MULTI-PART PATTERN ANALYSIS: COMBINING STRUCTURE ANALYSIS AND SOURCE SEPARATION TO DISCOVER INTRA-PART REPEATED SEQUENCES

Jordan B. L. Smith **Masataka Goto**
National Institute of Advanced Industrial Science and Technology (AIST), Japan
jordan.smith@aist.go.jp, m.goto@aist.go.jp

ABSTRACT

Structure is usually estimated as a single-level phenomenon with full-texture repeats and homogeneous sections. However, structure is actually multi-dimensional: in a typical piece of music, individual instrument parts can repeat themselves in independent ways, and sections can be homogeneous with respect to several parts or only one part. We propose a novel MIR task, multi-part pattern analysis, that requires the discovery of repeated patterns within instrument parts. To discover repeated patterns in individual voices, we propose an algorithm that applies source separation and then tailors the structure analysis to each estimated source, using a novel technique to resolve transitivity errors. Creating ground truth for this task by hand would be infeasible for a large corpus, so we generate a synthetic corpus from MIDI files. We synthesize audio and produce measure-by-measure descriptions of which instruments are active and which repeat themselves exactly. Lastly, we present a set of appropriate evaluation metrics, and use them to compare our approach to a set of baselines.

1. INTRODUCTION

Music structure is important to listeners and researchers, but annotating music is hard because typical songs include multiple independent instrument parts. For example, if two sections share the same basic melody, but one features an extra horn part, should one section be labeled as a repetition of the other? To decide, the annotator must consider all the ways in which the two sections are similar or different, but the outcome of their decision is encoded in a single bit: whether the label is the same or not. The annotation discards many of the decisions made by the listener, especially when these are made at the timescale of entire sections. For example, the second verse of Oasis' "Wonderwall" has the same chords and melody as the first, but different lyrics, and it includes two new instruments, cello and drums—the latter of which enters a measure late. A single large-scale section label cannot encode this interesting situation.

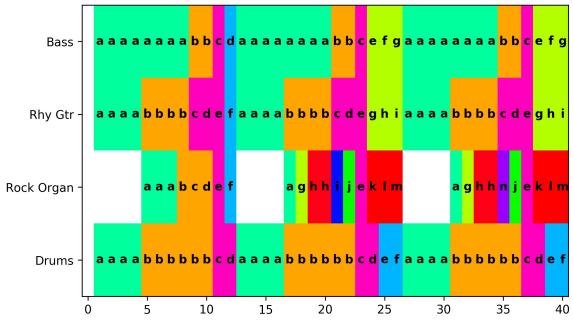

Figure 1. Example multi-part pattern description for the first 40 measures of"Come Together". Measures that repeat are given the same letter label. In this and later figures, the colors highlight repeated sequences instead of individual labels: if label i is always followed by j, and j always follows i, their color assignments are merged.

The multi-dimensional nature of structure has been commented on [22], and recent corpora of annotations have addressed it in different ways: the SALAMI dataset provides descriptions at two timescales, and of functions and leading instrument [29], and the INRIA dataset describes how segments and their component patterns are hierarchically related [2]. For music cognition research, [26] suggested that music be annotated multiple times on a per-feature basis: e.g., once while focusing only on harmony, again while focusing on timbre, and so forth. However, the challenge of hierarchy is different from the challenge of multiple independent parts. We argue that estimating the structure of these independent parts—i.e., creating a multi-part pattern analysis—should be a new MIR goal.

An example of a multi-part pattern description is shown in Fig. 1. It is derived from a MIDI transcription of "Come Together" by The Beatles from the Lakh dataset [23]. It indicates whenever an instrument in the mixture repeats itself, at the timescale of measures. This representation makes clear that the organ part (here substituting the lead vocals) is varied in the second verse, while the other instruments repeat themselves exactly. Compared to a one-dimensional structural analysis, the richer detail of a multi-part description would be more suitable for applica-

tions like automatically editing videos or choreographies to match an audio file.

We make four main contributions in this work. First, we define a new goal for MIR research; second, we propose an algorithm for accomplishing it, which uses existing technology and some new techniques; third, we propose an evaluation framework for the task, including metrics, baselines, and how to obtain ground truth; and finally, we conduct an evaluation.

In the next section, we discuss how our proposed task relates to existing MIR tasks. We present our algorithm in Section 3, present the evaluation framework in Section 4, and discuss the results in Section 5.

