Analyzing the influence of pitch quantization and note segmentation on singing voice alignment in the context of audio-based Query-by-Humming

Jose J. Valero-Mas
Pattern Recognition and Artificial Intelligence Group, University of Alicante
jjvalero@dlsi.ua.es

Justin Salamon
Music and Audio Research Laboratory, New York University
justin.salamon@nyu.edu

Emilia Gómez
Music Technology Group, Universitat Pompeu Fabra
emilia.gomez@upf.edu

ABSTRACT

Query-by-Humming (QBH) systems base their operation on aligning the melody sung/hummed by a user with a set of candidate melodies retrieved from polyphonic songs. While MIDI-based QBH builds on the premise of existing annotated transcriptions for any candidate song, audio-based research makes use of melody estimation algorithms for the songs. In both cases, a melody abstraction process is required for solving issues commonly found in queries such as key transpositions or tempo deviations. Full automatic music processes are commonly used for this, but due to the reported limitations in state-of-the-art methods for real-world queries, other possibilities should be considered. In this work we explore three different melody representations, ranging from a general time-series one to more musical abstractions, which avoid full automatic transcription, in the context of an audio-based QBH system. Results show that this abstraction process plays a key role in the overall accuracy of the system, obtaining the best scores when temporal segmentation is dynamically performed in terms of pitch change events in the melodic contour.

1. INTRODUCTION

Query-by-Humming systems constitute a particular case of content-based music similarity search schemes in which the input query is a sung, hummed or whistled section of a song, usually its main melody [1, 2], and the output is the target song. Such a music retrieval paradigm stands as an interesting alternative to classic text-based retrieval frameworks (for instance, tag-based search) for its simple usage complemented by the fact that no musical knowledge from the user is required [3].

Research in QBH mainly focuses on addressing the inaccuracies found when producing the queries: on the one hand, *tuning issues* have to be considered as users may sing out of tune and/or in a different key [4]; on the other hand, *tempo deviations* among queries and candidates may also occur [4, 5]. For overcoming them, a *melody abstraction* process, which may range from general time-series codifications to more sophisticated music-based ones, followed by a *melody comparison* stage are performed for estimating the dissimilarity between the query and the candidates [6].

The process for obtaining the set of candidate melodies is not trivial [2, 5, 7]: main fundamental frequency (f0) estimation for queries and candidates cannot be assumed as an accurate process, especially when dealing with polyphonic songs [8]. While this estimation process is inevitable for the queries as they constitute the user audio input to the system, this issue has been typically avoided for the candidate songs by assuming the existence and availability of high-level annotated representations (for instance, MIDI files) of these melodies.

Due to the limitations the previous assumption implies, mostly in terms of practical systems, some QBH schemes try to estimate this melody algorithmically from audio. Although more realistic, this adds more complexity to the system since no melody estimation algorithm is error-free.

As aforementioned, melodic contours require of an abstraction process. For taking advantage of the large amount of research carried in the symbolic melodic similarity field, melodies estimated from audio sources are coded into high-level music representations [9], usually with full automatic music transcription systems. However, given the limitations current state-of-the-art transcription algorithms exhibit [10], it seems interesting to study alternative abstractions to such high-level representations.

In this paper we present a study of the influence of different *melody abstraction* processes which avoid the complexity of full automatic music transcription in the context of QBH. Particularly, we assess the influence of pitch quantization and note segmentation in singing voice alignment for QBH. For that, we take as starting point the scheme in Figure 1 and we evaluate three different melodic contour representations: the first one makes use of the time-series encoding algorithm Symbolic Aggregate Approximation (SAX) [11], which is based on a fixed-duration temporal segmentation and statistical encoding; the second one modifies the original SAX algorithm so that the encoding is performed using a semitone-band representation; finally, as a third method we propose to segment the melody using the pitch change events in the melodic contour.

To ensure the scalability of the system we use the melody estimation algorithm MELODIA [12]. This method estimates the predominant pitch from both monophonic and

polyphonic music signals. In terms of the contour comparison, we apply two sequence alignment algorithms: Smith-Waterman [13], originally meant for DNA sequences but with large application in the time series field, and Subsequence Dynamic Time Warping [14].

The rest of the paper is structured as it follows: Section 2 briefly reviews similar research proposals; Section 3 and Section 4 present the melody extraction algorithm MELODIA and the local alignment algorithms considered respectively; Section 5 introduces the assessed contour representations; Section 6 presents the evaluation methodology; Section 7 presents and discusses the results obtained; finally, Section 8 outlines the conclusions obtained and proposes possible future work.

