ENHANCED HIERARCHICAL MUSIC STRUCTURE ANNOTATIONS VIA FEATURE LEVEL SIMILARITY FUSION

Christopher J. Tralie *

Duke University
Department of Mathematics
Durham, NC, USA

Brian McFee

New York University
Music and Audio Research Lab & Center for Data Science
New York, NY, USA

ABSTRACT

We describe a novel pipeline to automatically discover hierarchies of repeated sections in musical audio. The proposed method uses similarity network fusion (SNF) to combine different frame-level features into clean affinity matrices, which are then used as input to spectral clustering. While prior spectral clustering approaches to music structure analysis have pre-processed affinity matrices with heuristics specifically designed for this task, we show that the SNF approach directly yields segmentations which agree better with human annotators, as measured by the "L-measure" metric for hierarchical annotations. Furthermore, the SNF approach immediately supports arbitrarily many input features, allowing us to simultaneously discover structure encoded in timbral, harmonic, and rhythmic representations without any changes to the base algorithm.

Index Terms— music structure analysis, similarity network fusion, spectral clustering

1. INTRODUCTION

Music has structure along many axes, such as timbre, melody, harmony, rhythm, etc. Since most methods for automatic music structure segmentation are tuned to find a particular kind of structure, extending them support other types is usually quite difficult. Hence, we would like to explore how to leverage multiple representations of the same audio to efficiently discover musical structure.

1.1. Our contributions

In this work, we show how to use similarity network fusion ([1, 2], Section 2.1) for musical structure analysis. In particular, our proposed method integrates disparate representations of timbre, harmony, and rhythm (Section 2.3) to produce a unified structure representation. We then couple this method with spectral clustering (Section 2.1) to produce multi-level structure analyses, and we evaluate the system for its ability to recover annotated structure in a diverse music collection (Section 3). Our evaluation includes multiple reference annotations for each track, accounting for subjectivity and diversity of opinion. Overall, we find that our proposed method is more robust than prior work, and gets closer to human-level agreement than prior work. L-recall (Section 3.2.1) is particularly strong with our technique, with a mean of 0.658 compared to the human inter-annotator recall mean of 0.664.

1.2. Related work

Similarity network fusion (SNF) is a joint random walk technique that was devised to leverage the strengths of different hand-designed similarity measures for shape classification 2D contours in images [1]. It has since been used in such tasks as cancer phenotype discrimination [2], image retrieval [3], and drug taxonomy [3]. SNF was introduced to the music information retrieval community by the authors of [4] to leverage different cross-similarity alignment scores in automatic cover song identification. As in the original application, they use SNF *at the object (song) level*. By contrast, it was shown in [5] that using SNF at the *feature level* (i.e., beat-synchronous HPCP and MFCC) can improve cross-similarity matrices between pairs of covers without the need for a network of song-level similarity measures. A precursor to our work used SNF on frame-level features within a song to improve self-similarity matrices for visualization [6].

As for music structure analysis, the present work builds directly upon the Laplacian spectral decomposition (LSD) method [7]. This method operates by carefully constructing a graph which encodes short-term timbral continuity along with long-term harmonic repetition, and then partitions the graph at multiple scales to recover multi-level segmentations. While this can be effective, the graph construction depends heavily upon the choice of input features, and the resulting method can be somewhat brittle in practice. The method we propose here, in contrast, supports the fusion of arbitrarily many input representations, which facilitates the discovery of both long- and short-range structure along many different musical dimensions, including timbre, harmony, and rhythm.

*C. Tralie was supported by an NSF RTG grant NSF-DMS 1045133.