2. RELATED WORK

Identifying repeating motives has long been of interest to musicologists in MIR. Although most research in this area has focused on symbolic data analysis (see, e.g., [5]), when "Discovery of Repeated Themes" was added to MIREX in 2013, it included both symbolic and audio tracks (e.g., [21])—but the focus of that task is different: in it, the challenge is precisely to ignore the differences between instruments (if the piece being analyzed contains multiple parts) as well as, potentially, to ignore differences in key or modality. Our task, multi-part pattern analysis, involves a separate challenge: discovering repetitions expressed by a single voice within the mixture.

Since it involves describing the independent patterns in a mixture of tracks, the task is clearly related to source separation. Recently, approaches to source separation have become more structural, taking better advantage of the redundancies offered by repetition in music. One common technique, non-negative matrix factorization (NMF), separates sources by modeling steady states in the spectrum; an extended version, NMF decomposition (NMFD), models short sequences that are time-varying but exactly-repeating [28], and NMF was recently used to detect long loops [15]. Median filtering, which was used to efficiently perform harmonic-percussive source separation (HPSS) [6], was used in the REPET algorithm to separate a repeating background from a mixture [14]; REPET was later adapted to looping backgrounds that change over time and heterogenous backgrounds [25]. Although estimating a multi-part pattern analysis will *require* source separation, the desired output is an abstract description, not a set of separated tracks. Thus, whereas a source separation is evaluated with signal reconstruction error, a pattern analysis will be evaluated more like a structural analysis.

As for structural analysis, it has evolved toward modeling hierarchy. Early segmentation-only approaches [7] were followed by approaches that also estimate labels [8], and by approaches that model similarity differently at different timescales [11]. Since the creation of the multi-scale SALAMI and INRIA annotations, approaches to hierarchical description have been refined [17], as has the methodology for evaluating them [18]. Hierarchy is partly a consequence of multiple sources behaving independently: three repetitions of the chorus could be considered the same at a coarse timescale (the context of the song), but differences in range or instrumentation could differentiate them at a finer timescale (the context of the three choruses). Models of hierarchy will always be ambiguous, since its perception is ambiguous [12]. In contrast, the multi-layered composition of a song can be described more concretely. Thus, multi-part pattern analysis is worth treating separately from hierarchical structure, and a good multi-part analysis may be very useful for describing hierarchy.

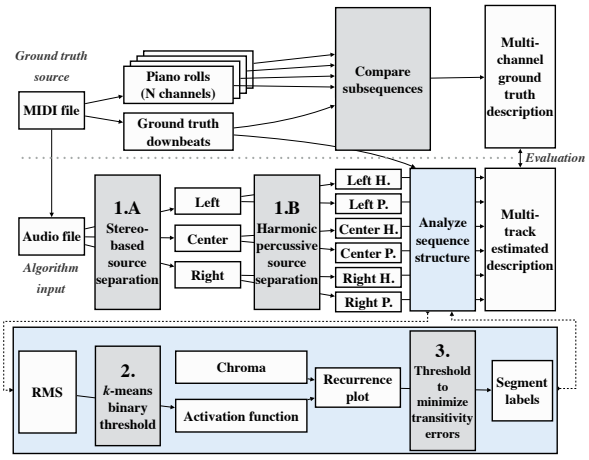

Figure 2. Algorithm and ground truth generation pipeline.

Finally, two works have directly bridged source separation and structural analysis: First, [10] found that structure analysis could be performed more accurately with multi-track audio as input. Second, [27] discovered that spikes in the reconstruction error of a source separation algorithm can indicate structural boundaries. In defining the task of multi-part pattern description, we hope to bring these fields closer together.

3. PROPOSED APPROACH

Our proposed algorithm is outlined in Fig. 2, and data at certain intermediate steps are illustrated in Fig. 3. The three key stages of the algorithm are:

1. Source separation. We apply source separation to the audio to convert the stereo recording to an estimated multi-track recording. We do this with two median spectral filters [13]: first, we take the median of the left and right channels to estimate the center channel, and subtract this from the original signals, resulting in three tracks. Second, we apply HPSS to each track [6]. Even if a track contains multiple pitched instruments, HPSS can separate instruments with different attacks, such as piano vs. strings, or rhythm guitar vs. organ. We end up with 6 audio tracks (see Fig. 3b).

