COMPARING REPETITION-BASED MELODY SEGMENTATION MODELS

Marcelo E. Rodríguez López, Anja Volk, W. Bas de Haas

Department of Information and Computing Sciences, Utrecht University, Netherlands

Correspondence should be addressed to: M.E.RodriguezLopez@uu.nl

Abstract: **This paper reports on a comparative study of computational melody segmentation models based on repetition detection. For the comparison we implemented five repetition-based segmentation models, and subsequently evaluated their capacity to automatically find melodic phrase boundaries in a corpus of 200 folk melodies. We systematically investigate the effects that the choice of melodic representation, similarity measure, and parameter settings have on each model's performances. We discuss at length issues such as parameter sensitivity, generalization capability, and efficiency. The best performing model employs a similarity matrix to identify repetitions, and selects which repetitions are used to segment the input melody using an optimisation-based search algorithm.**

1. INTRODUCTION

Melody segmentation refers to the identification of structural units within a melody, i.e. the perceptual capacity of partitioning a melody into its constituent parts. Computational models of melody segmentation attempt to mimic this perceptual capacity. Computational modelling of melody segmentation is considered important for fields like Music Cognition (to test segmentation theories), Music Information Research (for tasks such as automatic music archiving, retrieval, and visualisation), and Computational Musicology (for automatic or human-assisted music analysis).

Computer models often simplify the task of segmenting melodies to that of detecting the boundaries between segments, i.e. the time points separating two contiguous segments. In this paper we focus on boundary detection of segments resembling music-theoretic phrases. That is, the boundaries of melodic units ranging from roughly 4-5 note events to 4-8 bars.

A factor often considered fundamental for human listeners to perceive phrase boundaries in melodies is the (exact or approximate) repetition of melodic fragments (cells, figures, and in cases whole phrases or larger fragments) [1, 2, 3, 4]. However, only a handful of computer models for phrase-level segmentation based on repetition detection have been published. Moreover, their performance has not been systematically studied, limiting further development in the field. To address this issue, we implement five different segmentation models based on repetition detection [3, 5, 6, 7, 8], and test their ability to identify phrase boundaries on 100 instrumental folk melodies and 100 vocal folk melodies. We systematically investigate the effects that the choice of melodic representation, similarity measure, and parameter settings have on each model's performances. We also discuss at length other issues such as parameter sensitivity, generalization capability, and efficiency. Under the scope of this study, the best performing model is [8], followed by [5] and [7].

The rest of the document is organised as follows: In §2 we describe the processing chain of repetition-based segmentation models, in §3 we present a short description of the compared models, in §4 we present and discuss the performances obtained by the models, and finally in §5 we summarise conclusions and outline future work.

2. PROCESSING CHAIN OF REPETITION-BASED SEGMENTATION MODELS

The segmentation models compared in this study have very diverse functionality. Hence, in this section we identify and describe shared processing stages to aid with their description in §3.

2.1. Aim and Methodology

The main aim of repetition-based segmentation models is to automatically detect segment boundaries that are cued by repetition.[1] The methodology to do so is (a) *identify repetitions*, i.e. locate all melodic fragments that are significantly similar,[2] (b) *select repetitions*, i.e. keep only perceptually salient repetitions,[3] and (c) *select boundaries*, i.e. use start and/or end locations of significantly similar and salient fragments as boundary positions.

Figure 1 shows a processing chain that implements the methodology described above. This chain has been adopted by the models compared in this study and is described in the following section.

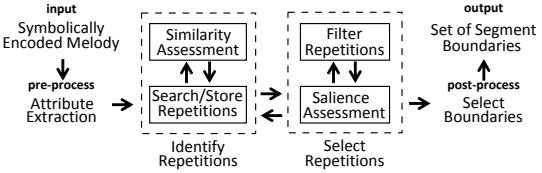

Figure 1: Processing chain of repetition-based melody segmentation models.

2.2. Processing Chain Description

The processing chain shown in Figure 1 consists of two main processing modules, as well as pre/post processing stages. The two main modules identify and select repetitions, respectively. These two modules are interdependent and rely on sub-modules performing search/storage, similarity assessment, salience assessment, and repetition filtering. Below we describe the processing chain components.

Input/Output: The models compared take as input a symbolic representation of the melody, i.e. a sequence of temporally ordered symbols approximating note-like musical events. Each symbol in the sequence represents either a note's chromatic pitch or its quantized duration (or the combination of the two). The output is a list of note event locations that correspond to segment boundaries.[4]

Attribute Extraction: Segmentation models often compute attributes deemed perceptually more salient than the {chromatic pitch, quantized duration} representation of note-like melodic events. One of the aims of this study is to investigate how the choice of melodic representation might influence repetition-based segmentation. Hence, in our comparative study we evaluate the performance of segmentation models using various melodic

[1] In this study we use 'cue' to refer to the musical factors that affect segment perception, and 'repetition' to refer to the identification of a melodic fragment as an instance of a melodic fragment heard elsewhere in the melody. Also, we a take the meaning of 'melodic fragment' in its broadest sense: any sequence of contiguous melodic events heard while listening to the melody. The models compared assume that the melody is mentally represented as a sequence of perceptually discrete events similar to music-theoretic notes, and hence melodic fragments are assumed perceived as a concatenation of these events.

