

# A CROSS-SCAPE PLOT REPRESENTATION FOR VISUALIZING SYMBOLIC MELODIC SIMILARITY

Saebiyul Park      Taegyun Kwon      Jongpil Lee  
Jeounghoon Kim      Juhan Nam

Graduate School of Culture Technology, KAIST

{saebiyul\_park, ilcobo2, richter, miru, juhannam}@kaist.ac.kr

## ABSTRACT

Symbolic melodic similarity is based on measuring a pair-wise distance between two songs from diverse perspectives. The distance is usually summarized as a single value for song retrieval. This obscures observing the details of similarity patterns within the two songs. In this paper, we propose a cross-scape plot representation to visualize multi-scaled melody similarity between two symbolic music encodings. The cross-scape plot is computed by stacking up a minimum local distance between two segments from each of the two songs. As the layer goes up, the segment size increases and it computes incrementally more long-term distances. This hierarchical representation allows for capturing the location and length of similar segments between two songs in a visually intuitive manner. We show the effectiveness of the cross-scape plot by evaluating it on examples from folk music collections with similarity-based categories and plagiarism cases.

## 1. INTRODUCTION

Melodic similarity is a key concept in the field of ethnomusicology, music analysis, musicology, music psychology, copyright issues in music, and music information retrieval [13]. From music analysis to content-based retrieval, a great deal of effort and attention have been paid to quantitative measurement of melodic similarity using knowledge from various domains [5, 12, 13, 18–20, 22, 25]. The applications of melodic similarity include song retrieval and classification, music indexing, and music alignment systems [3].

When evaluating the similarity of two songs, it is essential to extract information about what parts and how long they are similar to each other. For example, considering a plagiarism case, a short three-second segment can be regarded as qualitatively significant if it can be easily recognized as a chorus or hook of a popular song [10]. If the similar parts of these two songs are meaningfully similar or *substantially similar* [9], even if the other parts of the two

songs are different, people may effectively recognize it as a plagiarized song [10]. However, existing similarity measures often summarize the distance as a single value for song retrieval. This obscures the details of similar pattern within the two songs. This way, even if the two songs has almost identical parts locally, the overall similarity may be diluted by the calculation for the entire song.

In this paper, we tackle this issue by proposing a cross-scape plot representation that visualizes multi-scaled melody similarity between two symbolic music. The cross-scape plot is computed by stacking up a minimum local distance between two segments from each of the two songs. We segment songs from small to large units, and the local similarity is performed by a sequence-based similarity algorithm for all possible segments of the two songs. As the layer goes up, the segment size increases and it computes progressively more long-term distances. This results in a hierarchical visual representation with a triangular or trapezoidal shape.

The cross-scape plot provides rich information in an intuitive way, which a single value derived from most similarity measures cannot provide. Even with a simple glance, we can observe various characteristics of similarity, such as the location and length of similar parts by the pattern shape in the plot, and the overall similarity by the color. Even when the melody has the same similarity value in a single measurement, the visual representation can be very different. We show this aspect with examples from classified folk songs Meertens Tune Collections (MTC-ANN) [21] and music plagiarism cases. We also validate the multi-scaled melodic similarity by summarizing it into a similarity value and conduct a melody classification experiment on MTC-ANN.

## 2. MULTI-SCALE SIMILARITY ANALYSIS

This section describes three steps of multi-scaled similarity analysis to obtain the cross-scape plot.

### 2.1 Feature Extraction

Many choices of features have been extracted for melodic similarity analysis including pitch, pitch interval, duration, onset, duration weighted pitch sequence or metric weights [7]. In this study, we use pitch interval as a pitch feature and inter-onset interval ratio as a rhythmic feature. Pitch interval is computed by the pitch difference between two



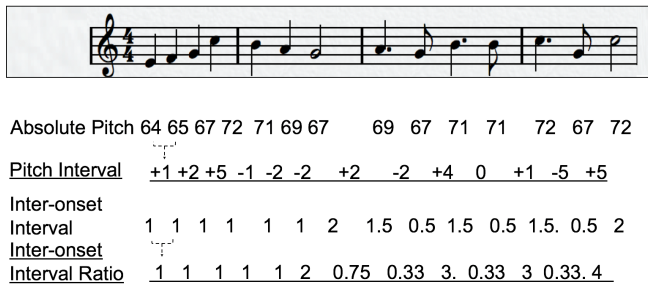


Figure 1: An example of pitch interval and inter-onset interval ratio of the rhythm extracted from melody.

successive notes. This is invariant to key of the songs. The pitch interval is limited to the range within 2 octaves in our setting. Inter-onset interval (IOI) ratio is computed from the relative IOI of three successive notes. This is invariant to tempo of the songs. The IOI is quantized to the unit of 0.5 and limited not to exceed 4 in our setting. Figure 1 shows that a melody is converted to a sequence of the pitch and rhythm features. We analyze similarity for both feature sequences independently.