2. RELATED WORK

One of the first proposed QBH systems was the one by Ghias et al. [15] in which queries were transcribed using autocorrelation for pitch tracking, the candidate elements were MIDI files and the search was performed using a fuzzy string matching algorithm. Although many similar systems based on some kind of full automatic music transcription have been proposed since then, the work by Dannenberg et al. [3] with the MUSART Testbed, a framework for the assessment of this type of QBH systems, stands as a relevant example.

In terms of systems not based on full automatic music transcription, a relevant example is the one by Duda et al. [1] in which a series of audio descriptors (Mel-Frequency Cepstrum Coefficients, Power, Fundamental frequency contour, Voice Formants and Chroma) are extracted from the audio files and are then encoded using SAX [11]; similarity is performed using Edit distance [17].

Another example can be found in the system by Ito et al. [5]. In this case, instead of obtaining a single melodic contour for the candidate elements, multiple fundamental frequency candidates are retrieved, using a variation of the PreFEst algorithm [18], for comparison to the query contour using a basic scoring function. Salamon et al. [2] proposed a system in which melodies are quantized into semitones and mapped into one octave. Similarity is performed using the Q_{max} algorithm [19].

In terms of the automatic extraction of melodies, some explored techniques use fundamental frequency extraction algorithms [5, 16], main singing voice extraction [1, 7] or the use of predominant melody estimation algorithms [2].

All approaches are summarized in Table 1.

3. MELODY ESTIMATION

Melodies from both queries and candidate songs are obtained using the predominant melody estimation algorithm MELODIA [12] [1] . For a given music piece, the algorithm estimates the fundamental frequency of the predominant melodic line in the song. This particular algorithm outperformed all other state-of-the-art methods in the 2011 Music Information Retrieval Evaluation eXchange (MIREX) campaign [2] in the *Audio Melody Extraction* task.

In a more detailed analysis, results in [12] report its robustness in terms of octave errors (properly tracking pitch values in the correct octave) and voiced frame detection (frames belonging to the predominant melody). However, it must be also pointed out that the algorithm tends to confuse unvoiced elements as voiced, thus lowering the overall performance.

Finally, we provide a brief explanation to the four stages MELODIA comprises: an initial *Sinusoid extraction* step estimates the predominant frequency values at each instant in the signal; then, a *Salience function* based on a harmonic series is derived; after that, a series of *Pitch contours* are created using a set of rules based on Auditory Scene Analysis (ASA) for finally selecting the predominant melody in the *Melody selection* stage. In this experimentation, MELODIA has been configured to its default analysis rate ($\Delta t_{\text{MEL}} = 2.9$ ms).

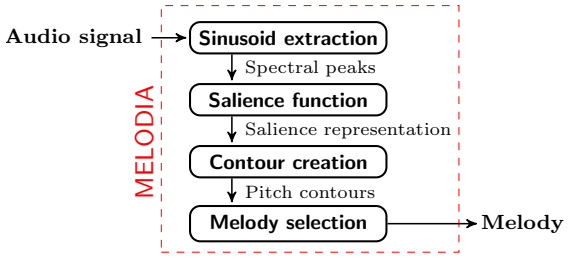

Figure 2. Block diagram of the MELODIA algorithm.

4. MELODY ALIGNMENT

In this work, similarity between the query and the candidate melodies is estimated by means of sequence alignment methods. This premise suits the QBH task as queries may contain tempo deviations with respect to the corresponding melodies of the actual song to be retrieved. The two algorithms considered are now introduced.

4.1 Smith-Waterman

The Smith-Waterman (SW) method [13] is an alignment algorithm formerly proposed for DNA sequences. This algorithm performs a search for the most similar regions between a pair of sequences, coded as strings, in a time-warped scenario. Smith-Waterman requires a series of costs to be defined: a reward for symbol matches (C_{MATCH}), a penalty for mismatches (C_{MISMATCH}) and two costs for time warps ($C_{\text{INSERTION}}$ and C_{DELETION}). Table 2 shows the different configurations considered.