[2] A 'significantly similar' set of fragments is that in which a human listener would perceive all fragments of the set to be equivalent (i.e. fragments of a set are heard as repetitions of the first occurring fragment).

[3] We consider repetitions to be 'perceptually salient' if they are relevant for segment boundary perception.

[4] Whether the boundaries correspond to segment starts or ends depends on the output configuration of the model.

representations, considering absolute, relative, and linked (i.e. tuples of) attributes. The attributes studied are listed in Table 1.[5]

Table 1: Melodic attributes and similarity measures tested.

Attribute Representation		Similarity Measures	
Abbreviation	Meaning	Abbreviation	Meaning
`cp`	chromatic pitch	`ham`	Hamming
`cp-cs`	chromatic pitch class	`lcs`	Largest Common Sub.
`cp-iv`	chromatic pitch interval	`lev`	Levenshtein
`sli`	step-leap pitch interval	`jac`	Jaccard
`ioi`	inter-onset-interval	`dic`	Dice
`ior`	`ioi` ratio	`cos`	Cosine
`pXi`	pitch and `ioi`	`euc`	Euclidean
`iXr`	`cp-iv` and `ior`	`cit`	City Block

Search/Store Repetitions: To identify repetitions the models compared employ the following methodology: (a) select a given fragment of the melody, (b) search for significantly *similar* instances (i.e. repetitions) of that fragment, and (c) store and organise identified repetitions of the fragment. Formally, the aim is to locate repetitions of sub-sequences of the type $x_{i...j} = x_i,...,x_j$ from a given melodic attribute sequence $x = x_1,\ldots,x_i,\ldots,x_N$ of length N, with $i,j \in [1:N]$. A repeated pair occurs when $x_{i...i+l-1} \sim x_{j...j+l-1}$, where $\sim$ denotes the pair is significantly similar, $i \neq j$, and l is an integer indicating the length of the repeated pair. Efficient search and storage of repetitions in symbolic sequences is commonly tackled using data structures. The models compared in this paper use suffix trees (`ST`), similarity matrices (`SM`), and hash tables (`HT`) as data structures. The way the first two data structures (`ST` and `SM`) organise identified repetitions is exploited by segmentation models to select relevant repetitions, making them a fundamental part of the models they are used in. Conversely, the models employing `HTs` use the data structure only for efficiency reasons, and thus it could in principle be replaced by any other. In §2.3 we provide a brief description of `ST` and `SM`.

Similarity Assessment: When searching for repetitions of a melodic fragment, the models compared use one of two strategies: (a) they simplify the problem of repetition detection to that of locating exact matches, i.e. cases where there is an exact match of the attributes representing the melodic fragments compared, or (b) they allow repetitions to be approximate matches, i.e. cases where there is an approximate match of the attributes representing the melodic fragments compared. One of the aims of this study is to investigate how the choice of similarity measure might influence repetition-based melodic segmentation. Hence, for models detecting approximate-match repetitions, we study the performance of segmentation models using various standard similarity measures (listed in Table 1). We can classify the approximate match similarity measures into string metrics (`ham`, `lcs`, `lev`), statistical (`jac`, `dic`), and geometric (`cos`, `euc`, `cit`). When assessing similarity using string metrics, the two fragments compared are represented as attribute sequences (i.e. $x_{i...i+l-1}, x_{j...j+l-1}$). Conversely, when assessing similarity using geometric and statistical measures the two fragments compared are represented using a vector space.[6]

Salience Assessment of Identified Repetitions: Computer models are likely to identify many repetitions, yet the number of *perceived* repetitions is generally much smaller [9]. Moreover, the number of repetitions that are relevant for boundary perception is suspected to be even smaller [3, 10]. Thus, repetition-based segmentation models prune the space of identified repetitions using salience assessment and filtering to select only those that are relevant for boundary detection. The models compared in this study use one or more of the following factors to assign salience:

- repetition **length** (`L`)
- repetition **frequency** (`F`)
- amount of **temporal overlap** between repetitions (`TO`)
- repetition **position** in the melody (`P`)
- repetition start/end coincides with **temporal gaps** (`TG`)[7]

The motivation to use these factors is that repetitions that are longer `L`, frequent `F`, with low temporal overlap `TO`, and that occur earlier in the melody `P` or after a large temporal gap `TG` are hypothesized to be perceptually salient. The first three factors are widely used for assigning relevance to repetitions in text and biological symbolic sequences, and are hence domain independent. Conversely, the last two are, to the best of our knowledge, used mainly in music and are hence domain dependent.

Filtering of Identified Repetitions: The models compared in this paper use the factors listed above as heuristics to filter out non salient repetitions. The heuristics are implemented in either parametric or non-parametric form. Parametric form refers to cases where the user has direct access to control the selection factors by means of input parameters (models [3, 7]). Conversely, non-parametric form refers to cases where the user has only indirect or no access to control over the selection factors (models [6, 8, 5]). The models compared also employ naïve filtering strategies, e.g. user defined minimum/maximum length of repetitions. We describe filtering heuristics and naïve filtering strategies in more detail in §3.

2.3. Data Structures for Search/Storage

This section provides a brief introduction to data structures used by segmentation models for the identification and storage of repetitions. We describe only data structuress integral to the models they are used in (i.e. suffix trees `ST` and similarity matrices `SM`).