### 2.2 Multi-segmentation

Multi-segmentation is the process of dividing the melody sequence into smaller sub-sequences of all sizes. This idea was inspired by Sapp’s work [15, 16]. Sapp proposed a plotting method called *scape plot* that can display the results of an analysis of segments of varying lengths as a single image. The *scape plot* is a simple but effective method to understand similarity patterns that occur on every possible timescale. The *scape plot* was named in *landscape* because it shows small-scale features similar to the foreground of the picture, as well as large-scale features similar to the background. The original *scape plot* method was designed for structural analysis of harmony in musical scores [15] and has been applied in a variety of ways in different contexts, for example, tonality analysis [14, 27], musical performance analysis [17, 24] and audio thumb-nailing [8].

Following the approach, we segment the sequence into multi-scale units from the smallest to the entire sequence. Figure 2 is an example of dividing the features sequences into the different sizes. The sequence *ABCDE* is divided into 15 sub-sequences by sequentially grouping them into units of the smallest unit (that correspond to segment size of 1) to the maximum of 5, which is the length of the entire sequence (that correspond to segment size of 5). Formally speaking, given the sequence of melody features,  $S = (x_1, x_2, \dots, x_{|S|})$  where  $|S|$  is the length of  $S$ , let  $s_k^n$  denote the  $k$ th sub-sequence with the length  $n$  of  $S$  so that  $s_k^n = (x_k, x_{k+1}, \dots, x_{k+n-1})$ ,  $1 \leq k \leq |S| - n + 1$ . These sub sequences  $s^n$  are used to obtain the local distance between two songs. The following section describes how to calculate the local similarity using these multiple sub-sequences of melody.

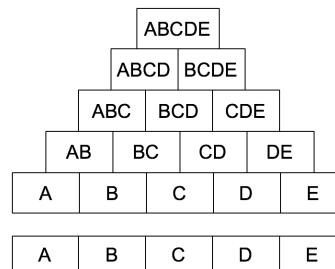


Figure 2: Example of multiple segments of melody. The sequence *ABCDE* is divided into 15 sub-sequences.

### 2.3 Similarity Calculation

We use the multi-segmentation to compute multi-scale similarity between two songs. We first define the distance measure between two segments.

#### 2.3.1 Distance Measure

In order to obtain the local distance of each segment, we adopt edit distance [11], the most commonly used string matching similarity calculation method in music research [5]. The edit distance, also known as the Levenshtein distance, is a metric that computes the minimum number of operations needed to transform one sequence into the other.

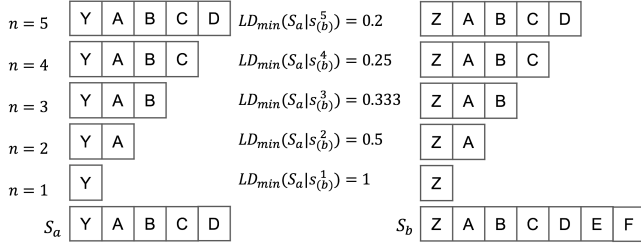
The operations between sequences include *deletion*, *insertion*, and *substitution* of symbols. To find out the minimum path to obtain edit distance, we use dynamic programming algorithm known as Wagner—Fischer algorithm. For the compared melodic sequence  $S_a = (a_1, a_2, \dots, a_n)$ , and the  $S_b = (b_1, b_2, \dots, b_m)$ , where  $a_i$  and  $b_j$  are features of the melody.<sup>1</sup> We set 1 to the cost of *deletion*, *insertion* and 2 to the cost of *substitution*. Let  $d(i, j)$  denote the edit distance of sub-sequence  $(a_1, a_2, \dots, a_i), (b_1, b_2, \dots, b_j)$ . The calculation of edit distance  $D(S_a, S_b) = d(n, m)$  is defined using a recursive algorithm as below [6]:

$$\begin{aligned}
 d(i, 0) &= i, \\
 d(0, j) &= j, \\
 d(i, j) &= \begin{cases} d(i-1, j-1) & \text{if } a_i = b_j \\ \min \begin{cases} d(i-1, j) + 1 \\ d(i, j-1) + 1 \\ d(i-1, j-1) + 2 \end{cases} & \text{if } a_i \neq b_j \end{cases} \\
 &\quad \text{for } 1 \leq i \leq n, 1 \leq j \leq m
 \end{aligned} \tag{1}$$