4.2 Subsequence Dynamic Time Warping

Subsequence Dynamic Time Warping (S-DTW) constitutes a modification on Dynamic Time Warping (DTW) proposed by Müller in [14]. While DTW forces a global alignment between two sequences, S-DTW eliminates that restriction for allowing local matches between the sequences. The modification makes it suitable for query-by-example

[1] http://mtg.upf.edu/technologies/melodia

[2] http://www.music-ir.org/mirex/wiki/MIREX_HOME

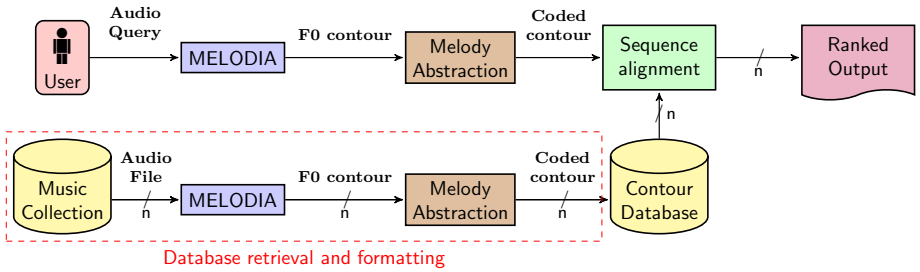

Figure 1. Scheme of the QBH system proposed. Main melodies are estimated from the audio files (query and candidate songs) using the melody estimation algorithm MELODIA, being then encoded using a certain contour representation; local alignment between the query and each element in the database is then performed and the results are eventually ranked.

First Author	Feature(s)	Feature extraction		Abstraction	Similarity
		Query	Music collection		
Ghias [15]	Main F0 contour	Pitch tracking (autocorrelation)	MIDI files	Strings representing changes in contour: U (up), D (down) and S (same)	Fuzzy string matching
Dannenberg [3]	Main F0 contour	Pitch tracking (autocorrelation)	MIDI files	IOI + Relative pitch, Fixed-Time Segmentation + Relative pitch	Note Interval, N-gram, Contour Matching, HMM Matching, CubyHum Matcher
Duda [1]	MFCC, Audio Power, F0, Voice Formants, Chroma + derivatives (1st and 2nd order)	No extraction	Stereo pan removal to retrieve lead singing voice	SAX coefficients	Edit distance
Jeon [16]	Main F0 contour	Constant-Q Transform + heuristics	Constant-Q Transform + heuristics	Wavelet coefficients	Coefficient's comparison
Ito [5]	Multiple F0 contours	PreFEst variation	PreFEst variation	Tempo normalization + logarithm of frequencies values	Scoring function (absorbs key differences)
Salamon [2]	Main F0 contour	MELODIA	MELODIA	Semitone-band based chromagrams with fixed-time segmentation	Q_{max}
Rocamora [7]	Lead singing voice	YIN + energy-based segmentation and extraction	Singing voice detection and (+ query process)	Pitch and duration ratios (relative encoding)	Edit distance

Table 1. Summary of related QBH approaches.

	C_{MATCH}	$C_{MISMATCH}$	$C_{INSERTION}$	$C_{DELETION}$
T1	1	-0.5	-0.5	-0.5
T2	1	-1	-0.5	-0.5
T3	1	-1	-1	-1
T4	1	-0.5	-1	-1

Table 2. Weights of the four tested configurations for the Smith-Waterman alignment algorithm.

applications [20] as queries usually constitute an excerpt of the element to be retrieved. The cost function used in this paper has been the Edit distance (ED) [17].

5. MELODY ABSTRACTIONS

We now describe the three considered melody abstractions for encoding the estimated melodic pitch contours.

5.1 Symbolic Aggregate Approximation (SAX)

SAX, introduced by Lin et al. [11] in 2007, is a symbolic representation for time series (sequences encoded as strings) able to cope with two major drawbacks usually found in other methods: the need for both a *dimensionality reduction* and a *lower bound* in the distance computations. Although reported as a fast and competitive algorithm for similarity search, SAX has not been widely used in Music Information Retrieval (MIR). Some of the few examples in

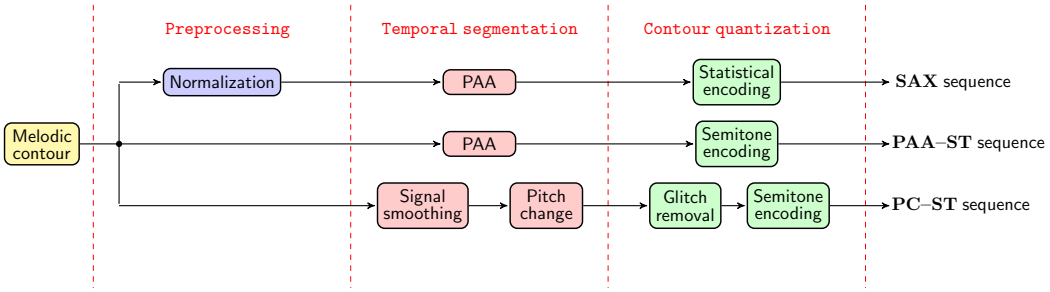

Figure 3. Diagram depicting the different stages the three proposed abstractions comprise.

this field can be found in the study of guitar articulations [21], Beijing opera singing similarity [22] or in QBH [1].