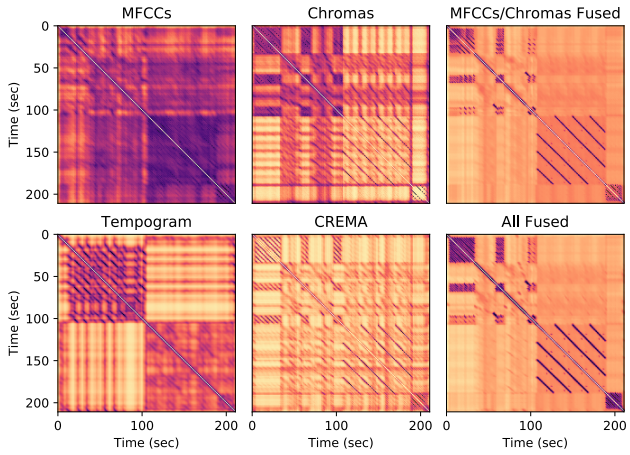

Fig. 1. Applying SNF on the song "Tango Apasionado" by Astor Piazzolla (936 in the SALAMI dataset [8]). Affinity matrices are shown before and after fusion.

2. METHODS

2.1. Fusion

We now provide details of frame-level similarity network fusion. Given F sets of features which are each computed at the same N time intervals each corresponding to a stack-delayed sequence of frames (Section 2.3), we first compute the corresponding F $N \times N$ self-similarity matrices (SSMs) $\{D^f\}_{f=1}^F$ via feature specific distances (Section 2.3). Then, we convert each SSM to an "affinity matrix" $W_{ij}^f = \exp\left(-(D_{ij}^f/\sigma_{ij}^f)^2\right)$ with a pairwise autotuned time-dependent spatial bandwidth σ_{ij}^f, so that, as prescribed by [1] and [2]

$$\sigma_{ij}^f = \frac{1}{6}\left(\frac{1}{\kappa}\left(\sum_{k\in N_\kappa^f(i)} D_{ik}^f + \sum_{k\in N_\kappa^f(j)} D_{kj}^f\right) + D_{ij}^f\right) \quad (1)$$

where κ the number of nearest neighbors which is fixed a priori (we will explore the effect of κ in Section 3.3), and $N_\kappa^f(i)$ are the indices of the κ nearest neighbors of i, as measured by D^f. SNF then defines two additional normalized versions P^f and S^f of each affinity matrix as follows

$$P_{ij}^f = \left\{ \begin{array}{cc} W_{ij}^f/(2\sum_{k\neq i} W_{ik}^f) & j \neq i \\ 1/2 & j = i \end{array} \right\} \quad (2)$$

$$S_{ij}^f = \left\{ \begin{array}{cc} W_{ij}^f/(2\sum_{k\in N_\kappa^f(i)} W_{ik}^f) & j \in N_\kappa^f(i) \\ 0 & \text{otherwise} \end{array} \right\} \quad (3)$$

In other words, each P^f can be interpreted as a doubly-stochastic transition probability matrix associated to W^f, and S^f is the nearest neighbor thresholded version of P^f. Given these matrices, SNF proceeds with the following iterations

$$P_t^f = S^f \times \left(\frac{\sum_{k\neq f} P_{t-1}^k}{F-1}\right) \times (S^f)^\mathsf{T} \quad (4)$$

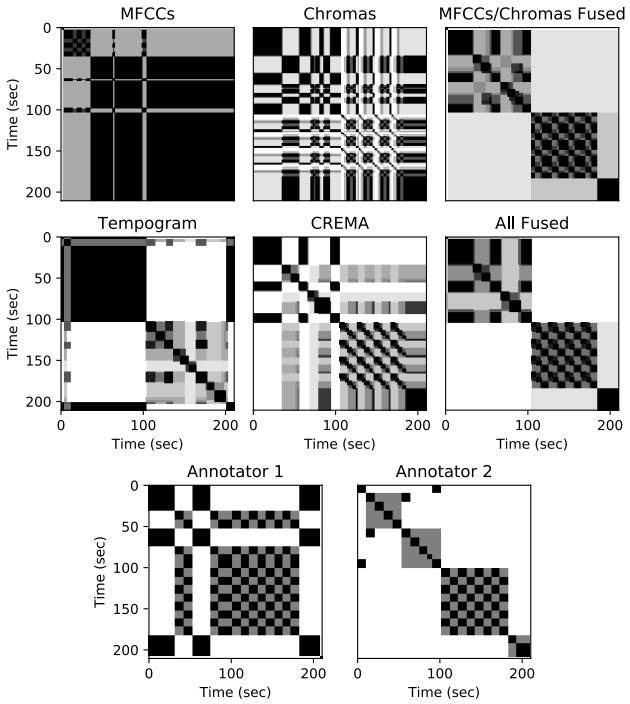

Fig. 2. Applying spectral clustering to the affinity matrices in Figure 1. Meet matrices [9] are shown before and after fusion, in addition to meet matrices from human annotators.