Suffix Trees: A suffix of a symbolic sequence is a sub-sequence that includes the last symbol of the sequence. A suffix tree is a data structure that represents all suffixes of a sequence as a tree structure, i.e. a tree where in each branch the concatenation of the edge labels (from root to leaf) spells out a suffix. Figure 2 shows a suffix tree for the pitch class sequence *ababc*. For further reading we refer to [11].

Similarity Matrices: For a given symbolic sequence of length N, a `SM` corresponds to the matrix of pairwise (dis)similarities $S = [a_{ij}]_{N\times N}$ between subsequences $a_{ij} = sim(x_{i...i+l-1}, x_{j...j+l-1})$, where l denotes the length of the subsequence. In a `SM` repetitions result in diagonal stripes parallel to the main diagonal. Figure 2 shows a `SM` (taking $l = 1$), where the repetition *ab* of the pitch class sequence *ababc* is depicted as a diagonal stripe. For further reading we refer to [12]. In §4.3 we provide specifics on the construction of the `SMs` used in this study.

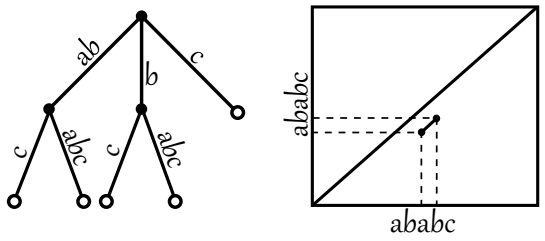

Figure 2: Data structures for the pitch class sequence '*ababc*'. Left: simplified view of a suffix tree. Right: simplified view of a similarity matrix and an off-diagonal stripe indicating a repetition.

[5] The specific formulas used for the computation of melodic attributes can be found in [6][ch.2]

[6] A vector space representing two melodic fragments being compared is computed by: (a) identifying all subsequences (ngrams) contained in the *union* of the two melodic fragments compared, up to a user defined ngram length (ngram lengths used in this study are specified in Table 3); (b) creating two vectors, each representing a melodic fragment, of size equal to the number of distinct ngrams identified. In (b) each dimension of each vector has a value equal to the number of occurrences (frequency) of the ngrams in the corresponding melodic fragment. Example: given two melodic fragments represented as pitch class sequences $f_1 = aba$ and $f_2 = bce$, the *unigram* space (or alphabet) for the union of the fragments is $\mathscr{S} = \{a,b,c,e\}$, and thus the vector of unigrams representing each fragment in that space is $\mathbf{f}_1 = (2,1,0,0)$ and $\mathbf{f}_2 = (0,1,1,1)$.

[7] In melodies temporal gaps correspond to either long note durations, long rests, or a combination of the two.

3. DESCRIPTION OF THE COMPARED MODELS

In this section we motivate our choice of segmentation models for comparison and then briefly describe the models.

Given the low number of models proposed for melodic segmentation at phrase-level granularity,[8] we expanded our criteria to include segmentation models that fulfil at least two of three conditions: (a) tested-on or developed-for melody segmentation, (b) targets segment granularities of phrases or larger (e.g. sections), and (c) used to segment symbolic encodings of music. The resulting list of repetition-based segmentation models can be seen in Table 2. Following Table 2 we provide a brief description of the models.

Table 2: *Model (abbreviation)*: first three letters of the main author's last name; *Attribute*: original attribute sequence used in publication (for abbreviations see Table 1); *Search/Store*: `SM` - similarity matrix, `ST`- suffix trees, `HT` - hash table (asterisk denotes the data structure is fundamental for identification and selection of repetitions) *Similarity*: `E` - exact matching, `A` - approximate matching, in parenthesis there is an abbreviation for the similarity measured employed in the publication (see Table 1); *Salience*: `L` - length, `F` - frequency, `TO` - temporal overlap, `TG` - temporal gap, `P` - position.

Model	Attribute	Identify Repetition		Select Repetition	
		Search/Store	Similarity	Salience	Technique
MUL [8]	chroma	SM*	A (cos)	L, F, TO	path finding
DEH [7]	chords	ST*	E	L, TO	preference rules
WOL [6]	iXr	SM*	A (cos)	L, TO	path finding
CAM [3]	iXr	HT	E	L, F, TO	peak selection
TAK [5]	cp	HT	A (lcs)	TG, P	preference rules

CAM: We implemented the segmentation model described in [3]. The model first identifies repetitions using an exact-match string pattern search algorithm. Second, the model scores the salience of identified repetitions using a heuristic function $h = \frac{\mathtt{L}^l \mathtt{F}^f}{10^{o\,\mathtt{TO}}}$, based on repetition length `L`, frequency `F`, and temporal overlap `TO`, where l, f, o are user defined weights. Third, the model selects meaningful repetitions using a 'boundary strength profile'.[9] The profile is computed by (a) assigning the salience score of a given set of repeated fragments to each instance of the set, and (b) summing the salience scores of all instances of different sets that begin and/or finish at the same time point. Peaks in the profile mark the starting and/or ending points of the most salient repetitions. Fourth, the model selects boundaries using the peak selection algorithm proposed in [13]. Our implementation of CAM has seven parameters: numeric weights l, f, o to control for repetition `L`, `F`, and `TO`, respectively; a numeric weight k to control thresholding in the peak selection algorithm; selection of repetition starts|ends|both to construct the boundary strength profile; minimum and maximum allowed length of repetitions.