SAX comprises three steps for coding any sequence:

5.1.1 Time-series normalization

Given a time series $C = \{c_1, c_2, ..., c_n\}$ of length **n**, this abstraction performs an initial normalization process:

$$c'_i = \frac{c_i - \mu}{\sigma} \qquad 1 \leq i \leq n \tag{1}$$

were c_i represents each element of the initial time series (the f0 contour in cents [3] retrieved by MELODIA) and μ and σ the mean and the standard deviation respectively.

5.1.2 Piecewise Aggregate Approximation (PAA)

This second stage takes the normalized time series C' of length **n** and maps it in an **M**-dimensional (modifiable parameter) vector $\bar{C} = \{\bar{c}_1, \bar{c}_2, ..., \bar{c}_M\}$ of equally-sized segments:

$$\bar{c}_i = \frac{M}{n} \cdot \sum_{j=\lfloor \frac{n}{M} \rfloor (i-1)+1}^{\lfloor \frac{n}{M} \rfloor i} c'_j \qquad 1 \leq i \leq M \tag{2}$$

Given the different length of the f0 sequences to encode, fixing a global **M** value would produce each segment to represent a different temporal duration in each sequence. Instead, we fix a frame temporal duration τ_t for all sequences. Since each c'_i represents Δt_{MEL}, the frame size in samples can be obtained as $\tau_s = \tau_t / \Delta t_{\text{MEL}}$. Thus, **M** is given by $M = n/\tau_s$. As an initial experiment, τ_t values considered are 0.3, 0.5, 0.8, 1 and 2 seconds.

5.1.3 Symbolic representation

The last stage maps $\bar{C}$ to a series of **a** (adjustable parameter) discrete symbols. To assure equiprobability of appearance for all symbols, **a** regions are defined based on a statistical distribution, typically Gaussian [11]. The group of breakpoints $B = (\beta_1, \beta_2, ..., \beta_{a-1})$ for delimiting such regions accomplish that the area under a $\mathcal{N}(0,1)$ Gaussian curve from β_j to β_{j+1} equals $1/\mathbf{a}$. In addition, $\beta_0 = -\infty$ and $\beta_a = +\infty$.

[3] The reference frequency is 55 Hz as it represents the minimum frequency value retrieved by MELODIA.

Each interval $[\beta_{j-1}, \beta_j)$ represents a certain symbol α_j. Therefore, **M**-length vector $\bar{C} = \{\bar{c}_1, \bar{c}_2, ..., \bar{c}_M\}$ is mapped into the **M**-length vector $\hat{C} = (\hat{c}_1, \hat{c}_2, ..., \hat{c}_M)$:

$$\hat{c}_i = \alpha_j \quad \text{if} \quad \bar{c}_i \in [\beta_{j-1}, \beta_j) \qquad \begin{matrix} 1 \leq i \leq M \\ 1 \leq j \leq a \end{matrix} \tag{3}$$

As an exploratory study, the **a** tested values have been 3, 4, 6, 8, 12, 16 and 20.

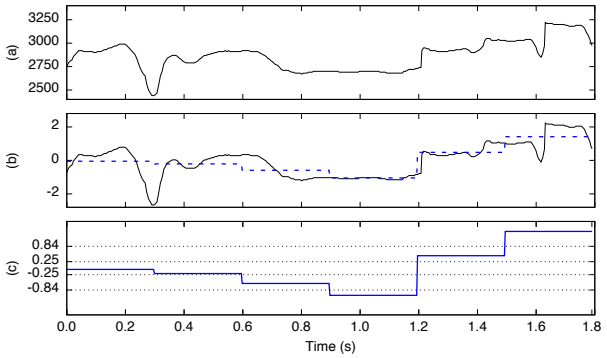

Figure 4. Example of the SAX abstraction process with $\mathbf{a} = 5$ and $\tau_s = 0.3$ *s*: (a) Initial time series in cents; (b) Normalized time series (solid) and PAA codification (dashed); (c) PAA codification (solid) and SAX encoding breakpoints (dotted).

5.2 PAA temporal segmentation with semitone quantization (PAA–ST)

The first proposed SAX modification revises the *Symbolic representation* stage: instead of using a statistical distribution approach for the vertical quantization, a fixed grid with semitone divisions is established. The minimum considered frequency value is 55 Hz given it is the minimum f0 retrieved by MELODIA. The normalization stage is omitted as it modifies the pitch range. Folding the contour to a single octave as in [2] was discarded as preliminary non-exhaustive experimentation did not report improvements.