2. Activation function estimation. The separated tracks may be sparse: e.g., if the left channel contains only strings, the left-percussive component may be nearly empty. We compute RMS to estimate when the channels are active. At this stage, we also use the ground truth downbeat labels to define our segment windows. In future

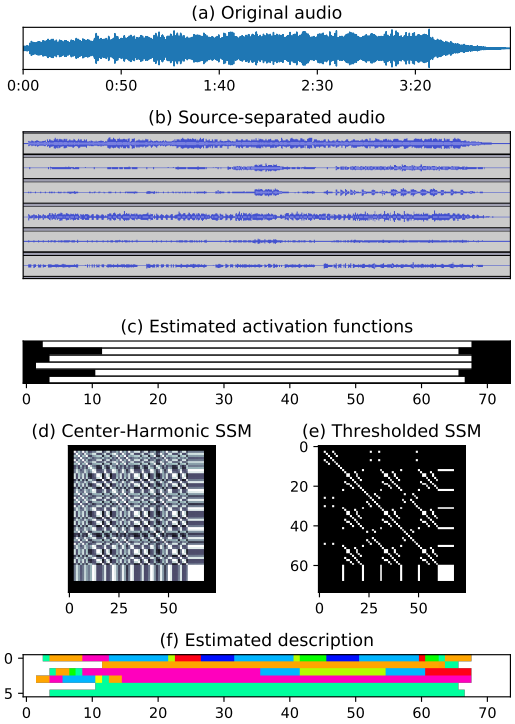

Figure 3. Data in intermediate stages of algorithm pipeline. The SSMs in plots (d) and (e) correspond to the center-harmonic track, which is the first track in plots (b) and (c). Sound example is "Across The Universe."

work, beats and downbeats could be estimated instead.

We take the mean over each window (i.e., each measure), and apply a k-means clustering to the RMS values, with $k = 2$, to classify windows as either silent or active. Even if the classes are very uneven, the difference between the two with respect to RMS tends to be extreme enough that this method is effective. At the end of this stage we have a set of estimated activation functions (see Fig. 3c).

3. Sequence analysis. We use self-similarity matrices (SSMs) to discover repetitions in each track. We compute chroma with the madmom package [4] and compute a measure-indexed SSM: element i, j gives the cosine similarity between the sequences of beat-synchronized chroma features of the i^{th} and j^{th} measures. We also use the previously-estimated activation functions to zero out the SSM when the track was judged inactive, as shown at the beginning and end of the track in Fig. 3d.

To estimate segment labels from the real-valued SSM, we choose a threshold t to binarize the matrix; then, to emphasize diagonal lines, we apply a single erosion-dilation operation (in time-lag space) with a kernal size k. We choose t and k in a novel way: my finding the values that minimize the number of transitivity errors. These errors are resolved with a novel lexical-sort approach. Transitivity errors are cases where a segment i is judged to be similar to both j and k, but segments j and k are not similar to each other; resolving these inconsistencies is a difficult part of interpreting structure from SSMs (e.g., see [20]).

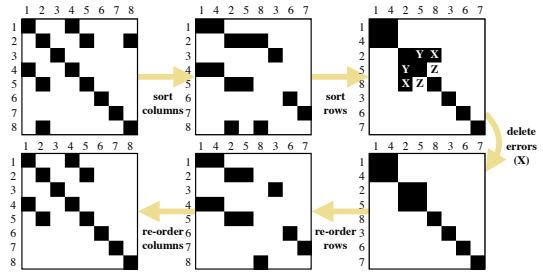

Figure 4. Eliminating transitivity errors with lexical sorting. Errors appear as inconsistent blocks in the sorted SSM. We can fix the error by eliminating pixels X, or pixels Y, or adding pixels at Z.

Given a binary SSM, we can collect repeating pixels into groups by sorting the rows lexically (i.e., alphabetically). The process is illustrated in Fig. 4: after sorting the SSM's rows and columns, groups of repeating elements become blocks on the main diagonal, and all other pixels represent transitivity errors. The third SSM in Fig. 4 can be fixed in three ways: zeroing the pixels at X, or at Y, or adding pixels at Z. We greedily eliminate the errors by walking along the main diagonal from the upper left and discarding off-diagonal elements that do not fit the current block, which corresponds to zeroing X. When the cleaned SSM is re-ordered, the result is guaranteed to be transitive.

We call the number of pixels deleted from an SSM the "strain", and the number of off-diagonal pixels that remain the "coverage". (For the example in Fig. 4, strain is 2 and coverage is 4.) Our goal is to choose t and k to maximize coverage and minimize strain, while avoiding redundant cases such as an empty SSM or an SSM that is all ones.