The final local distance between two sub-sequences, denoted by  $\bar{D}$ , is the normalized value of  $D(S_a, S_b)$  divided by the maximum possible distance (i.e., when the two sequences are completely different) as follows:

$$\bar{D}(S_a, S_b) = \frac{D(S_a, S_b)}{(|S_a| + |S_b|) \times 2} \tag{2}$$

<sup>1</sup>  $S$  can be either a whole melody, or a sub-sequence of it, and it can be pitch interval and the inter-onset interval ratio.



**Figure 3:** An example of minimum local distance calculation between  $S_a$  and  $S_b$

We particularly define the minimum local distance between a melody  $S_a$  and  $s_{k(b)}^n$ , a sub-sequence of  $S_b$  at the  $k$ th position with length  $n$ :

$$LD_{min}(S_a|s_{k(b)}^n) = \min_{\forall s_{(a)}^n \in S_a} \overline{D}(s_{(a)}^n, s_{k(b)}^n) \quad (3)$$

Because  $LD_{min}(S_a|s_{k(b)}^n)$  is a minimum distance between a segment  $s_{k(b)}^n$  and all possible sub-sequences of  $S_a$  with the same length  $n$ , it is not a commutative operation. However, we call it distance because it shows the distance between  $s_{k(b)}^n$  and most-matched segment in  $S_a$ . For example, suppose that there are two melody sequences to be compared,  $S_b$  is a longer melodic sequence, and  $S_a$  is shorter (see Figure 3).  $S_a$  and  $S_b$  can be divided into sub-sequences.<sup>2</sup> The figure shows the  $LD_{min}$  value of the first segment for each  $n$ . For example,  $LD_{min}(S_a|s_{1(b)}^1)$  is 1 because in the first sub-sequence with  $n = 1$  (“Z” in the example) does not match any sub-sequences of  $S_a$ . As the number of notes shared by  $S_a$  and  $S_b$  increases,  $LD_{min}(S_a|s_{k(b)}^n)$  becomes smaller, that is, the degree of similarity increases. Note that since the lengths of two melodies are different, the maximum size of the  $n$  is the length of  $S_a$  (the shorter).

### 2.3.2 Multi-scale Similarity Stack

After the operation for all segments is performed, it forms a multi-scale similarity stack for  $S_a$  and  $S_b$ . The multi-scale similarity stack is important for similarity analysis, and also for *cross-scape plot* visualization (see Figure 5) which will be discussed in Section 3. In this study, we obtain a multi-scale similarity stack based on segments of longer songs.<sup>3</sup> Thus, the maximum of the x-axis of the multi-scale similarity stack becomes the length of the longer song, while the maximum of the y-axis is the length of the shorter melody, respectively. For a given pair of melodic sequences  $S_a$  and  $S_b$ , with  $|S_a| \leq |S_b|$  we can calculate  $LD_{min}(S_a|s_{k(b)}^n)$  for all sub-sequences with length  $n$  for all possible  $n$  in  $S_b$  against  $S_a$ . As a result, we create a two-dimensional multi-scale similarity stack (MSS) defined as below<sup>4</sup>:

<sup>2</sup> For readability, features are noted as letters.

<sup>3</sup> The opposite case, comparing segments of the song based on shorter songs, is also possible, but this is not covered in this study.

<sup>4</sup> Elements of the matrix only valid for  $1 \leq n \leq |S_a|, 1 \leq k \leq |S_b| - n + 1$

$$MSS(S_a|S_b) = [a_{n,k}]_{|S_a| \times |S_b|}, \quad (4)$$

$$a_{n,k} = 1 - LD_{min}(S_a|s_{k(b)}^n)$$

We can also summarize  $MSS$  and calculate the overall similarity as a single value. In this case, we discard a subset of elements in  $MSS$  which corresponds to similarity between segments less than three notes. To compensate low similarity values between longer sequence, we also define a weighted multi-scale similarity stack,  $wMSS$ , where the weight  $\lambda$  is a function of  $n$ . Finally, the overall similarity  $Sim_{S_a, S_b}$  of two songs is obtained by averaging  $wMSS$ . This overall similarity is derived in the same way for both pitch interval and inter-onset interval ratio of rhythm.