for T iterations $t = 1, 2, \ldots, T$, cycling through $f = 1, 2, \ldots, F$ at each iteration, and with $P_1^f = P^f$ and S^f fixed. The final fused affinity is then taken to be $\frac{1}{F}\sum_{f=1}^F (P_T^f)$. Intuitively, each iteration for feature type f performs a random walk using neighbors of f but probabilities from the other feature types averaged together, thereby fusing information from all features. These iterations have been shown to converge quickly in practice [1, 2], and we find that $T = 10$ suffices.

2.2. Spectral clustering

Once we have clean affinity matrices, we can extract segments from them via spectral clustering [10]. Spectral clustering refers to a family of methods for partitioning graphs based on the characteristics of the eigenvector decomposition of their Laplacian matrix. In this work, we use the *random-walk normalized Laplacian* formulation. Given a symmetric graph affinity matrix $A \in \mathbb{R}_+^{N\times N}$, the normalized Laplacian is defined as

$$L := I - \Delta^{-1}A, \quad (5)$$

where $\Delta = \text{diag}(A\mathbf{1})$ is the diagonal *degree* matrix of A.

L is positive semi-definite, and the eigenvectors associated with the smallest eigenvalues encode the large-scale structure of the graph. Let L have eigenvector decomposition $L = \sum_i \lambda_i v_i v_i^\mathsf{T}$ with λ_i in ascending order. Spectral clustering proceeds by using the first k eigenvectors as

$V_k = [v_0, v_1, \ldots, v_{k-1}] \in \mathbb{R}^{N \times k}$ as a k-dimensional feature representation of the nodes of the graph, which is then given as input to a k-means clustering algorithm (we use sklearn for k-means [11]). The resulting cluster assignments provide a partition of the nodes of the graph into k disjoint subsets.

This general idea was previously applied to multi-level music segmentation [7] by iterating over multiple values of k: small values of k produce few segment *types*, though potentially many individual *segments* of each type. For each value of k, segment boundaries are inferred by finding the nodes $(n, n+1)$ (corresponding to time or beat indices) which receive distinct cluster assignments. In this work, we create a hierarchy of segment labels by varying k from 2 to 10.

Figure 2 shows an example of "meet matrices" [9] on the results of spectral clustering on the affinity matrices from Figure 1. Darker pixels in these matrices correspond to regions which are more consistently labeled across different levels in the label hierarchy. As in [12] and [7], we observe that tight diagonals in the SSMs are expanded as blocks in the annotations. Hence, since SNF enhances diagonals in this example (Figure 1), it leads to cleaner block structures.

2.3. Features

We evaluate the proposed fusion clustering method using four different audio representations, meant to encode various aspects of timbre, harmony, and rhythm. Features are computed with librosa 0.6.2 [13], and sampled at a framerate of 23.2 ms.

As a coarse timbre descriptor, we use 20 mel frequency cepstral coefficients (MFCCs), derived from a 128-dimensional mel spectrum covering 0–11025Hz. We apply an exponential lifter $\hat{x_c} = (c^{0.6})x_c$, to each coefficient $x_c, c = 1, 2, \ldots, 20$.

To encode harmonic content, we use chroma derived from a constant-Q spectrogram of 36 bins per octave. Chroma features capture harmonic content by aggregating pitch class energy across octaves, and can therefore be sensitive to overtones and transients. To capture longer-term harmonic stability, we introduce a second set of features derived from the CREMA chord estimation model [14]. This model uses convolutional-recurrent neural network for large-vocabulary chord recognition, and as a byproduct, produces conditional likelihood of each pitch class being active at each frame. While these features can be interpreted as chroma-like, the recurrent aspect of the model tends to enforce local consistency while suppressing transients and passing tones.[1]

Finally, rhythmic content is encoded by a tempogram derived from the local auto-correlation of the onset strength envelope [15]. The onset strength envelope is calculated with a SuperFlux [16] local max filter of 5 bins on the previous frame, which suppresses vibrato while preserving attack transients [16]. Tempogram auto-correlations are estimated over a window of 384 frames ($\sim$ 9 seconds) and peak-normalized.