TAK: We implemented the segmentation model described in [5]. The model first estimates the locations of temporal gaps in the melody (long note durations or rests) using the method of [14]; second, it uses the locations of the identified temporal gaps as melodic fragment boundaries; third, it uses approximate match and a automatic thresholding method to identify which fragments can be considered repetitions; fourth, using preference rules it selects repetitions which have an instance located (a) after a long temporal gap (`TG`), or (b) at the beginning of the melody (`P`). The starting points of selected repetitions are used as the locations of phrase boundaries. Our implementation of TAK has one parameter: minimum allowed length of repetitions.

DEH: We implemented and adapted the segmentation model described in [7]. The model was originally conceived to find section-level segment boundaries in symbolic sequences of chords. We adapted the model to identify phrase-level segment boundaries in melodies. The model uses an exact-match string pattern search algorithm to identify repetitions, and selects salient repetitions based on `L` and `TO`. First, a suffix tree is constructed. Second, all repetitions that are *maximal* or *supermaximal* (`L`) are selected. A set of repeated fragments is maximal if extending any fragment in the set to the left or right side breaks the equality with the other fragments. Repetitions are supermaximal if they are maximal and not included in any other maximal repeat. The model checks for temporal overlap (`TO`) by requiring contiguous repetitions to have different prefixes. Third, segment boundaries are selected at every melodic position where a salient repetition start and/or ends. Our adaptation of DEH has four parameters: use maximal|supermaximal repetitions; remove overlapping repetitions (no|yes); minimum allowed length of repetitions; select whether repetition starts|ends|both are used as phrase boundaries.[10]

WOL: We implemented the segmentation model described in [5]. The model was originally conceived to give a section-level segmentation of symbolically encoded polyphonic pieces. We adapted the model to identify phrase-level segment boundaries in melodies. The model identifies repetitions as diagonal stripes on the input `SM`, using an optimisation-based search algorithm that considers constraints on `TO` and `L`. To identify diagonal stripes the model first prunes the main diagonal, and then employs dynamic programming [15] to search for the 'best path' through the matrix, i.e. the path that moves through cells of high similarity subject to constraints in motion direction and step size. The search algorithm uses the constraints in motion direction to prioritise moving through diagonals, and the constraints in step size to 'jump' between diagonals. The jumps between diagonals identify the start/end positions of repetitions. Salient repetitions are selected by keeping only the longest (`L`) and less temporally overlapping (`TO`) repetitions. The start/end positions of salient repetitions are taken as segment boundary candidates. Our adaptation of WOL has two parameters: minimum allowed temporal overlap; minimum allowed length of repetitions.

MUL : We tested the segmentation model described in [8], using the implementation provided in [16]. In the past the model has been employed to automatically identify stanza boundaries in folk melody audio recordings with [17] and without [18] reference to score information. In this study we use the model to identify phrase-level segment boundaries in melodies, by using a `SM` constructed from symbolic data (parameters and definition are specified in §4.3). To the best of our knowledge this is the first time MUL is used to segment symbolic melody encodings. MUL identifies salient repetitions as diagonal stripes on the input `SM`, using an optimisation-based search algorithm that considers constraints on `L`, `F`, and `TO`. The 'best path' search is more exhaustive than that of WOL, first identifying path *families* rather than single paths, and then looking for the optimal path family. MUL rejects temporally overlapping repetitions (`TO`) when creating path families, and incorporates `F` and `L` as constraints for the selection of the optimal path family (trying to establish a balance between the two). The implementation provided in [16] has one parameter: minimum allowed length of repetitions.

4. COMPARATIVE EXPERIMENT RESULTS

In this section we first describe the experimental setting for comparison, then present the performance results in Table 4, and finally provide some brief discussion on the results.

[8] To the best of our knowledge only [3, 2, 5] use segmentation for identification of phrase boundaries in melodies. From these [2] was not included in the study since is has not been made publicly available and is not described in sufficient detail to be implemented.

[9] A boundary strength profile is a vector of length equal to the input melody length. In the profile each element value encodes the strength with which a segmentation model 'perceives' a boundary at the temporal location of the element.

[10] The code of DEH is available at `https://bitbucket.org/bash/zmidi-segment`

4.1. Experimental Setting: Melodic Corpora

To test the segmentation models' capacity to identify melodic phrase boundaries, we used a set of 100 instrumental folk songs randomly sampled from the Liederenbank collection[11] (LC) and 100 vocal folk songs randomly sampled from the German subset of the Essen Folk Song Collection[12] (EFSC). We chose to use the EFSC due to its benchmark status in the field of melodic segmentation. Additionally, we chose to use the LC to compare the performance of segmentation models in vocal and non-vocal melodies.[13]

The EFSC consists of ∼6000 songs, mostly of German origin. The EFSC data was compiled and encoded from notated sources. The songs are available in EsAC and **kern formats. The origin of phrase boundary markings in the EFSC has not been explicitly documented (yet it is commonly assumed markings coincide with breath marks or phrase boundaries in the lyrics of the songs).

The instrumental (mainly fiddle) subset of the LC consists of ∼2500 songs. The songs were compiled and encoded from notated sources. The songs are available in MIDI and **kern formats. Segment boundary markings for this subset comprise two levels: 'hard' and 'soft'. Hard (section) boundary markings correspond with structural marks found in the notated sources. Soft (phrase) boundary markings correspond to the musical intuition of two experts annotators.[14] For our experiments we use the soft boundary markings.