$$wMSS(S_a|S_b) = [\lambda(n) \times a_{n,k}]_{|S_a| \times |S_b|}, \quad (5)$$

$$a_{n,k} = 1 - LD_{min}(S_a|s_{k(b)}^n),$$

$$\lambda(n) = 0.5 + 0.5 \times n/|S_a|$$

Finally, the overall similarity  $Sim(S_a, S_b)$  of two songs is obtained by averaging the weighted multi-scale similarity stack:

$$Sim(S_a, S_b) = \text{mean}((wMSS(S_a|S_b))_{n,k} n \geq 3) \quad (6)$$

This overall similarity can be computed in the same way for both pitch and rhythm features but using different weights. We implemented the algorithm using MATLAB and MIDI Toolbox [4]. The source code is available at the Github repository.<sup>5</sup>

## 3. CROSS-SCAPE PLOT

This section introduces the visualization method using the multi-scaled similarity of two melody sequences. As aforementioned, this method was inspired by [15, 16]. While previous studies using the original scape-plot method focused on analyzing a single song based on self-similarity, this study applies the idea to analyzing the similarity of two different songs.

### 3.1 Procedure

Figure 4 illustrates the procedure of drawing the cross-scape plot using the multi-scale similarity stack. We arrange the sub-sequences on top of each other to have the hierarchical patterns. Also, to increase readability, the multi-scale similarity stack is center-aligned. Unlike the original scape plot, the cross-scape plot has a trapezoidal shape because it is computed from two songs with different lengths. The more different the lengths of the two songs, the more likely the shape of the trapezoid. The x-axis indicates the position of the longer melody. The y-axis represents the segment size.

The cross-scape plot displays local similarity with a color. As the local similarity is higher, the color becomes darker, and vice versa. To distinguish the two features, we

<sup>5</sup> [https://github.com/saebuyulpark/cross\\_scapeplot\\_visualization](https://github.com/saebuyulpark/cross_scapeplot_visualization)

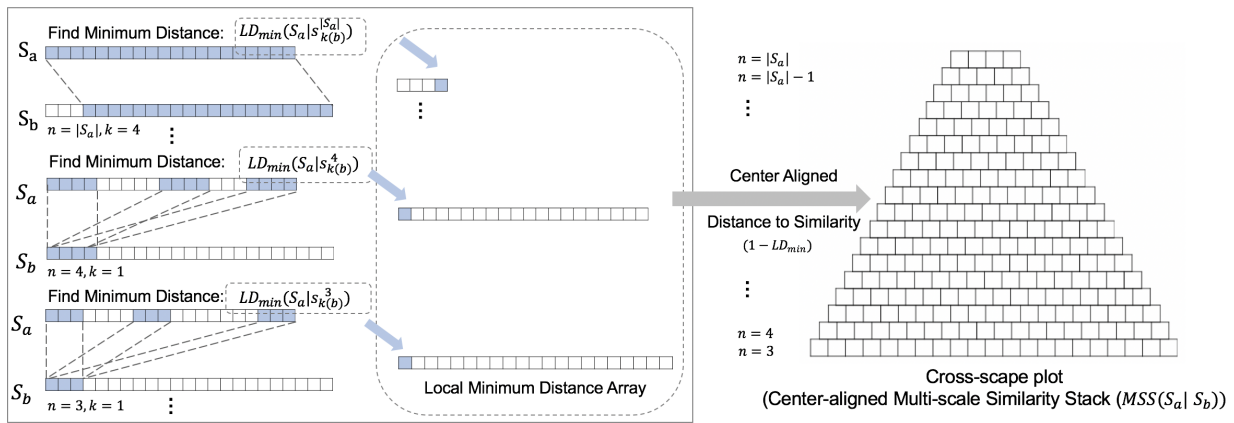


Figure 4: An illustration of steps for drawing a cross-scope plot.