Finally, relative pitch encoding is applied (storing intervals between segments) to provide transposition invariance. In this abstraction, the assessed time durations for the PAA segments have been the same as in the SAX abstraction.

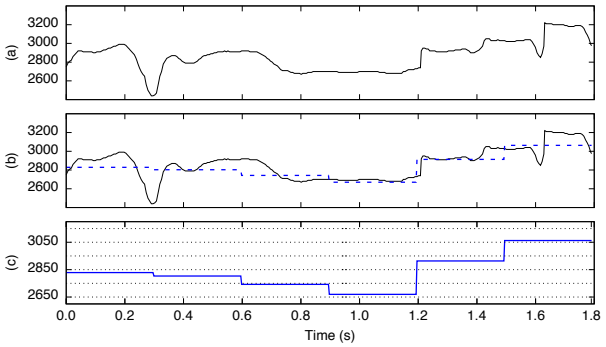

Figure 5. Example of the PAA–ST abstraction process with $\tau_s = 0.3\ s$: (a) Initial time series in cents (solid); (b) Initial time series in cents (solid) and PAA codification (dashed); (c) PAA codification (solid) and semitone grid breakpoints (dotted).

5.3 Pitch change segmentation with semitone quantization (PC–ST)

This second modification builds on the previous one but avoids PAA and dynamically segments the melodic contour when there is a pitch change event. Vertical quantization using a semitone grid is maintained. In order to avoid *false* segments due to artifacts and fast pitch changes the pitch contour may contain, a softening process is applied.

The softening process comprises two steps: (a) an initial *signal smoothing* using an average filter of τ_{SM} duration with sliding window (applied before the semitone quantization process) and (b) a *glitch removal* step by applying a median filter of τ_{GR} with sliding window for removing segments shorter than a certain duration (applied after the semitone quantization step).

We have studied four different filter durations: 25, 50, 75 and 100 pitch samples. Given the MELODIA analysis rate, these values correspond to filter durations τ_{SM} and τ_{GR} of 70, 140, 218 and 290 milliseconds respectively.

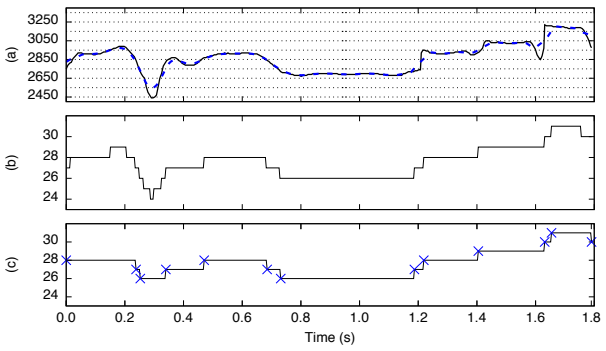

Figure 6. Example of the PC–ST abstraction process with $\tau_{SM} = 70\ ms$ and $\tau_{GR} = 140\ ms$: (a) Initial time series in cents (solid), smoothed contour after the first filter (dashed) and semitone grid (dotted); (b) absolute semitone encoding; (c) absolute semitone encoding after the second filter, the cross symbol ($\times$) points out each new temporal segment.

6. EVALUATION METHODOLOGY

6.1 Dataset

The evaluation data is the same as in [2] and it comprises a query corpus and a music collection.

The music collection, or candidate songs, contains 2125 commercial songs [19] distributed in 523 groups (each one being a group of covers of the same song). Song lengths range from 0.5 to 8 minutes with an average duration of 3.6 minutes. Following the evaluation strategy in [2], the collection is divided into two subsets: a first one containing only canonical songs [4] from the corpus (481 elements) and a second one comprising the entire music collection (2125 elements).

The freely-available query corpus set [5] comprises a total of 118 queries recorded by 17 users (9 female and 8 male) whose musical knowledge ranged from none to amateur musician, with an average of 6.8 queries per user (1 as a minimum and 11 as a maximum). As reference songs, users chose among the 481 canonical subset of the music collection. Queries range from 11 to 98 seconds, with an average length of 28.6 seconds.

6.2 Measures

Generally, a QBH system is assessed using rank metrics as its output is a sorted list of the similarity scores between the query and each candidate melody. In these terms, the two most common evaluation measures are the Mean Reciprocal Rank (MRR) and the Top-X Hit Rate.