[1] CREMA features are produced at a framerate of 44100/4096=10.7Hz, and up-sampled by nearest-neighbor interpolation.

After the initial computation, all features are averaged within non-overlapping chunks of 10 windows, slowing the framerate down to 0.232 seconds. Unlike other works, we keep this constant across all songs, rather than using beat-synchronous sampling, which is can be brittle on certain genres. Next, to promote temporal continuity when comparing windows, we stack delay overlapping blocks of 20 windows for each feature, as in [17]. Hence, each block spans roughly 4.64 seconds. We then compute Euclidean SSMs between the MFCC and tempogram blocks, and we compute SSMs based on the cosine distance between the Chroma and CREMA blocks. For a marginal improvement, we can enhance temporal continuity in a manner similar to [7] by performing a 9-tap median filter on each diagonal of the affinity matrices for each feature before applying SNF.

3. EVALUATION

Below, we describe the data and summary statistics we use. As a baseline algorithm, we compare to LSD [7] as implemented in MSAF [18].

3.1. Data

We use the SALAMI dataset [8] with multilevel annotation corrections [9] to quantiatively evaluate our algorithm. This dataset consists of 1,359 tracks across a wide variety of genres which each have at least one annotator who has marked "coarse" and "fine" segments. In our work, we focus on a subset of 884 songs which have two distinct annotators, so that we can compare to a human-level annotator agreement.

3.2. Evaluation criteria

Numerous methods have been proposed to evaluate the accuracy of musical structure estimation systems. For most choices of evaluation criteria (e.g., segment boundary accuracy or segment labeling), there are two critical sources of variation which must be accounted for: ambiguity in structural depth, and subjectivity across reference annotators.[2]

3.2.1. L-measures for hierarchical structure

The L-measure [9] was proposed as a generalization of pairwise frame classification [20] to support comparison between multi-level time-series segmentations, which we briefly summarize here. Multi-level segmentations are assumed to be provided as a sequence of collections of labeled intervals $H = (\Pi_0, \Pi_1, \ldots)$, where each Π_i partitions the input signal in time, and the sequence is ordered from *coarse* to *fine*. Typically, Π_0 is a single interval which spans the entirety of the input, and subsequent Π_i provide refinements into collections of smaller segments.

[2] All evaluations are implemented using `mir_eval` 0.5 [19]

If $\Pi_i(t)$ denotes the label of the interval containing time t at the i^{th} level of the segmentation, then a similarity between instants (t, u) can be derived from the maximum i such that $\Pi_i(t) = \Pi_i(u)$. This pairwise similarity function gives rise to a partial ordering over pairs of time instants, which can be summarized by the set of all triples (t, u, v) such that (t, u) are more similar than (t, v). Given two multi-level segmentations H^R (the reference) and H^E (the estimate), the L-precision (L-recall) is defined as the fraction of such triples in the estimate (reference) also found in the reference (estimate). The L-measure is defined as the harmonic mean of L-precision and L-recall.

As demonstrated in prior work, the L-measures facilitate holistic comparison between multi-level segmentations of differing depths, and are robust to level-alignment errors [9]. These properties make them well-suited to evaluating the ability of the proposed fusion method to capture multiple forms of structure in music. Note that because the estimators under comparison in this work all produce annotations of greater depth than the reference annotations—which all have two non-trivial levels—the precision scores may not be reliable. We therefore focus our evaluation on L-recall, which measures how much structure in the reference annotation was identified in the estimate. However, for completeness, we provide a full report of L-precision, L-recall, and L-measure.

3.2.2. *Inter-annotator agreement comparison*

Most evaluations of music structure analysis systems assume a single *ground truth* reference annotation for each track, compare the system's estimate to that reference, and summarize the distribution of evaluation scores over all tracks, e.g., by reporting the mean score. However, recent work has shown that multiple annotators often exhibit divergence of interpretation of musical structure, and this variation should be taken into account when evaluating systems.

