse in terms of number and gender of lead artists, number of rāgas, year of release and various forms within Carnatic music.

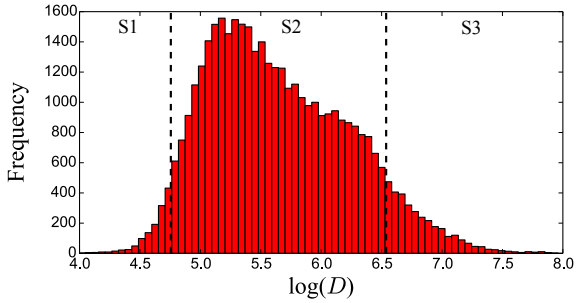

Figure 5. Distance distribution of seed patterns. Three seed pattern categories are marked by S1, S2 and S3.

B. Evaluation Methodology

One of the challenges in any data-driven task is evaluation. We here perform a quantitative evaluation based on expert feedback. For the entire dataset we obtain over 15 million search patterns for each of the rank refinement methods. We divide seed patterns into three categories based on the distance between the seed pairs, which we denote by D. Then, to have an equal representation from the range of values of D, 200 seed pairs equally distributed among these categories are randomly selected for evaluation (Fig. 5). Seed category boundaries are $\mu \pm 1.5\sigma$, where μ and σ are the mean and the standard deviation of the distribution of D. For every selected seed pattern we consider the first 10 search patterns for each of the four rank refinement methods. Thus, in total, we obtain 200 seed pairs and 8,000 search patterns for expert evaluation.

Expert evaluation is performed by a professional Carnatic musician who has received over 20 years of music education. For examining similarity between two melodic patterns, the musician listened to the audio fragments corresponding to these patterns and scored a 0 for melodically dissimilar and a 1 for melodically similar. The musician annotated melodic similarity for each seed pair and between the seed and its search patterns for every rank refinement method.

To quantify the musician's assessment of the similarity between the melodic patterns we use mean average precision (MAP), a typical evaluation measure in information retrieval [33], which is also very common in MIR. This way, we have a single number to evaluate the performance of the four different rank refinement methods. For the computation of the MAP scores we consider the total number of relevant patterns as the number of relevant patterns retrieved in the top 10 search results. For assessing statistical significance we use the Mann-Whitney U test [34] with $p < 0.05$. To compensate for multiple comparisons, we apply the Holm-Bonferroni method [35]. Thus, eventually we use a much more stringent criteria than $p < 0.05$ for measuring statistical significance. We use ROC curves to analyze the separation between the distance distribution of melodically similar and dissimilar subsequences [33].

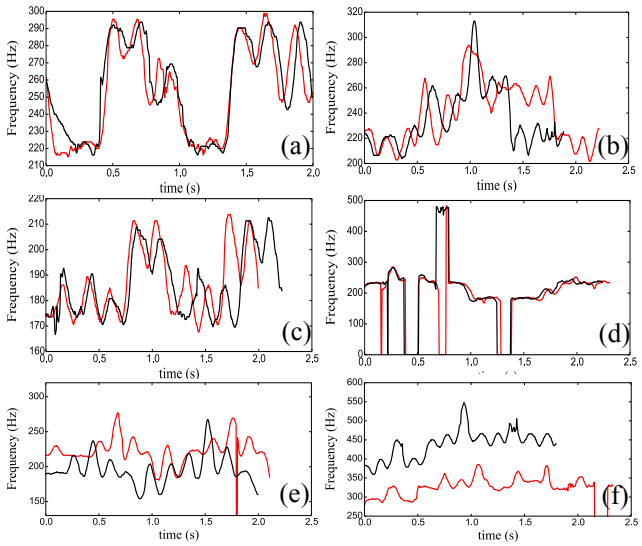

Figure 6. Examples of the discovered melodic patterns.

Table I
PERCENTAGE OF EXITS AFTER A LOWER BOUND COMPUTATION WITH RESPECT TO THE TOTAL NUMBER OF DISTANCE COMPUTATIONS.