6.2.1 Mean Reciprocal Rank (MRR)

For a given user query **Q**, corresponding to a target song **A**, the system returns sorted list in which song **A** is located at position (or rank) **r**. The Reciprocal Rank (RR) for **A** is defined as 1/**r**. Generalizing for a series of **n** queries, the Mean Reciprocal Rank (MRR) is defined as:

$$\mathrm{MRR} = \frac{1}{\mathrm{N}} \cdot \sum_{i=1}^{\mathrm{N}} \frac{1}{\mathrm{r}\left(\mathrm{Q}_i\right)} \tag{4}$$

Scores obtained fall in the range $0 \leq \mathrm{MRR} \leq 1$, where 0 stands for the worst case and 1 for the best.

For any of the evaluation sets considered, **r** is assumed to be highest-ranked version matching query **Q**.

6.2.2 Top-X Hit Rate

Given the resulting rank, this measure checks whether the position **r** of the matching element of **Q** is among the first **X** positions of the list, *i.e.* $\mathbf{r}\left(\mathbf{Q}_i\right) \leq \mathbf{X}$. This estimates the frequency of retrieving the correct result among the first **X** positions [2].

As in the previous case, the highest-ranked version which matches query **Q** is considered as **r**.

[4] The songs as published by the band who composed/played it.
[5] http://mtg.upf.edu/download/datasets/MTG-QBH.

7. RESULTS AND DISCUSSION

7.1 Results

Results obtained for the abstractions and alignment algorithms considered are presented in Table 3. Due to space requirements, only best result obtained for each configuration is reported. In order to consistently assess these results, a baseline configuration has been added: for each query, the candidates' rank is randomly sorted (without performing any similarity measure) and the evaluation figures are then obtained; the results shown for this configuration constitute the average of 10,000 repetitions. Results from [2] are also included for a comparative assessment.

We note that all the proposed configurations significantly outperform the MRR figure of 0.014 obtained with the considered baseline. However, the results are still considerably lower than the ones obtained in [2]. Nevertheless, the differences in performance among the different configurations allow us to make some interesting observations.

We see that the combination of SAX with the Smith-Waterman alignment obtains an MRR of 0.05 when evaluated against the canonical (481 songs) test set. The semitone quantization step, which constitutes the only difference with the SAX abstraction process, does not significantly affect the results with respect to the SAX ones (MRR score is now around 0.04). This is a point to be remarked since, although the abstraction is more related to an actual music representation, the accuracy scores obtained are similar to the ones obtained with SAX.

PC–ST assesses the influence of note segmentation in the process. Focusing on the canonical set and the Smith-Waterman alignment, this particular encoding methodology achieves an MRR score around 0.09, thus outperforming the two other abstractions. This suggests that musically-informed temporal segmentation of pitch sequences may benefit the performance of the system.

As expected from [2], the inclusion of cover songs among the candidates set enhances retrieval accuracy for our configurations, except for the PAA–ST: while for both SAX and the PC–ST there is an improvement of 0.05 in the MRR measure, results in the PAA–ST do not significantly vary in comparison with the canonical set.

Results obtained for the Top-X Hit Rate measure also support our observation that a proper temporal segmentation in the process is beneficial for the system. When only considering the canonical set, the correct candidate is retrieved on the first position around 3 % and 1 % of the time for the SAX and the PAA–ST respectively while, when considering the PC–ST, this figure goes close to 6 %. This same conclusion can be observed with the rest of the Hit Rates (3, 5 and 10) as well as with the inclusion of covers among the candidates.

Focusing on the alignment algorithms, although the different proposed Smith-Waterman configurations show some influence on the overall accuracy, there is no clear outperforming configuration for all the cases. Results obtained with Subsequence Dynamic Time Warping show lower performance than the other considered alignment algorithm. This may be improved with the use of more complex cost functions rather than the considered Edit distance.

7.2 Discussion

While the proposed SAX abstraction has been shown to perform successfully for a variety of time-series tasks [11], results in the experiments proposed suggest that this is not the case for musical time-series data in the context of QBH. The most likely reason for this to happen is the fact that SAX does not consider any particularities the origin domain of the time series may have. Thus, in the case of QBH, SAX may be abstracting away musically-related information from the melodic contours required for properly performing the alignment. This idea is further supported by the improvement in the results when using the PC–ST abstraction as, although in a very naïve way, it tries to segment the different musical notes present in the contour.

The results obtained in the two modifications proposed support the relevance of using musically-informed temporal segmentation of the contour. In this study, the use of a basic temporal segmentation based on pitch change events leads to accuracy improvements when compared to the use of the PAA dimensionality reduction algorithm. The most likely reason for this is again the fact that the use of the PAA algorithm does not take into account the musical nature of the data to encode, thus abstracting away relevant information necessary for the alignment. In these terms, the use of more sophisticated temporal segmentation techniques for music data, as for instance onset detection, could improve these results.