We follow the design of [9], and compare the distributions of L-measures when comparing an estimator to *multiple* reference annotations to the distribution of scores arising from comparing the annotators to *each other*. Using the subset of the SALAMI collection for which we have multiple reference annotations (each containing multi-level segmentations), we compute the L-measure scores for each pair of annotations for each track, producing a sample of scores p_a. For each estimator e, we then compare each estimated structure to all annotations, which results in a second sample of scores p_e. The collections p_a and p_e are then compared using the two-sample Kolmogorov-Smirnov test statistic (K), which measures the maximum absolute difference between their (discrete) cumulative distribution functions: small values indicate similar distributions. This comparison measures the performance of the estimator relative to inter-annotator disagreement. For completeness, we also report the mean L-measure scores (across all tracks and annotators) to provide an absolute measure.

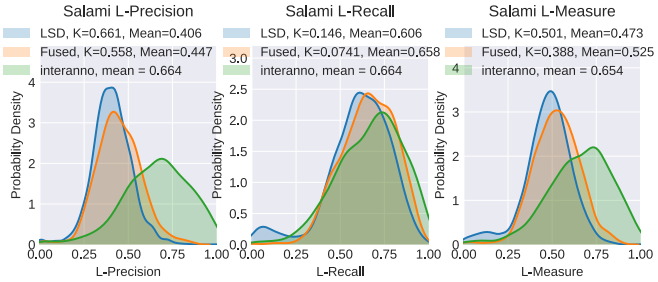

Fig. 3. Distributions of L-precision, L-recall, and L-measure for inter-annotator agreement, the spectral clustering technique of [7], and our fusion result. The Komolgorov-Smirinov statistic and mean score are shown in the legends.

	μ(P)	K(P)	μ(R)	K(R)	μ(L)	K(L)
Inter-Anno	0.664	—	0.664	—	0.654	—
MFCCs	0.371	0.663	0.295	0.617	0.283	0.713
Chromas	0.320	0.767	0.287	0.717	0.271	0.792
Tempogram	0.337	0.768	0.464	0.476	0.382	0.678
CREMA	0.392	0.668	0.529	0.342	0.441	0.558
Fused MFC/Chr	0.422	0.601	0.612	0.163	0.491	0.465
Fused Tgr/CRE	0.388	0.670	0.631	0.119	0.473	0.501
Fused $\kappa = 3$	**0.447**	**0.558**	**0.658**	**0.074**	**0.525**	**0.388**
Fused $\kappa = 10$	0.424	0.606	0.623	0.167	0.498	0.445
LSD[7]	0.406	0.661	0.606	0.146	0.473	0.501

Table 1. The means μ and K-scores of L-precision (P), L-recall (R), and L-measures (L) for different segmentations.

3.3. Results

Figure 3 shows probability density functions for L-precision, L-recall, and L-measure for our technique with a hierarchy of 2–10 clusters and $\kappa = 3$. The distributions for our fusion are closer to human level agreement than those of LSD [7], and they also correct a cluster of failure cases present in [7]. Table 3.3 shows the mean (higher is better) and K-scores (lower is better) of precision, recall, and L-measure for individual features and various combinations of fusion (MFCC/Chroma, Tempogram/CREMA, and all). In all cases, fusion improves over individual features, and fusing all features performs the best across all statistics. We also show that using a smaller number of neighbors κ for the spatial bandwidth is advantageous, as it tends to promote diagonal regions in the fused affinity matrices without connecting dissimilar blocks.

4. DISCUSSION / CONCLUSION

This work has shown promise of SNF + spectral clustering for hierarchical structure annotations, and we believe there will be other applications of feature-level SNF on affinity matrices in MIR. There is also room for general theoretical development of the interplay between SNF and the graph Laplacian.

5. REFERENCES

[1] Bo Wang, Jiayan Jiang, Wei Wang, Zhi-Hua Zhou, and Zhuowen Tu, "Unsupervised metric fusion by cross diffusion," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2997–3004.