We sweep values of t between 0.99 and 0.8, and k between 4 and 8 measures. A set of real-world examples are shown in Fig. 5. The left column contains 5 binary SSMs derived from chroma computed on an audio track. The second and third columns give the lexically-ordered SSMs (and their strains) and their cleaned versions (and coverages). The fourth column gives the cleaned SSMs and the difference between coverage and strain, which is maximized by choosing $k = 7$. The result is a binary SSM that is sparse but not empty, and free of transitivity errors, as in Fig. 3e. It is then trivial to label the segments. The six estimated part descriptions are collected in Fig. 3f.

In structure analysis, we typically search for long repeating subsequences and long homogeneous stretches, and apply strong smoothing to the SSM to gloss over variations. In contrast, the above pipeline was designed to focus on tracking shorter patterns and to find when they repeat exactly, with the expectation of obtaining a much sparser SSM with few transitivity errors.

4. EVALUATION FRAMEWORK

4.1 Data and Ground Truth

To test the quality of a multi-part pattern analysis algorithm, we need audio files with multiple layers, with each

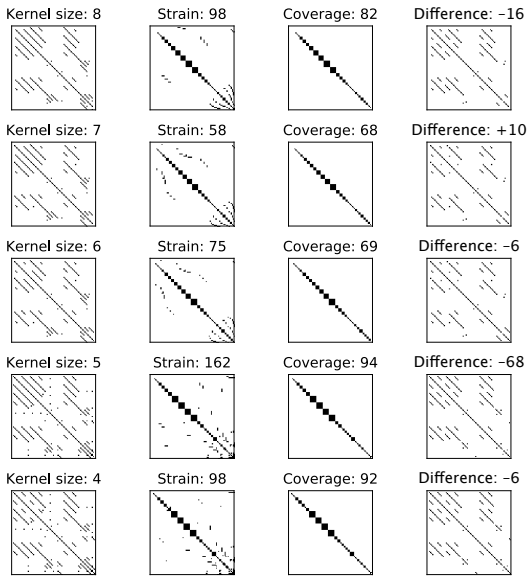

Figure 5. Illustration of strain-coverage optimization approach on a track estimated from a recording of "All My Loving." The four columns, from left, give: (1) binary SSMs filtered with different kernel sizes; (2) lexically sorted SSMs; (3) SSMs with errors removed; (4) cleaned SSMS restored to original column and row order. Kernel size 7 maximizes coverage while minimizing strain.

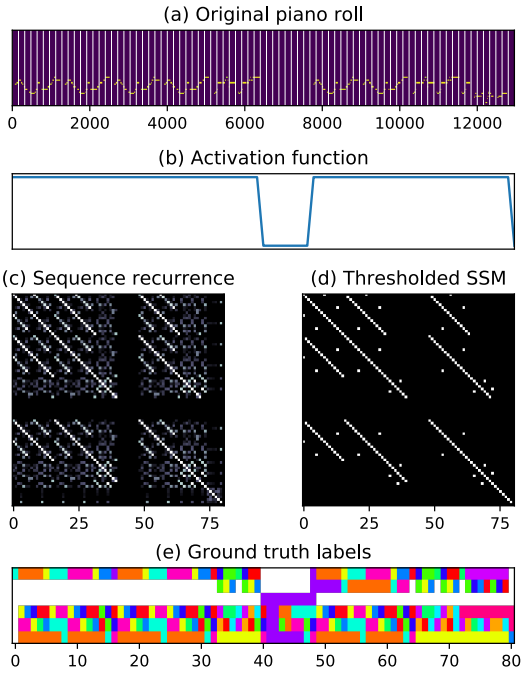

Figure 6. Data in intermediate stages of ground truth generation for the vocal channel of "All My Loving." The song's multi-part description is shown in (e).

layer annotated to indicate repeating patterns. Creating ground truth for this task by hand would be infeasible for a large corpus. There are many public datasets of multitrack audio, but only rarely are the tracks annotated in detail. The existing dataset that most closely meets our needs is MedleyDB [3], which contains multitrack audio and melody f0 annotations for a subset of stems, but not annotations of repetitions in each track.

However, we can generate an appropriate dataset from MIDI. We used a portion of Lakh MIDI dataset [23] called the "Clean MIDI subset", which contains most of the Beatles catalogue, and used FluidSynth [1] to convert these to audio files. When there were duplicate MIDI files to choose from, we selected the version where the average panning setting of the tracks had the highest standard deviation. (Many MIDI transcriptions have no panning information at all, which would work against our algorithm.)