Although the abstractions studied in this paper are not competitive in terms of a practical QBH system, evidence from previous work (cf. [2]) shows non-transcription abstractions may lead to successful results. These results encourage the exploration of other abstractions to provide competent alternatives to transcription-based QBH systems.

8. CONCLUSIONS

Query-by-Humming (QBH) systems constitute a particular type of music search engine in which the query is a sung or hummed excerpt of the main melody of a song. Most often, these schemes rely on both existing music annotations and fully-automated music transcription algorithms for performing the melodic similarity. Although many examples of QBH systems have been proposed under this premise, its limited scalability together with the fact that no full automatic transcription algorithm is error-free clearly limits their performance in practical situations.

In this work we assessed the influence of this particular step in such systems by using of three melody encoding alternatives which avoid full music transcription. More precisely, starting from the general time-series encoding method Symbolic Aggregate Approximation (SAX), we modify this algorithm by incorporating music-based pitch quantization and segmentation for evaluating their influence in the context of a QBH system. Results obtained suggest that the time-series representation algorithm SAX does not seem to be suitable for melody alignment in the context of Query by Humming. In this sense, the main out-

Approach	Evaluation subset	Alignment algorithm	Algorithm configuration	MRR	Top-X Hit Rate (%)			
					1	3	5	10
SAX	Canonical	SW	T1	0.0500	2.54	5.93	7.63	9.32
			T2	0.0566	2.54	5.93	5.93	11.02
			T3	0.0632	4.24	5.93	5.93	9.32
			T4	0.0472	3.39	4.24	5.08	6.78
		S-DTW	ED	0.0333	1.69	3.39	3.39	8.47
	Complete	SW	T1	0.1117	7.63	11.86	12.71	17.80
			T2	0.1155	7.63	11.86	12.71	17.80
			T3	0.0962	5.08	10.17	11.86	14.41
			T4	0.0849	5.08	8.47	11.02	12.71
		S-DTW	ED	0.0443	2.54	4.24	5.08	8.47
PAA–ST	Canonical	SW	T1	0.0515	2.54	4.24	6.78	11.02
			T2	0.0421	1.69	3.39	4.24	9.32
			T3	0.0391	1.69	2.54	4.24	6.78
			T4	0.0424	1.69	4.24	4.24	5.93
		S-DTW	ED	0.0346	1.69	2.54	3.39	5.93
	Complete	SW	T1	0.0396	1.69	2.54	5.93	9.32
			T2	0.0424	1.69	3.39	4.24	8.47
			T3	0.0406	1.69	3.39	5.08	8.47
			T4	0.0558	3.39	5.08	5.93	9.32
		S-DTW	ED	0.0334	1.69	2.54	6.78	9.32
PC–ST	Canonical	SW	T1	0.0894	5.93	9.32	10.17	12.71
			T2	0.0967	6.78	11.86	12.71	15.25
			T3	0.0957	6.78	8.47	12.71	14.41
			T4	0.0772	5.08	6.78	8.47	12.71
		S-DTW	ED	0.0165	0.00	0.85	1.69	4.24
	Complete	SW	T1	0.1447	10.17	14.41	17.80	24.58
			T2	0.1460	10.17	16.95	19.49	22.88
			T3	0.1563	11.02	16.95	17.80	22.88
			T4	0.1447	10.17	14.41	17.80	24.58
		S-DTW	ED	0.0181	0.00	0.85	0.85	3.39
Baseline	Canonical	Random		0.0140	0.21	0.62	1.03	2.06
	Complete	Random		0.0039	0.05	0.15	0.25	0.50
Salamon [2]	Canonical	Q_{max}		0.45	40.68	47.46	49.15	51.69
	Complete	Q_{max}		0.56	50.85	58.47	61.02	66.10

Table 3. MRR and Top-X Hit Rate results obtained for the proposed experimentation. Figures represent the best score achieved in each particular abstraction configuration.

come of this study is that, given the complexity of Query by Humming, musically-related abstractions should be considered for encoding the contours.

Future work will consider the incorporation of the conclusions obtained in this work to the abstraction proposed in [2]: as the abstraction in the cited work performs a chromagram representation with a fixed-time temporal segmentation, the incorporation of dynamically-based segmentation could improve the results obtained. Moreover, given the relevance of the user in this particular task, interactive pattern recognition paradigms for addressing the similarity step could be considered: when a query is incorrectly answered, the system could modify the dissimilarity measure (metric learning) to incorporate the user's feedback.