[2] Bo Wang, Aziz M Mezlini, Feyyaz Demir, Marc Fiume, Zhuowen Tu, Michael Brudno, Benjamin Haibe-Kains, and Anna Goldenberg, "Similarity network fusion for aggregating data types on a genomic scale," *Nature methods*, vol. 11, no. 3, pp. 333, 2014.

[3] Ning Chen, "Ci-snf: Exploiting contextual information to improve snf based information retrieval," *Information Fusion*, 2018.

[4] Ning Chen, Wei Li, and Haidong Xiao, "Fusing similarity functions for cover song identification," *Multimedia Tools and Applications*, pp. 1–24, 2017.

[5] Christopher J Tralie, "Early mfcc and hpcp fusion for robust cover song identification," in *18th International Society for Music Information Retrieval Conference*, 2017.

[6] Christopher J Tralie, "Graphditty: A software suite for geometric music structure visualization," in *19th International Society for Music Information Retrieval (ISMIR), Late Breaking Session*, 2018.

[7] B. McFee and D. P. W. Ellis, "Analyzing song structure with spectral clustering," in *15th International Society for Music Information Retrieval Conference*, 2014, ISMIR.

[8] Jordan Bennett Louis Smith, John Ashley Burgoyne, Ichiro Fujinaga, David De Roure, and J Stephen Downie, "Design and creation of a large-scale database of structural annotations.," in *ISMIR*. Miami, FL, 2011, vol. 11, pp. 555–560.

[9] Brian McFee, Oriol Nieto, Morwaread M. Farbood, and Juan Pablo Bello, "Evaluating hierarchical structure in music annotations," *Frontiers in Psychology*, vol. 8, pp. 1337, 2017.

[10] Ulrike Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al., "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[12] Harald Grohganz, Michael Clausen, Nanzhu Jiang, and Meinard Müller, "Converting path structures into block structures using eigenvalue decompositions of self-similarity matrices.," in *14th International Society for Music Information Retrieval Conference*, 2013, ISMIR, pp. 209–214.

[13] Brian McFee, Matt McVicar, Stefan Balke, Carl Thomé, Vincent Lostanlen, Colin Raffel, Dana Lee, Oriol Nieto, Eric Battenberg, Dan Ellis, Ryuichi Yamamoto, Josh Moore, WZY, Rachel Bittner, Keunwoo Choi, Pius Friesch, Fabian-Robert Stöter, Matt Vollrath, Siddhartha Kumar, nehz, Simon Waloschek, Seth, Rimvydas Naktinis, Douglas Repetto, Curtis "Fjord" Hawthorne, CJ Carr, João Felipe Santos, JackieWu, Erik, and Adrian Holovaty, "librosa/librosa: 0.6.2," Aug. 2018.

[14] B. McFee and J.P. Bello, "Structured training for large-vocabulary chord recognition," in *18th International Society for Music Information Retrieval Conference*, 2017, ISMIR.

[15] Peter Grosche and Meinard Müller, "Tempogram toolbox: Matlab implementations for tempo and pulse analysis of music recordings," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR), Miami, FL, USA*, 2011, pp. 24–28.

[16] Sebastian Böck and Gerhard Widmer, "Maximum filter vibrato suppression for onset detection," in *Proc. of the 16th Int. Conf. on Digital Audio Effects (DAFx). Maynooth, Ireland (Sept 2013)*, 2013.

[17] Joan Serra, Xavier Serra, and Ralph G Andrzejak, "Cross recurrence quantification for cover song identification," *New Journal of Physics*, vol. 11, no. 9, pp. 093017, 2009.

[18] Oriol Nieto and Juan Pablo Bello, "Systematic exploration of computational music structure research.," in *ISMIR*, 2016, pp. 547–553.

[19] Colin Raffel, Brian McFee, Eric J. Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, and Daniel P. W. Ellis, "Mir_eval: A transparent implementation of common MIR metrics," in *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*, 2014, pp. 367–372.

[20] Mark Levy and Mark Sandler, "Structural segmentation of musical audio by constrained clustering," *IEEE transactions on audio, speech, and language processing*, vol. 16, no. 2, pp. 318–326, 2008.