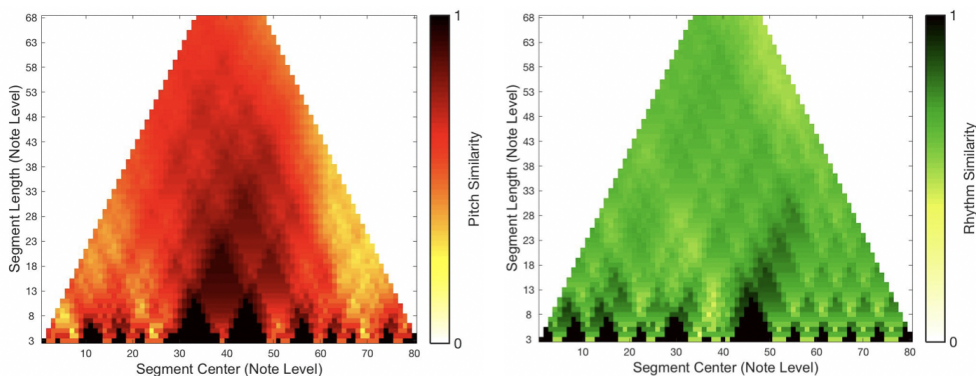


Figure 5: An example of a cross-scope plot by comparing similarities between two songs. The pixel value and the density of the color indicate the similarity score between the two sub-sequences.

set red to the cross-scope plot of the pitch feature and green to that of the rhythm feature. These different colors allows to easily recognize where and how the two songs are similar simply inspecting the plot. Figure 5 is an example of cross-scope plots where two songs are taken from the same tune family in the MTC-ANN datasets. The left part is the plot for the pitch feature and the right is for the rhythm feature. Through the cross-scope plot on the left, we can see that these two pieces are very similar in the middle parts. in the middle parts of the pitch feature, and the uppermost color suggests that the overall similarity is between 0.6 and 0.7. On the right plot, the rhythm feature shows a similarity with a certain periodicity in the latter part. Also, the slightly right part of the center represents a high similarity (black or 1) in the same way as the pitch feature, so we can assume that the melody of a particular part is nearly identical. Overall, a type of tendency is found where a large segment has a larger distance and a smaller segment has a smaller local distance. This is because the smaller the segment size is less distinct, the higher the probability that the segment is present in both songs.

### 3.2 Case Study: Toy Example

Figure 6 is a toy example of the cross-scope plot showing the similarity between the melodies generated such that they have the same edit distance between pitch interval sequences. One pair has exactly matching at a specific part

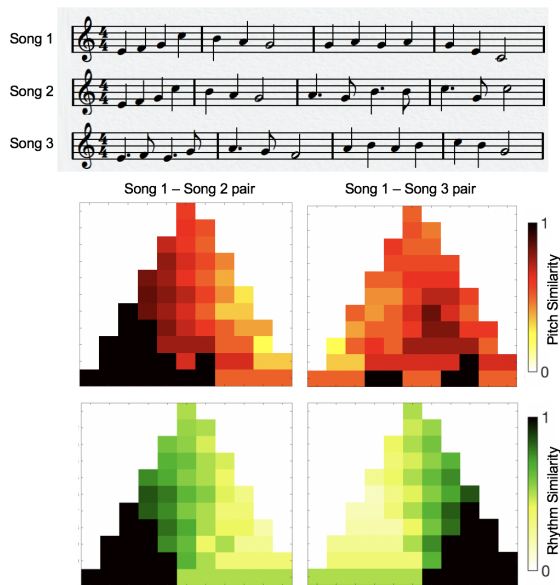
of the melody (Song 1 and 2, left part of Figure 6). In the other pair, the specific parts are not exactly the same, but they are slightly similar overall (Song 1 and 3, right part of Figure 6). In this example, the edit distance alone yields the same distance between the melodies (in both cases  $D = 12$  for pitch interval sequences). However, we can see that the first half of the song is exactly the same for both pitch and rhythm in the case of songs 1 and 2. On the other hand, the pitch features of Songs 1 and 3 are generally similar at random, but the rhythm is apparently similar to the latter half of the song. In this way, we can see various perspectives on the similarity of melody features even though they have the same similarity value.

### 3.3 Case Study: Plagiarism Cases

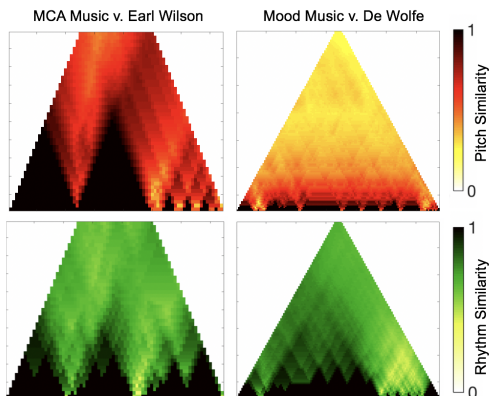
The benefits of cross-scope plots become more important for issues that require a certain level of qualitative judgment, such as originality or substantial similarity of plagiarism. Figure 7 shows cross-scope plots of the cases in our ongoing plagiarism research project. These two examples are cross-scope plots of songs with court rulings on copyright infringement. In both cases (*Mood Music v. De Wolfe*<sup>6</sup>: left side of figure and *MCA Music v. Earl Wilson*<sup>7</sup>: right side of figure), copyright infringement is ruled by the court or settled between the parties, where plagia-

<sup>6</sup> *Mood Music v. De Wolfe*, 1. Ch. 119 (1976).