Acknowledgments

This research work has been partially supported by Consejería de Educación de la Comunitat Valenciana through project PROMETEO/2012/017, Vicerrectorado de Investigación, Desarrollo e Innovación de la Universidad de Alicante through FPU programme (UAFPU2014–5883), the Spanish Ministerio de Economía y Competitividad through project TIMuL (No. TIN2013–48152–C2–1–R, supported by EU FEDER funds) and the Spanish entity Fundació

Obra Social 'laCaixa'. Authors would also like to thank José M. Iñesta for kindly proofreading this paper.

9. REFERENCES

[1] A. Duda, A. Nürnberger, and S. Stober, "Towards Query by Singing/Humming on Audio Databases," in *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Austria, 2007, pp. 331–334.

[2] J. Salamon, J. Serrà, and E. Gómez, "Tonal Representations for Music Retrieval: From Version Identification to Query-by-Humming," *International Journal of Multimedia Information Retrieval, special issue on Hybrid Music Information Retrieval*, vol. 2, no. 1, pp. 45–58, 2013.

[3] R. B. Dannenberg, W. P. Birmingham, B. Pardo, N. Hu, C. Meek, and G. Tzanetakis, "A Comparative Evaluation of Search Techniques for Query-by-humming Using the MUSART Testbed," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 5, pp. 687–701, 2007.

[4] D. Little, D. Raffensperger, and B. Pardo, "A Query by Humming System that Learns from Experience," in *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Austria, 2007, pp. 335–338.

[5] A. Ito, Y. Kosugi, S. Makino, and M. Ito, "A query-by-humming music information retrieval from audio signals based on multiple F0 candidates," in *Proceedings of the International Conference on Audio Language and Image Processing (ICALIP)*, China, 2010, pp. 1–5.

[6] M. Ryynänen and A. Klapuri, "Query by humming of midi and audio using locality sensitive hashing," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, USA, 2008, pp. 2249–2252.

[7] M. Rocamora, P. Cancela, and A. Pardo, "Query by humming: Automatically building the database from music recordings," *Pattern Recognition Letters*, vol. 36, no. 1, pp. 272–280, 2014.

[8] J. Salamon, E. Gómez, D. P. Ellis, and G. Richard, "Melody Extraction from Polyphonic Music Signals: Approaches, applications, and challenges," *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.

[9] R. Typke, "Music retrieval based on melodic similarity," Ph.D. dissertation, Utrecht University, Netherlands, February 2007.

[10] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions." *J. Intell. Inf. Syst.*, vol. 41, no. 3, pp. 407–434, 2013.

[11] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing SAX: A Novel Symbolic Representation of Time Series," *Data Mining and Knowledge Discovery*, vol. 15, no. 2, pp. 107–144, 2007.

[12] J. Salamon and E. Gómez, "Melody Extraction from Polyphonic Music Signals using Pitch Contour Characteristics," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.

[13] T. Smith and M. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981.

[14] M. Müller, *Information retrieval for music and motion*. Springer, 2007.

[15] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith, "Query by Humming: Musical Information Retrieval in an Audio Database," in *Proceedings of the 3rd ACM International Conference on Multimedia*, USA, 1995, pp. 213–236.

[16] W. Jeon, C. Ma, and Y. M. Chen, "An Efficient Signal-Matching Approach to Melody Indexing and Search Using Continuous Pitch Contours and Wavelets," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, Japan, 2009, pp. 681–686.

[17] R. A. Wagner and M. J. Fischer, "The String-to-String Correction Problem," *Journal of the Association for Computing Machinery*, vol. 21, no. 1, pp. 168–173, 1974.

[18] M. Goto, "A real-time music-scene-description system: predominant-F0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Communication*, vol. 43, no. 4, pp. 311–329, 2004.

[19] J. Serrà, H. Kantz, X. Serra, and R. G. Andrzejak, "Predictability of Music Descriptor Time Series and its Application to Cover Song Detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 514–525, 2012.

[20] X. Anguera and M. Ferrarons, "Memory efficient subsequence DTW for Query-by-Example Spoken Term Detection," in *IEEE International Conference on Multimedia and Expo (ICME)*, USA, 2013, pp. 1–6.

[21] T. H. Özaslan and J. L. Arcos, "Legato and Glissando identification in Classical Guitar," in *Proceedings of the 7th Sound and Music Computing Conference (SMC)*, Spain, 2010, pp. 457–463.

[22] S. Zhang, R. C. Repetto, and X. Serra, "Study of the Similarity Between Linguistic Tones and Melodic Pitch Contours in Beijing Opera Singing," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, Taiwan, 2014, pp. 343–348.