We processed the MIDI files (using Pretty MIDI [24]) to create, for each MIDI channel, a ground truth description of the measure-level patterns. The procedure for this is similar to our analysis algorithm (see Fig. 3). From a downbeat-segmented piano roll (Fig. 6a), we obtain an activation pattern, i.e., a timeline of 1s and 0s indicating whether an instrument has any MIDI note events during each measure-long window (Fig. 6b). Next, we estimate the similarity of every pair of measures within a track with an SSM (Fig. 6c). To compare two piano roll windows, we take the percentage of active note spans that overlap. To focus on exact repetitions, we should use a threshold of 1.0, but in practice, due to small timing differences and expressive gestures in the MIDI transcription, a threshold of 1.0 leads to extremely sparse recurrence plots—but on the other hand, lowering the threshold can lead to transitivity errors, as before. However, we found that a threshold of 0.9 was generally suitable to obtain non-empty recurrence plots without producing a large number of transitivity errors (Fig. 6d). Doing this for every track gives a multi-part description (Fig. 6e).

[1] `http://www.fluidsynth.org/`

4.2 Evaluation Metrics

After processing the MIDI data, we obtain a "ground truth" matrix of instrument patterns A where the element $A_{i,j}$ indicates the pattern label for the i^{th} instrument during the j^{th} measure. (Such information is displayed in Fig. 1 and Fig. 6e.) We set $A_{i,j} = 0$ when the i^{th} instrument is not active. Similarly, we can obtain an estimated description E with elements $E_{i,j}$, such as in Fig. 3f.

To compare two single-track descriptions (two rows of A and E), we can use any metrics from the field of structure analysis, such as the pairwise f-measure metric [19]. (For a comparison of structure evaluation metrics, see [16].) However, the rows of A and E are not necessarily aligned in the correct order. Moreover, the number of estimated tracks in E may be smaller or greater than the number of MIDI channels in A. We present two sets of evaluation metrics: one that requires matching the layers, and one that does not. We also devise a set of baselines.

Evaluation of descriptions. Suppose we have an N-layer estimated description and an M-layer ground truth

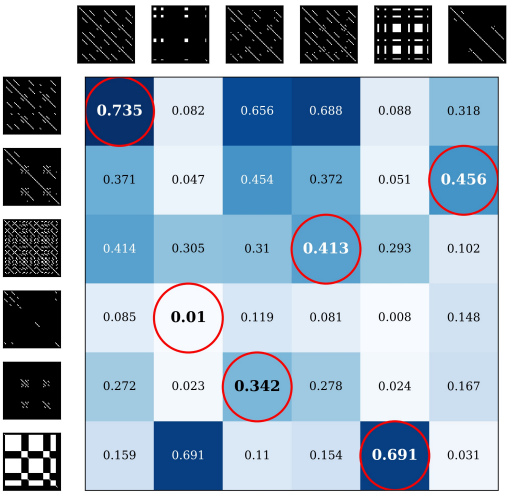

Figure 7. Optimal pairing of estimated tracks (columns) with ground truth channels (rows), according to pairwise f-measure between descriptions (which are illustrated as recurrence plots) The mean pw_f is 0.44.

description, and let $L = \min(M, N)$. We can compute the pairwise f-measure between all pairs of layers, giving a table of values like the one in Fig. 7. The Hungarian Algorithm [2] gives us the optimal one-to-one matching between L layers to maximize the average f-measure. Using the optimal pairings, we can compute the average precision and recall. Together, these serve as our set of 3 "generous" metrics, since it does not punish cases when $N \neq M$. If there are unmatched layers, whether in A or E, these should count against the estimate. We can compute a stricter mean f-measure by taking $pw_f * L/\max(M, N)$.

Evaluation of activations. The activation matrix that we estimate (e.g., Fig. 3c) is an important intermediate step. It is worth evaluating on its own, and we can do so without matching tracks to channels. We treat each column of the activation matrix, an N-dimensional binary vector, as a 'timbre label,' such that each unique column gets a unique label. (This calls to mind the timbre-mixture estimation of [1].) We perform the same process on the ground truth activation matrix. Then we can use pairwise f-measure to compare the two sequences of timbre labels.

This metric ignores the difference between an instrument being added to or subtracted from the mixture. To evaluate the retrieval of entrances and exits, we use a version of the boundary retrieval f-measure [19, p. 220], counting each entrance (or exit) in the ground truth as being correctly estimated only if some instrument in the estimated description also enters (or exits, respectively) in the same measure.