Lower bound	Intra-rec.(%)	Inter-rec.(%)
LB_KIM_FL	52	45
LB_Keogh_EQ	23	51
LB_Keogh_EC	1	3

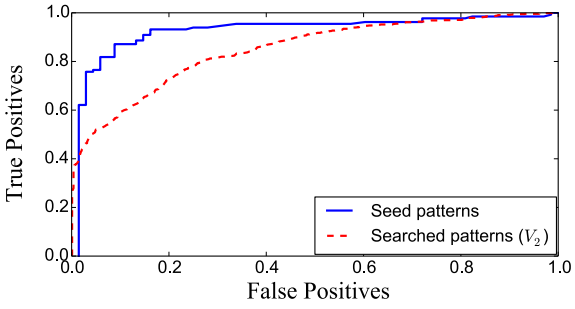

Figure 7. ROC curve for seed pairs and search patterns (using V_2) in the evaluation set.

IV. Results and Discussion

Before presenting formal evaluations, we show a few examples of the discovered melodic patterns in Fig. 6. Our approach robustly extracts patterns in different scenarios such as large local time warpings (b), uniform scaling (c), patterns with silence regions (d) and across different tonic pitches (e and f). It is worth mentioning that, during the process of annotation, the musician found several musically interesting results. For example, striking similarity between phrases of two different rāgas, between phrases in sung melodies and the melodies played on instruments (Violin or Vīṇa), and phrases sung by different artists. Many of the discovered patterns are the characteristic melodic phrases of the rāga, which are the primary cues for rāga recognition. Overall, the obtained results are musically relevant and can be used to establish meaningful relationships between audio recordings.

It is also interesting to analyze the contribution of different lower bounds in pruning the search space. In Table I we show in percentage the number of times the program counter exits after a lower bound computation with respect to the total number of distance computations. As mentioned before, the total number of distance computations are 1.413 trillion for intra-recording pattern discovery and 12.418 trillion for inter-recording pattern detection. From Table I it becomes evident that the lower bounding methods are more effective in inter-recording pattern detection. This is expected as different songs may correspond to different rāgas and hence use different set of musical notes.

We now proceed to formal evaluations. We first evaluate the performance of the intra-recording pattern discovery task. We find that the fraction of melodically similar seed pairs within each seed category S1, S2 and S3 consistently decreases: 0.98, 0.67 and 0.31, respectively. To further examine the separation between melodically similar and dissimilar seed pairs, we compute the ROC curve (Fig. 7, solid blue line). The knee of such curve corresponds to a precision of approximately 80% for 10% of false positive cases. This indicates that the chosen DTW-based distance measure is a sufficiently good candidate for computing melodic similarity for the case of intra-recording seed pattern discovery.

Next, we evaluate the performance of inter-recording pattern detection task and assess the effect of the four DTW cost variants of Sec II-D (denoted by $V_1 \ldots V_4$). To investigate the dependence of the performance on the category of the seed pair, we perform the evaluation within each seed category (Table II). In addition, we also present a box plot of corresponding average precision values (Fig. 8). In general, we observe that every method performs well for category S1, with a MAP score around 0.9 and no statistically significant difference between each other. For category S2, V_2 and V_3 perform better than the rest and the difference is found to be statistically significant. The performance is poor for the third category S3 for every variant. The difference in performance between any two methods across seed categories is statistically significant. We observe that MAP scores across different seed categories correlate well with the fraction of melodically similar seed pairs in that category (discussed above). This suggests that patterns which find good matches within a recording (i.e., low distance D) also correlate with more repetitions across recordings.

Finally, we analyze the distance distribution of search patterns for the best performing method V_2 (Fig. 7, dashed red line). We observe that the separability between melod-

Table II
MAP SCORES FOR FOUR VARIANTS OF RANK REFINEMENT METHOD (V_i) FOR EACH SEED CATEGORY (S1, S2 AND S3).