4.2. Experimental Setting: Evaluation Measures

We use the same evaluation measures as used in previous comparative studies of melodic segmentation [13], i.e. the well known $F1 = \frac{2 \cdot p \cdot r}{p+r}$, where precision $p = \frac{tp}{tp+fp}$ and recall $r = \frac{tp}{tp+fn}$, with tp, fp, and fn corresponding to the number of true positives (hits), false positives (insertions), and false negatives (misses).

4.3. Experimental Setting: Parameter Exploration

In this section we first provide specifics on the parameters used for the computation of the SMs, and then provide specifics on the model parameters used for experimentation.

In this paper we employ SMs that fulfil the normalisation properties $0 \leq S(i,j) \leq 1$ for $i,j \in [1:N]$, and $S(i,i) = 1$ for $i \in [1:N]$. SM computation depends on the choice of melodic fragment length l, melodic representation, similarity measure (plus ngram length for geometric and statistical measures), and parameters of denoising methods and diagonal enhancing methods [19]. The values of the parameters used for experimentation are listed in Table 3.[15] For a more detailed description of the computation procedures for the SMs used in this study we refer to [6][ch.5].

The segmentation model parameters investigated in this study include the choice of melodic representation, similarity measure (plus ngram length for geometric and statistical measures), choice and setting of selection heuristics (e.g. repetition frequency and length), naïve filtering strategies (e.g. choice of minimum length of repetitions), and selection boundary heuristics (e.g. choosing whether the starts or ends of repetitions should be used as boundaries). The complete list of parameters and of the values used for experimentation is listed in Table 3. The parameter values used were selected after preliminary experimentation with the models. All parameter value combinations in Table 3 were tested. The parameter abbreviations defined in Table 3 are subsequently used in Table 5 to list the most successful parametric settings.

[11] http://www.liederenbank.nl/

[12] http://www.esac-data.org

[13] Vocal music has dominated previous evaluations of melodic segmentation (especially large-scale evaluations), which might give an incomplete picture of the overall performance and generalisation capacity of segmentation models.

[14] Instructions to annotate boundaries were related to performance practice (e.g. "where would you change the movement of the bow").

[15] In our study, we experimented with simple thresholding as a denoising method and gaussian smoothing to enhance diagonal structures. Both proved detrimental to the performance of the models using SMs. These parameters are hence excluded from Table 3.

Table 3: Parameters of compared models.

Parameters		Values for Experimentation
SM melodic fragment length	*fl*	fragment length ∈ {1,2,3,4,5,7,9} notes
vector space model	*ngm*	ngram length ∈ {2} (bigrams)
repetition frequency	*f*	CAM: weight ∈ {0,.5,1,2,3}
repetition length	*l*	CAM: weight ∈ {0,.5,1,2,3}; DEH: choice (maximal mx | supermaximal smx)
overlapping repetitions	*o*	CAM: weight ∈ {0,.5,1,2,3}; DEH: remove (no|yes); WOL: minimum ∈ {3,6,9} notes
minimum repetition length	*mnl*	all models: minimum ∈ {1,3,5,7} notes
maximum repetition length	*mxl*	CAM: maximum ∈ {7,14} notes
select boundary	*sb*	all models: choice (start|end|both)
peak selection algorithm	*ps*	CAM: threshold $k \in \{0,.5,1,2,3\}$

4.4. Results

In Table 4 we present mean recall $\overline{R}$, precision $\overline{P}$, and $\overline{F1}$ results obtained by the segmentation models. We present results at two tolerance levels. Tolerance=*narrow* (upper table) where boundaries are considered as predicted correctly (a tp) if the prediction identifies either the last event of a phrase or the first event of the following phrase. Tolerance=*broad* (lower table) where predicted boundaries are considered correct if the prediction identifies either the previous-to-last, last, first, or second notes of two contiguous phrases.

To allow a better analysis of the results we included four naïve baselines: RND40%, which predicts a segment boundary at random in 40% of the number of events of the melody, RND10%, which does the same as the previous baseline but only on 10% of the melody, ALWAYS, which predicts a segment boundary at every melodic event position, and NEVER which does not make predictions.

We also tested the statistical significance of the paired F1 differences between the compared models and the baselines. For the statistical testing we used a non-parametric Friedman test ($\alpha = 0.05$). Furthermore, to determine which pairs of measurements significantly differ, we conducted a post-hoc Tukey HSD test.

Table 4: Performance of models and baselines for instrumental and vocal melodies. From left to right: mean recall $\overline{R}$, precision $\overline{P}$, and $\overline{F1}$. Highest performances are marked in bold. ∗ indicates $F1$ performances that significantly differ from those of the baselines. + indicates $F1$ performances that are *not* significantly different to the highest performance.