<sup>7</sup> *MCA Music v. Earl Wilson* 425 F. Supp. 443 (S.D.N.Y. 1976)



**Figure 6:** Toy example of an exact match pair and an overall similar pair of cross-scope plots



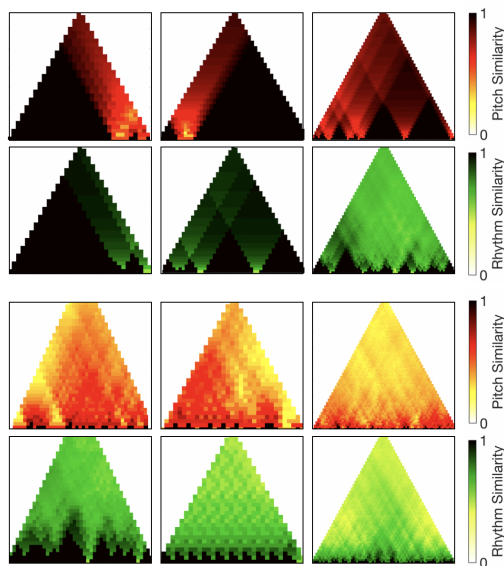
**Figure 7:** Cross-scope plots of plagiarism cases (*Mood Music v. De Wolfe*: top of the figure and *MCA Music v. Earl Wilson*: bottom of the figure).

rism is acknowledged. We show these cases because these are either the most confusing (case 1 where 93.5% of the participants were confused) or less confused (case 2 with 0% confusion rate) cases in a trial experiment in which the implicit memory task is performed (High confusion means two songs are very similar to each other).<sup>8</sup>

Although both are cases of copyright infringement, the ways in which similarities appear are very different. In *Music v. De Wolfe*, most parts of the song are almost identical on all sides of the pitch and rhythm features. On the other hand, in *MCA Music v. Earl Wilson*, the rhythm features are generally similar, but the pitch features are different. Thereby, for the case *MCA Music v. Earl Wilson*, it can be assumed that the plagiarism judgment is based on other factors such as rhythm, harmony, arrangement or lyrics rather than the tonal characteristics of the melody.<sup>9</sup>

<sup>8</sup> This experiment is an ongoing project; a similar experiment and result can be found in [28]

<sup>9</sup> Indeed, in spite of the defendant’s arguments contending fair use



**Figure 8:** Cross-scope plots on examples of MTC-ANN. The three pairs on the top belong to the same tune family whereas the three on the bottom belong to different tune families.

In this way, gaining rich information about similarities can be of great help in making this kind of intuitive and sophisticated decision regarding of similarity.

### 3.4 Case Study: MTC-ANN

Figure 8 shows more examples of cross-scope plots showing how different pairs of MTC-ANN represent similarities of different characteristics. The three pairs on the top are similarities between songs in the same tune family, the three on the bottom are for those included in other tune families. The similarity of the two songs decreases from left to right. Looking at these pairs, we can observe that similarities in pitch, rhythm, position, and lengths of similar parts appear in a variety of ways, even in pairs in the same group.

## 4. MELODY CLASSIFICATION

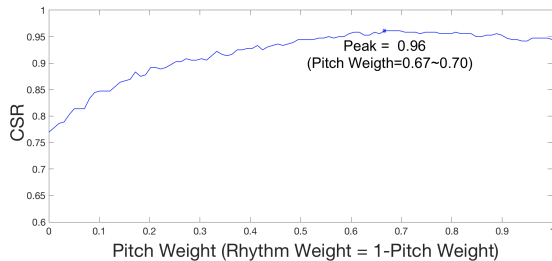
We can utilize the outcome of cross-scope plots for the symbolic melodic similarity task. In this section, we conduct the task with a classification experiment using summarized similarity derived from the multi-scale similarity stack.