4.3 Baseline Methods

We compare our algorithm against a set of naive baselines to gauge the success of our algorithm, but also to learn how the proposed evaluation metrics behave. The labeling baselines are:

- $B_{constant}$: all measures repeat the same pattern;
- B_{null}: all measures are unique;
- $B_{periodic}$: there are three concurrent tracks playing sequence loops of length 2, 4 and 8 measures: i.e., three sequences $[ab]*$ (i.e., ab repeated), $[abcd]*$, and $[abcdefgh]*$;
- B_{block}: there are three concurrent tracks that alternate static textures with periods 2, 4 and 8 measures: i.e., $[ab]*$, $[aabb]*$, and $[aaaabbbb]*$.

The activation matrix baselines are:

- $B_{uniform}$: the song has a single texture;
- $B_{buildup}$: new instruments enter in measures 3, 5 and 7.

In addition to these naive baselines, we tested two simplified versions of our proposed approach. The first skips the source separation step: instead of estimating patterns from 6 separated tracks, we can estimate patterns from the full-audio chroma features, and then duplicate the result 6 times to match the number of estimated sources as the proposed approach ("Chr. w/o SS"). Second, since the activation matrix is evaluated as if it were a timbre label, we also estimate timbre labels by computing full-audio MFCCs, and using NMF to label the measures ("MFCCs"). All section transitions are treated as predictions of entrances and exits.

5. RESULTS AND DISCUSSION

We applied all the approaches described above to the dataset of 200 Beatles songs. The results for the multi-part pattern description task are shown in Table 1 ("Standard approach"). We find that the proposed approach outperforms the naive baselines, but that the simpler approach that skips the source separation step performs even better, even though it has lower recall. The pw_f values are almost all dominated by the lower precision values; like in structure analysis, it seems harder to achieve high precision than high recall. By tweaking the evaluation metric, we can understand why. In the bottom half of the table, we compute pw_f counting the elements on the main diagonal. The B_{null} baseline, which guesses that every measure is different, now becomes very competitive.

The explanation is that unlike in the usual structure analysis task, the ground truth for this task is very sparse. Recall that pairwise f-measure tells us how well the similarity relationships of one description are captured by the similarity relationships in another. In other words, given two binary SSMs that encode similarity descriptions, pw_f assesses how well the positive parts of these SSMs coincide. Since it is trivial to guess that each segment is similar to itself, we should ignore the contributions of the main diagonal. This does not usually affect the outcome of structure evaluation, since the repeating blocks ensure that the

[2] https://en.wikipedia.org/wiki/Hungarian_algorithm

Standard approach				
	Generous			*Strict*
	pw_f	pw_p	pw_r	pw_f
Proposed	.245	.211	.71	.184
Chr. w/o SS	**.297**	**.312**	.529	**.222**
$B_{constant}$	.144	.092	**.95**	.106
B_{null}	0	0	0	0
$B_{periodic}$	.149	.136	.318	.112
B_{block}	.06	.103	.074	.044
Counting self-labeled measures				
	pw_f	pw_p	pw_r	pw_f
Proposed	.365	.309	.819	.274
Chr. w/o SS	.466	.442	.695	.346
$B_{constant}$	.183	.115	**.962**	.135
B_{null}	**.515**	**.749**	.477	**.384**
$B_{periodic}$	.255	.218	.461	.192
B_{block}	.411	.44	.465	.309

Table 1. Above: results for estimated multi-track description quality using the proposed metric. Below: the results if self-labeled measures are counted as correct. The high retrieval for B_{null} illustrates the sparseness of the ground truth.

ground truth SSM has very many off-diagonal pixels to estimate. However, in our application, the ground truth matrices are extremely sparse: in cases where a part never repeats itself exactly, there are no off-diagonal elements.

On the other hand, this task is unlike structure analysis because in our case, elements on the main diagonal *can* equal 0, if the corresponding source is not active. This means that the B_{null} baseline does not actually achieve perfect precision: from the bottom part of Table 1 we can see that on average, sources are active for 75% of the song.

Results for the activation detection task are shown in Table 2. According to the pw_f measure, the best approach to characterize the changing timbre of the piece was our proposed one. However, the uniform baseline performs almost as well according to this metric. Although some songs have over a dozen tracks, with many entrances and exits, it seems that the majority of songs have an instrumentation that changes little. As a result, the uniform and buildup baselines achieve near-perfect recall while precision does not fall below 30%. That said, these naive baselines fail to detect nearly all the entrances and exits of instruments from the mixture, so the proposed approach beats them handily on entrance/exit f-measure.