	Database	Vocal			Instrumental		
	Model	$\overline{R}$	$\overline{P}$	$\overline{\mathbf{F1}}$	$\overline{R}$	$\overline{P}$	$\overline{\mathbf{F1}}$
Tolerance: narrow	MUL [8]	0.36	0.38	**0.36**∗	0.35	0.32	+0.32
	DEH [7]	**0.46**	0.33	+0.33∗	0.34	0.34	+0.32∗
	WOL [6]	0.34	0.30	0.29	**0.37**	0.24	+0.27
	TAK [5]	0.31	**0.46**	+0.33	0.36	**0.47**	**0.35**∗
	CAM [3]	0.28	0.24	0.25	0.33	0.19	0.23
	RND40%	0.44	0.16	0.22	0.45	0.13	0.20
	RND10%	0.16	0.19	0.17	0.17	0.16	0.16
	ALWAYS	1.00	0.03	0.05	1.00	0.02	0.03
	NEVER	0.00	0.00	0.00	0.00	0.00	0.00
Tolerance: broad	MUL [8]	0.48	**0.62**	**0.53**∗	0.47	0.51	**0.47**∗
	DEH [7]	**0.52**	0.46	0.44	**0.61**	0.38	0.42
	WOL [6]	0.45	0.43	0.41	0.51	0.36	0.40
	TAK [5]	0.35	0.55	0.38	0.41	**0.55**	+0.43
	CAM [3]	0.44	0.39	0.40	0.44	0.35	0.37
	RND40%	0.60	0.29	0.38	0.60	0.26	0.36
	RND10%	0.26	0.36	0.29	0.25	0.29	0.26
	ALWAYS	1.00	0.03	0.05	1.00	0.02	0.03
	NEVER	0.00	0.00	0.00	0.00	0.00	0.00

Table 5: Parameter configuration of the best performances for the vocal and instrumental set. All models worked best with *sb* : repetition starts. In the table a = attribute, and s = similarity measure, for other abbreviations see Table 3.

		Database	
	Model	Vocal (best params)	Instrumental (best params)
Tolerance: narrow	MUL	*a* :ioi; *s* :lev;*fl* :4	*a* :ioi; *s* :lev;*fl* :4
	DEH	*a* :ior; *o* :yes; *l* :mx; *mnl* :4	*a* :pXi; *o* :yes; *l* :smx; *mnl* :1
	WOL	*a* :ior; *s* :lev;*fl* :3	*a* :cp-iv; *s* :lev;*fl* :4
	TAK	*a* :cpc; *s* :cos	*a* :pXi; *s* :cos
	CAM	*a* :cpc; *mnl* :2; *mxl* :7; *l* :1; *f* :3; *o* :3; *k* :1	*a* :cp; *mnl* :2; *mxl* :7; *l* :1; *f* :3; *o* :3; *k* :1
Tolerance: broad	MUL	*a* :ioi; *s* :lev;*fl* :4	*a* :ioi; *s* :lev;*fl* :4
	DEH	*a* :sli; *o* :yes; *l* :smx; *mnl* :1	*a* :pXi; *o* :yes; *l* :smx; *mnl* :1
	WOL	*a* :ior; *s* :lev;*fl* :4	*a* :cp-iv; *s* :lev;*fl* :3
	TAK	*a* :cpc; *s* :cos	*a* :cpc; *s* :cos
	CAM	*a* :cp-iv; *mnl* :2; *mxl* :7; *l* :1; *f* :3; *o* :3; *k* :1	*a* :cpc; *mnl* :2; *mxl* :7; *l* :1; *f* :3; *o* :3; *k* :1

4.5. Discussion

In this section we analyse the results shown in Table 4 and Table 5. We begin with an analysis of the highest performances and of the statistical significance of the difference between performances. Then we move on to discuss the effect of different parametric settings on the performance of models. We finalise with a discussion on parameter sensitivity, generalisation capability, and efficiency.

Best Performances and Statistical Significant Differences: In the case of vocal melodies MUL obtains the highest $\overline{F1}$ performance (at both narrow and broad tolerance levels). However, the difference between MUL and all other models is only significant when tolerance=broad (when tolerance=narrow the $F1$ performances of MUL are not significantly different to those of TAK and DEH). The differences between the $F1$ performances of all other models is statistically significant, both at tolerance=narrow and at tolerance=broad. A single exception is CAM, whose $F1$ performances significantly differ from those of all other models when tolerance=narrow. In respect to the baselines, only MUL and DEH are significantly different to all baselines at tolerance=narrow, and only MUL is significantly different to all baselines at tolerance=broad. These results point to a numerical and statistical superiority of MUL over the other models, followed closely by DEH. Its important to note that the performance of MUL is driven by precision, while the performance of DEH is driven by recall.

In the case of instrumental melodies, TAK obtains highest $\overline{F1}$ performance when tolerance=narrow, and MUL obtains the highest $\overline{F1}$ performance tolerance=broad. However, the $F1$ performances of TAK are not significantly different to those of MUL, DEH, and WOL when tolerance=narrow, and the $F1$ performances of MUL are not significantly different to those of TAK when tolerance=broad. The differences between the $F1$ performances of all other models are not statistically significant, both at tolerance=narrow and at tolerance=broad. A single exception is CAM, whose $F1$ performances significantly differ from those of all other models when tolerance=narrow. In respect to the baselines, when tolerance=narrow only the $F1$ performances of DEH and TAK are significantly different to those of the baselines (for tolerance=narrow all other models performances are at least equal to those of RND40%). When tolerance=broad, MUL is again the only model whose performance is significantly different than that of the baselines (for tolerance=broad all other models performances are at least equal to those of RND40%). These results point to a numerical and statistical superiority of MUL and TAK over the other models, followed by DEH. Its important to notice that the performance of TAK is driven by precision, the performance of DEH is driven mainly by recall, and the performance of MUL is the result of a balance between precision and recall.