### 4.1 The dataset

We used the annotated corpus of the Meertens Tune Collection (MTC-ANN), version 2.0.1 [21] for the evaluation. It contains 360 melodies divided into 26 tune families annotated by musicological experts. The MTC-ANN dataset has been used in a variety of music studies. This allowed us to compare our model with recent studies of measuring melodic similarity.

of parody, infringement was considered based on identical bass line, a general harmonic similarity and certain specialized rhythmic patterns (<https://blogs.law.gwu.edu/mcir/case/mca-music-v-earl-wilson/>).

	CSR	AUC	MAP
Pitch Interval	0.95	0.89	0.68
Inter-onset Interval Ratio	0.76	0.82	0.49
Combined	0.96	0.91	0.71

**Table 1:** Results for the proposed model

**Figure 9:** Accuracy that varies with the weighting value of the pitch feature and the rhythm feature.

## 4.2 Similarity Measures

We performed the classification of MTC-ANN songs into the same tune family with each of pitch and rhythm features or their combinations. We compared the performance to those from previous work that conducted the same task over the last three years [1, 2, 26].

## 4.3 Evaluation Metrics

We used the three evaluation metrics following the previous works.

**Classification Success Rate (CSR)** represents a correctly classified rate of the total when the melody is indexed into the same tune family using k-Nearest Neighbors (k-NN) classifier to which the melody with the highest similarity belongs. This evaluation method was used in all reference papers to be compared.

**Area Under the Curve (AUC)** is calculated by adaptively modifying the decision threshold of the similarity score. In this case, the songs within the same tune family is assigned to a ground truth. The ranking scores are then calculated and averaged.

**Mean Average Precision (MAP):** has the same ground truth as the one used in the AUC calculation. The songs are sorted by the similarity score and the songs that have the same tune family with the query song are treated as a correct one. We repeat this procedure by adjusting the number of correct answers and averaged the whole scores.

## 4.4 Evaluation Results

Table 1 shows the results of the proposed model to classify 360 songs of MTC-ANN into 26 family tunes, using pitch features, rhythm features and both of them. In addition, Figure 9 shows the plot of CSR when the relative weight in computing the overall similarity ( $\lambda$  in Equation 6) between two features changes from 0 to 1. the highest score is obtained when the weight (pitch weight) ranges between 0.67 and 0.7.

	CSR	AUC	MAP
Boot [1]	0.92	-	-
Bountouridis [2]	0.94	-	0.70
Walshaw [26]	0.93	0.89	-
Proposed	0.96	0.91	0.71

**Table 2:** Comparison with previous studies using the MTC-ANN dataset

As a result of the classification evaluation, the melody of a family group is sufficiently classified by a single pitch feature, whereas a rhythm feature has a limitation in performing a classification task alone. In addition, classification with pitch and rhythmic features combined with appropriate weights has achieved the highest performance, which indicates that pitch and rhythm together complement the similarity of the melody, although the pitch feature contributes to it more.

## 4.5 Comparison with Recent Studies

Table 2 provides a comparison with recent studies that performed classification of MTC-ANN using the entire melody. The results show that our approach not only provides the visualization but also comparable performance to them.

## 5. DISCUSSION AND CONCLUSION

We proposed a cross-scape plot which visualizes symbolic melodic similarity between two songs based on multi-segmentation analysis. With the examples of folk songs and plagiarism cases, we showed that the cross-scape plot can reveal the melodic similarity in various ways. We also performed a classification task of the MTC-ANN dataset based on the summarized similarity derived from the method.

The proposed method used edit distance but this can be replaced with with other sequence-based distances. On the other hand, however, it has already been shown that many studies yield high classification results when sequence alignment techniques are used as a measure of similarity between folk songs [23]. Thus, the high classification performance in this study may reflect the advantages of the alignment method used as the main distance. Therefore, the performance evaluation of the present method as a measure of similarity requires further study to experiment with more data sets and distance measures. Besides, since only segments with the same length are compared and are performed based only on shorter melodies, there is a limitation that some loss of information can occur. There is also a disadvantage that this method is time-consuming because it repeatedly performs calculations among all the segments. Despite several limitations, we believe that this multi-scaled approach provides a wealth of insight that will help us to understand the properties of similarity.