In contrast, the MFCC approach tends to find a majority of the entrances and exits, and narrowly beats the proposed approach in terms of entrance/exit f-measure. The cost of this apparent over-segmentation is lower pairwise retrieval, and the lowest overall pw_f, for labeling the timbres.

In designing the evaluation, we made an effort to reuse metrics that are used for structure analysis. We did not expect the sparseness of the ground truth to have such an impact on the metrics, but the impact is plain to see in the success of the baselines. Perhaps we should have

	Timbre labeling			*Entrance/exit*		
	pw_f	pw_p	pw_r	f	p	r
Proposed	**.450**	.456	.546	.248	.271	.296
MFCCs	.3	**.549**	.319	**.273**	.195	**.566**
$B_{uniform}$	.433	.306	**.962**	0	0	0
$B_{buildup}$	.446	.328	.909	.071	**.351**	.045

Table 2. Results for estimated activation matrix quality.

anticipated this: data sparseness is often a problem when translating a one-dimensional function (here, the overall structure) into a higher-dimensional space (a per-channel representation). One potential way to resolve this issue is to automatically process both the ground truth and the estimated descriptions using a fixed sequences-to-blocks conversion step, such as that proposed by [9]. This would allow us to compare nearly-equivalent representations that are much less sparse.

Needless to say, the multi-track analysis approach we have proposed could be improved in many ways. We have used two source separation kernels, in a fixed way, but it is possible to apply more kernels, and to do so in an optimization framework to increase the independence of the estimated tracks [13]. Future work should also test a greater variety of source separation methods, especially NMF-based approaches. However, this first effort has helped us to understand the special challenge of this task, which is the sparseness of the ground truth.

6. CONCLUSION AND FUTURE WORK

We have described a new MIR task, multi-part pattern analysis, in which the goal is to describe each independent layer of a piece of music. The task complements recent work on estimating hierarchical structure. We have also proposed a method for estimating multi-part pattern analyses using a combination of existing source-separation tools, SSM-based structure estimation methods, and a novel approach to thresholding SSMs in order to minimize transitivity errors.

To support future work on this problem, we have proposed a method of creating ground truth annotations from MIDI files, and a set of evaluation metrics that can estimate the similarity between two multi-part descriptions or two multi-part activation functions.

In our evaluation, we found the sparseness of the data to be an issue, but it is a direct consequence of how we chose to create the ground truth. As we refine the methodology for this task in future work, we will study the impact of different ways of converting multi-channel files into ground truth recurrence plots.

7. ACKNOWLEDGEMENTS

This work was supported in part by JST ACCEL Grant Number JPMJAC1602, Japan.

8. REFERENCES

[1] J.-J. Aucouturier and Mark Sandler. Segmentation of musical signals using hidden Markov models. In *Proc. of the Audio Engineering Society Convention*, Amsterdam, The Netherlands, 2001.

[2] Frédéric Bimbot, Gabriel Sargent, Emmanuel Deruty, Corentin Guichaoua, and Emmanuel Vincent. Semiotic description of music structure: An introduction to the Quaero/Metiss structural annotations. In *Proc. of the AES Conference on Semantic Audio*, 2014.

[3] Rachel M. Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Pablo Bello. MedleyDB: A multitrack dataset for annotation-intensive mir research. In *Proc. of ISMIR*, pages 155–160, Taipei, Taiwan, 2014.

[4] Sebastian Böck, Filip Korzeniowski, Jan Schlüter, Florian Krebs, and Gerhard Widmer. madmom: a new Python Audio and Music Signal Processing Library. In *Proc. of the ACM International Conference on Multimedia*, pages 1174–1178, Amsterdam, The Netherlands, 2016.

[5] Tom Collins, Andreas Arzt, Sebastian Flossman, and Gerhard Widmer. SIARCT-CFP: Improving precision and the discovery of inexact musical patterns in point-set representations. In *Proc. of ISMIR*, pages 549–554, Curitiba, Brazil, 2013.

[6] Derry FitzGerald. Harmonic/percussive separation using median filtering and amplitude discrimination. In *Proc. of the International Conference on Digital Audio Effects*, Graz, Austria, September 2010.

[7] Jonathan Foote. Automatic audio segmentation using a measure of audio novelty. In *Proc. of the IEEE International Conference on Multimedia & Expo*, pages 452–455, 2000.