The worst performing model is CAM, which not only obtains low $\overline{F1}$ values, but its $F1$ performances are also often not significantly different to those of the RND40% and RND10% baselines. However, it must be noted that CAM was created only for exploratory purposes, and differently from the other models evaluated in this paper, it was not intended to be a stand alone model of segmentation. In fact, in [3] it is suggested that CAM should operate alongside a 'discontinuity detection based' model of segmentation, and also that it should be employed to find repetitions only within short term temporal windows. Since no details are given as to how these extensions should be carried out, in our implementation we use CAM to find repetitions over the whole melody and without access to discontinuity information.

Melodic Fragment Definition, Representation, and Similarity Assessment: These aspects of repetition identification are encoded in the compared model parameters 'fragment length' fl, 'attribute' a, and 'similarity' s. The best performing configurations of these parameters are listed in Table 5. Our experiments with the fl, a, s parameter configurations aim to investigate (a) how long might perceptually repetitions might be, (b) what attribute representation, if any, might be perceptually preferable for detecting salient repetitions, and (c) what type of similarity measure, if any, might be perceptually preferable for detecting salient repetitions.

In respect to (a), we experimented with various fl sizes for the computation of SMs. For both MUL and WOL the best performances are obtained when relatively brief melodic fragments (3-4 notes long) are used to construct the matrix. This might simply be the result of lower distortions in diagonal stripe structures of the SMs,[16] which might not have a cognitive interpretation. However, the relatively high $\overline{F1}$ and $\overline{P}$ performances obtained by MUL (when compared to those of CAM and DEH), suggest that identifying repetitions locally (i.e. by concatenation of relatively brief similar fragments) might be more appropriate than identifying repetitions of large fragments 'at once' (as done by CAM and DEH).

In respect to (b) our results revealed that none of the melodic attributes investigated is clearly preferable for detecting salient repetitions. The parameter configurations for '*a*' in Table 5 might seem to indicate otherwise, but for none of the models investigated did the choice of attribute selection result in significantly different $F1$ performances. This suggests that the choice of attribute representation in which humans might hear repetitions depends on the specific characteristics of the melody that is been listened to. This finding goes against the intuitions of the researchers that created the compared models, which, as shown in Table 2, use (and often motivate) a specific type of attribute representation.

In respect to (c) our results revealed that approximate match algorithms perform better than exact match algorithms.[17] Yet, our results showed no preference for a specific type of approximate-match similarity measure. That is, for none of the models did the choice of similarity measure selection result in significantly different $F1$ performances. This finding goes against results obtained for assessment of similarity over whole melodies, where generally string measures outperform geometric and statistical measures.

Repetition Selection and Boundary Selection: These aspects of melody segmentation are encoded in the model parameters related to repetition selection heuristics, naïve filtering, boundary selection heuristics. In respect to repetition selection, our results show that the best performing algorithms tend to prioritise length L over frequency F, and allow very little or no temporal overlap TO between repetitions. Also, the high $\overline{P}$ performances obtained by TAK suggest that position P and temporal gaps TG play an important role in repetition selection. In respect to naïve filtering heuristics, filtering short repetitions (1-3 notes) consistently improved the performance of the compared models. Conversely, filtering overly long repetitions resulted in higher efficiency but not in significant performance improvements. In respect to boundary selection heuristics, all models performed best when the starts of repetitions

[16] Short fl (1-2 notes) results in defined but overly short diagonal structures, and long fl (7-9 notes) results in longer but blurry diagonal structures.

[17] The fact that DEH organises identified repetitions by prefixes allows for variations in the suffixes, thus DEH can be said support approximate repetition identification to some extent.

were selected as boundaries.

Parameter Sensitivity: The less sensitive model is MUL, which has the lowest variance in $F1$ performances when the input attribute representation and similarity measure is modified. The models with highest parameter sensitivity are CAM and WOL. The $F1$ performances of CAM are highly dependent on the type of input attribute representation and l, f, o parameters. The $F1$ performances of WOL are mostly independent of the similarity measure, but highly dependent on the attribute representation.

Generalisation and Flexibility: Our experiment tests generalisation in respect to the instrumental tradition to which the folk melodies belong. In general the $\overline{F1}$ performances in the vocal set are slightly higher than those in the instrumental set (for both tolerance levels). However, the difference in performances is not large enough to conclude repetition-based models are not generalisable to different instrumental traditions. The only model that performs better in the instrumental set is TAK. The $\overline{F1}$ performance improvement of TAK can be explained by an increase in recall. This increase in recall suggests that in the instrumental set temporal overlap `TO` and position `P` might have a larger influence in the perception of repetition-cued boundaries than in the vocal set.

We discuss flexibility in two respects: (a) efficiency, and (b) input adaptability, i.e. how easy it is to adapt the model to handle different types of input musical representations, such as monophony/polyphony or symbolic/subsymbolic.

In respect to (a), the main factor affecting model efficiency is the repetition search algorithm employed. The most efficient model is TAK, due to the heuristic used to reduce the set of possible melodic fragments to analyse (see TAK's description in §3). DEH and CAM are also fairly efficient, due to the use of string search based methods (for which linear complexity algorithms exist). The most inefficient models are WOL and MUL, given that by using `SM`s these models have an exponential complexity lower bound.