Seed Category	V_1	V_2	V_3	V_4
S1	0.92	0.92	0.91	0.89
S2	0.68	0.73	0.73	0.66
S3	0.35	0.34	0.35	0.35

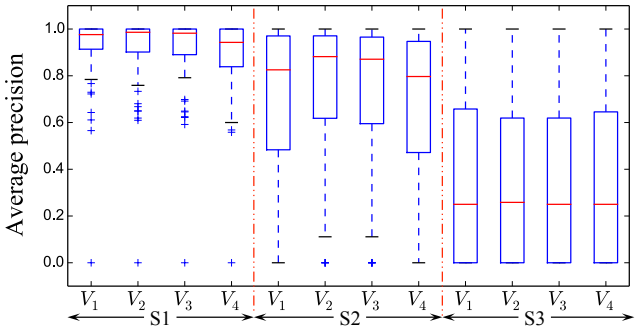

Figure 8. Boxplot of average precision for variants of rank refinement method (V_i) for each seed category.

ically similar and dissimilar subsequences in this case is poorer than the one obtained for the seed pairs (solid blue line). This indicates that it is much harder to differentiate melodically similar from dissimilar patterns when the search is performed across recordings. This can be attributed to the fact that phrases of two allied rāgas are differentiated based on subtle melodic nuances [13]. Hence, one faces a much more difficult task.

V. Conclusion and Future Work

We presented a data-driven unsupervised approach for melodic pattern discovery in large audio collections of Indian art music. A randomly sampled subset of the extracted melodic patterns was evaluated by a professional Carnatic musician. We first discovered seed patterns within a recording and later used those as queries to detect similar occurrences in the entire dataset. We used DTW-based distance measures to compute melodic similarity and compared four different rank refinement methods. We showed that a variant of DTW using cityblock distance performs slightly better than the rest. We also found that a DTW-based distance measure performs reasonably well for intra-recording discovery. However, we require better melodic similarity measures for searching occurrences across recordings. This is a clear direction for future works. Our results also indicate that patterns which find close matches within a recording have a larger number of repetitions across recordings. As mentioned before, the data and the code used in this study are available online.

Future work includes the improvement of the melodic similarity measure, finding musically meaningful pattern boundaries and making melodic similarity invariant to transpositions across octaves. We also plan to perform a similar analysis in an Hindustani audio music collection.

VI. Acknowledgments

This work is partly supported by the European Research Council under the European Union's Seventh Framework Program, as part of the CompMusic project (ERC grant agreement 267583). J.S. acknowledges 2009-SGR-1434 from Generalitat de Catalunya, ICT-2011-8-318770 from the European Commission, JAEDOC069/2010 from CSIC, and European Social Funds.

References

[1] A. Porter, M. Sordo, and X. Serra, "Dunya: A system for browsing audio music collections exploiting cultural context," in *14th International Society for Music Information Retrieval Conference (ISMIR 2013)*, Curitiba, Brazil, 2013, pp. 101–106.

[2] J. Buhler and M. Tompa, "Finding motifs using random projections." *Journal of computational biology: a journal of computational molecular cell biology*, vol. 9, no. 2, pp. 225–42, Jan. 2002.

[3] C. Herley, "ARGOS: automatically extracting repeating objects from multimedia streams," *IEEE Transactions on Multimedia*, vol. 8, no. 1, pp. 115–129, Feb. 2006.

[4] M. Atallah, Y. Genin, and W. Szpankowski, "Pattern matching image compression: algorithmic and empirical results," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 7, pp. 614–627, Jul. 1999.

[5] L. J. Van der Maaten and E. O. Postma, "Texton-based analysis of paintings," in *SPIE Optical Engineering+ Applications*. International Society for Optics and Photonics, 2010, pp. 77 980H–77 980H.

[6] N. Cook, *A guide to musical analysis*. London, UK: J.M. Dent and Sons, 1987.

[7] F. Lerdahl and R. Jackendoff, *A generative theory of tonal music*. Cambridge: MIT Press, 1983.

[8] J. Paulus, M. Müller, and A. Klapuri, "State of the art report: Audio-based music structure analysis." in *Proc. of Int. Society for Music Information Retrieval Conf. (ISMIR)*, 2010, pp. 625–636.

[9] B. Janssen, W. B. D. Haas, A. Volk, and P. V. Kranenburg, "Discovering repeated patterns in music: state of knowledge, challenges, perspectives," in *Proc. of the 10th International Symposium on Computer Music Multidisciplinary Research*, Marseille, 2013, pp. 225–240.

[10] O. Lartillot, "Multi-dimensional motivic pattern extraction founded on adaptive redundancy filtering," *Journal of New Music Research*, vol. 34, no. 4, pp. 375–393, 2005.