## 6. REFERENCES

- [1] Peter Boot, Anja Volk, and W Bas de Haas. Evaluating the role of repeated patterns in folk song classification and compression. *Journal of New Music Research*, 45(3):223–238, 2016.
- [2] Dimitrios Bountouridis, Daniel Brown, Frans Wiering, and Remco Veltkamp. Melodic similarity and applications using biologically-inspired techniques. *Applied Sciences*, 7(12):1242, 2017.
- [3] J Stephen Downie. Music information retrieval. *Annual Review of Information Science and Technology*, 37(1):295–340, 2003.
- [4] Tuomas Eerola and Petri Toiviainen. *MIDI Toolbox: MATLAB Tools for Music Research*. University of Jyväskylä, Jyväskylä, Finland, 2004.
- [5] Kuldeep Gurjar and Yang-Sae Moon. Comparative analysis of music similarity measures in music information retrieval systems. *Journal of Information Processing Systems*, 14(1), 2018.
- [6] Dan Gusfield. *Algorithms on strings, trees, and sequences: computer science and computational biology*. Cambridge university press, 1997.
- [7] Berit Janssen, Peter van Kranenburg, and Anja Volk. A comparison of symbolic similarity measures for finding occurrences of melodic segments. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, pages 659–665, 2015.
- [8] Nanzhu Jiang and Meinard Müller. Towards efficient audio thumbnailing. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5192–5196, 2014.
- [9] Stephanie J Jones. Music copyright in theory and practice: An improved approach for determining substantial similarity. *Duquesne Law Review*, 31:277, 1992.
- [10] Jeffrey F Kersting. Singing a different tune: Was the sixth circuit justified in changing the protection of sound recordings in bridgeport music, inc. v. dimension films. *University of Cincinnati Law Review*, 74:663, 2005.
- [11] Vladimir I Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710, 1966.
- [12] Daniel Müllensiefen and Marc Pendzich. Court decisions on music plagiarism and the predictive value of similarity algorithms. *Musicae Scientiae*, 13(1\_suppl):257–295, 2009.
- [13] Daniel Müllensiefen and Klaus Frieler. Cognitive adequacy in the measurement of melodic similarity: Algorithmic vs. human judgments. *Computing in Musicology*, 13(2003):147–176, 2004.
- [14] Meinard Müller and Nanzhu Jiang. A scape plot representation for visualizing repetitive structures of music recordings. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, pages 97–102, 2012.
- [15] Craig Stuart Sapp. Harmonic visualizations of tonal music. In *International Computer Music Conference (ICMC)*, volume 1, pages 419–422, 2001.
- [16] Craig Stuart Sapp. Visual hierarchical key analysis. *Computers in Entertainment (CIE)*, 3(4):1–19, 2005.
- [17] Craig Stuart Sapp. Comparative analysis of multiple musical performances. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, pages 497–500, 2007.
- [18] Patrick E Savage and Quentin D Atkinson. Automatic tune family identification by musical sequence alignment. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, pages 162–168, 2015.
- [19] Patrick E Savage, Charles Cronin, and Daniel Müllensiefen Quentin D Atkinson. Quantitative evaluation of music copyright infringement. In *Proceedings of the 8th International Workshop on Folk Music Analysis (FMA2018)*, pages 61–66, 2018.
- [20] Rainer Typke. Music retrieval based on melodic similarity. *Ph.D thesis*, 2007.
- [21] Peter van Kranenburg, Berit Janssen, and Anja Volk. The meertens tune collections: The annotated corpus (MTC-ANN) versions 1.1 and 2.0.1, 2016.
- [22] Peter van Kranenburg, Anja Volk, and Frans Wiering. A comparison between global and local features for computational classification of folk song melodies. *Journal of New Music Research*, 42(1):1–18, 2013.
- [23] Peter van Kranenburg, Anja Volk, Frans Wiering, and Remco C Veltkamp. Musical models for folk-song melody alignment. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, pages 507–512, 2009.
- [24] Carlos Vaquero. A quantitative study of seven historically informed performances of Bach’s BWV1007 prelude. *Early Music*, 43(4):611–622, 2015.
- [25] Valerio Velardo, Mauro Vallati, and Steven Jan. Symbolic melodic similarity: State of the art and future challenges. *Computer Music Journal*, 40(2):70–83, 2016.
- [26] C. Walshaw. Tune classification using multilevel recursive local alignment algorithms. In *Proc. 7th Intl Workshop on Folk Music Analysis (FMA)*, pages 80–87, 2017.
- [27] Christof Weiß and Meinard Müller. Quantifying and visualizing tonal complexity. In *Conference on Interdisciplinary Musicology (CIM)*, 2014.

- [28] Anna Wolf and Daniel Müllensiefen. The perception of similarity in court cases of melodic plagiarism and a review of measures of melodic similarity. In *Int. Conf. of Students of Systematic Musicology (SysMus)*, 2011.