In respect to (b), models using similarity matrices (MUL and WOL) are are more flexible, as they are able to work with sub-symbolic representations of a melody. Models using similarity matrices can also deal with polyphonic input (in fact, when used to segment audio recordings, the input is generally polyphonic). On the other hand, models like DEH and CAM operate strictly over symbolic input, and thus would require an automatic transcription step if used with audio input. Also, the extension of these models to handle polyphony is not trivial.

5. Conclusions and Future Work

In this study we have compared five different models of repetition-based melody segmentation [3, 5, 6, 7, 8]. We tested the capacity of these models to identify the phrase boundaries of 200 folk melodies (100 vocal and 100 instrumental). The differences between the segmentations of vocal and instrumental melodies are not significant, showing that the models are generalisable to these two sets. Under the scope of this study, the highest performing model is MUL [8], followed by TAK [5] and DEH [7]. The lowest performing model is CAM [3]. The model with lowest parameter sensitivity is MUL [8], and the ones with highest parameter sensitivity are CAM [3] and WOL [6]. The most efficient model is TAK [5] and the less efficient model is MUL [8]. We also provide a large discussion on aspects such as the choice of search method, melodic representation, similarity measure, and parametric configuration.

In future work we will attempt to improve the performance of MUL by incorporating strategies from the other models that seemed to have had a large impact on their performance and that are not yet included in MUL (e.g. temporal gap information and location information for repetition selection as done by TAK). Also, we will conduct a statistical study focused on assessing similarity between *annotated* phrases, to be able to know with more certainty which boundaries within a melody might be cued by repetition perception. This information is instrumental to conduct a more complete evaluation of repetition-based segmentation models, more specifically, to better evaluate the effect of different search strategies used for repetition identification and of the heuristics used to detect perceptually salient repetitions.

Acknowledgments: M. Rodríguez López, A. Volk, and W.B. de Haas are supported by the Netherlands Organization for Scientific Research (NWO-VIDI grant 276-35-001).

References

[1] O. Lartillot: *Reflections towards a generative theory of musical parallelism*. In *Musicae scientiae Discussion Forum*, volume 5:195–229, 2010.

[2] S. Ahlbäck: *Melodic similarity as a determinant of melody structure*. In *Musicae Scientiae*, volume 11(1):235–280, 2007.

[3] E. Cambouropoulos: *Musical parallelism and melodic segmentation*. In *Music Perception*, volume 23(3):249–268, 2006.

[4] F. Lerdahl and R. Jackendoff: *A generative theory of tonal music*. MIT press, 1983.

[5] A. Takasu, T. Yanase, T. Kanazawa, and J. Adachi: *Music structure analysis and its application to theme phrase extraction*. In *Research and Advanced Technology for Digital Libraries*, pages 854–854, 1999.

[6] J. M. Wolkowicz: *Application of Text-Based Methods of Analysis to Symbolic Music*. Ph.D. thesis, Facoulty of Computing Sciences, Dalhousie University, 2013.

[7] W. B. de Haas, A. Volk, and F. Wiering: *structural segmentation of music based on repeated harmonies*. In *IEEE International Symposium on Multimedia (ISM2013)*, pages 255–258. 2013.

[8] M. Müller, P. Grosche, and N. Jiang: *A Segment-Based Fitness Measure for Capturing Repetitive Structures of Music Recordings.* In *ISMIR*, pages 615–620. 2011.

[9] D. Meredith, K. Lemström, and G. A. Wiggins: *Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music*. In *Journal of New Music Research*, volume 31(4):321–345, 2002.

[10] E. H. Margulis: *Musical repetition detection across multiple exposures*. In *Music Perception: An Interdisciplinary Journal*, volume 29(4):377–385, 2012.

[11] D. Gusfield: *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge University Press, 1997.

[12] J. Foote: *Visualizing music and audio using self-similarity*. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, pages 77–80. ACM, 1999.

[13] M. Pearce, D. Müllensiefen, and G. Wiggins: *The role of expectation and probabilistic learning in auditory boundary perception: A model comparison.* In *Perception*, volume 39(10):1365, 2010.

[14] B. Frankland, S. McAdams, and A. Cohen: *Parsing of melody: Quantification and testing of the local grouping rules of Lerdahl and Jackendoff's A Generative Theory of Tonal Music*. In *Music Perception*, volume 21(4):499–543, 2004.

[15] S. Dasgupta, C. H. Papadimitriou, and U. Vazirani: *Algorithms*. McGraw-Hill, Inc., 2006.

[16] M. Müller, N. Jiang, and H. Grohganz: *SM Toolbox: MATLAB Implementations for Computing and Enhancing Similarity Matrices*. In *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[17] M. Müller, P. Grosche, and F. Wiering: *Robust Segmentation and Annotation of Folk Song Recordings*. In *ISMIR*, pages 735–740. 2009.

[18] M. Müller, N. Jiang, and P. Grosche: *A Robust Fitness Measure for Capturing Repetitions in Music Recordings With Applications to Audio Thumbnailing*. In *IEEE Transactions on audio, speech, and language processing*, volume 21(3):531–543, 2013.

[19] M. Muller and F. Kurth: *Enhancing similarity matrices for music audio analysis*. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 5. 2006.