[11] R. B. Dannenberg and N. Hu, "Pattern discovery techniques for music audio," *Journal of New Music Research*, vol. 32, no. 2, pp. 153–163, 2003.

[12] T. Collins, S. Böck, F. Krebs, and G. Widmer, "Bridging the audio-symbolic gap: The discovery of repeated note content directly from polyphonic music audio," in *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[13] T. Viswanathan and M. H. Allen, *Music in South India*. Oxford University Press, 2004.

[14] P. Rao, J. C. Ross, K. K. Ganguli, V. Pandit, V. Ishwar, A. Bellur, and H. A. Murthy, "Classification of melodic motifs in raga music with time-series matching," *Journal of New Music Research*, vol. 43, no. 1, pp. 115–131, Jan. 2014.

[15] J. C. Ross, T. P. Vinutha, and P. Rao, "Detecting melodic motifs from audio for Hindustani classical music," in *Proc. of Int. Conf. on Music Information Retrieval (ISMIR)*, 2012, pp. 193–198.

[16] V. Ishwar, S. Dutta, A. Bellur, and H. Murthy, "Motif spotting in an Alapana in Carnatic music," in *Proc. of Int. Conf. on Music Information Retrieval (ISMIR)*, 2013, pp. 499–504.

[17] E. Cambouropoulos, "Musical parallelism and melodic segmentation: a computational approach," *Music Perception*, vol. 23, no. 3, pp. 249–268, 2006.

[18] D. Conklin, "Discovery of distinctive patterns in music," *Intelligent Data Analysis*, vol. 14, pp. 547–554, 2010.

[19] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proc. of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, New York, USA, 2003, pp. 2–11.

[20] A. Mueen, E. Keogh, Q. Zhu, S. Cash, and B. Westover, "Exact discovery of time series motifs," in *Proc. of SIAM Int. Con. on Data Mining (SDM)*, 2009, pp. 1–12.

[21] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh, "Addressing big data time series: mining trillions of time series subsequences under dynamic time warping," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 7, no. 3, pp. 10:1–10:31, Sep. 2013. [Online]. Available: http://doi.acm.org/10.1145/2500489

[22] J. Salamon and E. Gómez, "Melody extraction from polyphonic music signals using pitch contour characteristics," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.

[23] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "Essentia: an audio analysis library for music information retrieval," in *Proc. of Int. Society for Music Information Retrieval Conf. (ISMIR)*, 2013, pp. 493–498.

[24] S. Gulati, A. Bellur, J. Salamon, H. Ranjani, V. Ishwar, H. A. Murthy, and X. Serra, "Automatic tonic identification in Indian art music: approaches and evaluation," *Journal of New Music Research*, vol. 43, no. 1, pp. 55–73, 2014.

[25] M. Slaney, "Auditory toolbox: A matlab toolbox for auditory modeling work," *Technical Report*, 1998.

[26] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning*, 2nd ed. Berlin, Germany: Springer, 2009.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] S. Gulati, J. Serrà, K. K. Ganguli, and X. Serra, "Landmark detection in hindustani music melodies," in *Proc. of Int. Computer Music Conf., Sound and Music Computing Conf.*, Athens, Greece, 2014, pp. 1062–1068.

[29] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. on Acoustics, Speech, and Language Processing*, vol. 26, no. 1, pp. 43–50, 1978.

[30] Y. Zhu and D. Shasha, "Warping indexes with envelope transforms for query by humming," in *proc. of the ACM SIGMOD Int. Conf. on on Management of data*, New York, USA, 2003, pp. 181–192.

[31] X. Serra, "Creating research corpora for the computational study of music: the case of the Compmusic project," in *Proc. of the 53rd AES International Conference on Semantic Audio*, London, Jan. 2014.

[32] A. Srinivasamurthy, G. K. Koduri, S. Gulati, V. Ishwar, and X. Serra, "Corpora for music information research in indian art music," in *Proc. of Int. Computer Music Conf., Sound and Music Computing Conf.*, Athens, Greece, 2014, pp. 1029–1036.

[33] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge university press Cambridge, 2008, vol. 1.

[34] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, vol. 18, no. 1, pp. 50–60, 1947.

[35] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian journal of statistics*, vol. 6, no. 2, pp. 65–70, 1979.