[8] Jonathan Foote and Matthew Cooper. Media segmentation using self-similarity decomposition. In Minerva Yeung, Rainer Lienhart, and Chung-Sheng Li, editors, *Proc. of the SPIE: Storage and Retrieval for Media Databases*, volume 5021, pages 167–175, Santa Clara, CA, USA, 2003.

[9] Harald Groghanz, Michael Clausen, Nanzhu Jiang, and Meinard Müller. Converting path structures into block structures using eigenvalue decompositions of self-similarity matrices. In *Proc. of ISMIR*, 2013.

[10] Steven Hargreaves, Anssi Klapuri, and Mark Sandler. Structural segmentation of multitrack audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(10):2637–2647, 2012.

[11] Tristan Jehan. Hierarchical multi-class self similarities. In *Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 311–314, New Paltz, NY, United States, 2005.

[12] Fred Lerdahl and Ray S. Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983.

[13] Antoine Liutkus, Derry Fitzgerald, Zafar Rafii, Bryan Pardo, and Laurent Daudet. Kernel additive models for source separation. *IEEE Transactions on Signal Processing*, 62(16):4298–4310, 2014.

[14] Antoine Liutkus, Zafar Rafii, Roland Badeau, Bryan Pardo, and Gaël Richard. Adaptive filtering for music/voice separation exploiting the repeating musical structure. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 53–56, Kyoto, Japan, 2012. IEEE.

[15] Patricio López-Serrano, Christian Dittmar, Jonathan Driedger, and Meinard Müller. Towards modeling and decomposing loop-based electronic music. In *Proc. of ISMIR*, pages 502–508, New York, NY, USA, 2016.

[16] Hanna Lukashevich. Towards quantitative measures of evaluating song segmentation. In *Proc. of ISMIR*, pages 375–380, Philadelphia, PA, USA, 2008.

[17] Brian McFee and Daniel Ellis. Analyzing song structure with spectral clustering. In *Proc. of ISMIR*, pages 405–410, Taipei, Taiwan, 2014.

[18] Brian McFee, Oriol Nieto, and Juan Pablo Bello. Hierarchical evaluation of segment boundary detection. In *Proc. of ISMIR*, Málaga, Spain, 2015.

[19] Meinard Müller. Music structure analysis. In *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*, pages 167–236. Springer International Publishing, 2015.

[20] Meinard Müller, Nanzhu Jiang, and Peter Grosche. A robust fitness measure for capturing repetitions in music recordings with applications to audio thumbnailing. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(3):531–543, 2013.

[21] Oriol Nieto and Morwaread M. Farbood. Identifying polyphonic patterns from audio recordings using music segmentation techniques. In *Proc. of ISMIR*, pages 411–416, Taipei, Taiwan, 2014.

[22] Geoffroy Peeters and Emmanuel Deruty. Is music structure annotation multi-dimensional? A proposal for robust local music annotation. In *Proc. of the International Workshop on Learning the Semantics of Audio Signals*, pages 75–90, Graz, Austria, 2009.

[23] Colin Raffel. *Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching*. PhD thesis, Columbia University, 2016.

[24] Colin Raffel and Daniel P. W. Ellis. Intuitive analysis, creation and manipulation of midi data with pretty_midi. In *ISMIR Late Breaking and Demo Papers*, pages 84–93, 2014.

[25] Zafar Rafii, Antoine Liutkus, and Bryan Pardo. REPET for background/foreground separation in audio. In G. R. Naik and W. Wang, editors, *Blind Source Separation*, Signals and Communication Technology, pages 395–411. Springer-Verlag, 2014.

[26] Chris Sanden, Chad R. Befus, and John Z. Zhang. A perceptual study on music segmentation and genre classification. *Journal of New Music Research*, 41(3):277–293, 2012.

[27] Prem Seetharaman and Bryan Pardo. Simultaneous separation and segmentation in layered music. In *Proc. of ISMIR*, pages 495–501, New York, NY, USA, 2016.

[28] Paris Smaragdis. Non-negative matrix factor deconvolution: Extraction of multiple sound sources from monophonic inputs. In *Independent Component Analysis and Blind Signal Separation*, volume 3195 of *Lecture Notes in Computer Science*, pages 494–499. Springer-Verlag, Berlin, Heidelberg, 2004.

[29] Jordan B. L. Smith, J. Ashley Burgoyne, Ichiro Fujinaga, David De Roure, and J. Stephen Downie. Design and creation of a large-scale database of structural annotations. In *Proc. of ISMIR*, pages 555–560, Miami, FL, United States, 2011.