Tempo estimation from symbolic annotations with periodic functions

Nicolas Lazzari[1,2,*], Valentina Presutti[1]

[1]Dipartimento di Lingue, Letterature e Culture Moderne, University of Bologna, Via Cartoleria, 5, Bologna 40124, Italy
[2]Dipartimento di Informatica, University of Pisa, Largo B. Pontecorvo, 3, Pisa 56127, Italy

Abstract

The tempo estimation task has been traditionally performed on musical compositions mostly represented as audio or MIDI. Recent methods obtain near-perfect results. Nevertheless, the same methods applied to symbolic representations, such as textual chord annotations, result in inaccurate estimations. This hampers the harmonisation of heterogeneous datasets composed of symbolic annotations since a conversion step towards a common representation is needed. In this paper, we propose a novel method to obtain accurate tempo estimation on musical compositions encoded using textual symbolic annotations, relying on relevant cognitive and musicological theories. All the code is available at https://github.com/n28div/TEwPF.

Keywords

Music Information Retrieval, Music tempo estimation, Symbolic music

1. Introduction

Estimating the tempo of music compositions is a well-researched area driven by real-world applications, from recommender systems to similarity measures [1]. For instance, Gouyon and Dixon [2] perform genre classification using only tempo information. The results are comparable to the performances of the same algorithm when using audio representations. Indeed, the tempo of a composition has a great influence on the cognitive perception of listeners and composers [3] as well as computational applications [1]. Faster compositions tend to be perceived as happier while slower compositions as sadder. Moreover, it has been observed that neural activity modulates in the presence of music with a faster tempo [4], enhancing performances in reactive tasks.

The tempo estimation task is defined as the identification of the frequency that humans tap to a musical composition [5]. It is characterized by two sub-tasks: global and local tempo estimation. The global estimation assumes that a constant tempo can be observed throughout the whole musical composition [1] while local estimation relaxes such constraint and takes into account time fluctuations [6]. Global tempo estimation is a

CREAI 2024 - Workshop on Artificial Intelligence and Creativity, editors Allegra De Filippo, Francois Pachet, Valentina Presutti, Luc Steels
*Corresponding author.
nicolas.lazzari3@unibo.it (N. Lazzari); valentina.presutti@unibo.it (V. Presutti)
0000-0002-1601-7689 (N. Lazzari); 0000-0002-9380-5160 (V. Presutti)

subset of local tempo estimation, since it can be extracted from a set of local estimations, for instance by taking the median of the local estimations [1].

Audio representation is the most used input representation for the tempo estimation task. Despite the large number of datasets proposed, obtaining high-quality recordings represents the main issue in optimising and measuring Music Information Retrieval (MIR) methods [1]. This is mainly due to copyright issues. It is not possible to directly share high-quality audio samples and it is often impossible to obtain the same exact recording version and reproduce results.

This has led to a growing interest in the use of symbolic annotations, which can be openly shared and have been shown to outperform audio-based methods in chord recognition [7] and music generation [8, 9] tasks. In order to effectively exploit symbolic data, however, annotations need to be provided in a coherent form. Symbolic datasets annotated by experts (e.g. [10, 11]) are often difficult to combine. Among the many open challenges, a prominent issue is that annotations are mostly provided using absolute timing (in seconds) rather than the corresponding symbolic notation.

To the best of our knowledge, there are no previous attempts to perform tempo estimation based on symbolic annotations that use absolute timing. This hampers the harmonisation of heterogeneous symbolic datasets, such as ChoCo [12], since it is not directly possible to normalise all the annotations to rhythmic notation. This poses a limit on the development of methods that can benefit from symbolic representations. An accurate tempo estimations method enables the estimation of the meter of a composition [13, 14] which trivially allows the inference of the rhythmic notation. Alongside the enhancement of other MIR methods, it would also enable experts to analyse large music corpora, such as done by De Clercq and Temperley [15].

In this paper, we propose a novel method that estimates the tempo of a musical composition from its symbolic annotation expressed in absolute timing. Our method is based on works that model tempo from a cognitive [16, 17] or musicological [18, 19] perspective. The core intuition is to formulate a set of hypotheses and identify the one that best fits an annotation, similarly to the work of Grosche and Müller [20]. By exploiting techniques from the Computer Vision field, we estimate the local tempo of each annotation and extract a global estimation from them. Our method outperforms all the related works in this task, reaching an accuracy of 71%.

The rest of this paper is organised as follows: in Section 2 we analyse the most relevant related works; in Section 3 we describe our model; in Section 4 we introduce the experimental setting and in Section 5 we show the obtained results and how they compare to existing methods. Finally, in Section 6 we summarise the proposed work and highlight possible extensions in future works.

2. Related Works

The task of tempo estimation is a prolific research area in the Music Information Retrieval (MIR) field, driven by multiple real-world applications [1] that can be grouped in performance analysis, perceptual modelling, audio analysis, and performance synchronization

[5, 1].

Most of the existing literature focuses on the use of audio representations. Proposed methods conceptually consist in a pipeline involving two steps: the first step processes audio to produce a representation that is then fed to the second step, which extracts the final tempo estimation. Given our focus on symbolic annotations, we only address the most relevant and recent methods, focusing on how tempo extraction (i.e. the second step) is implemented.

A common approach is to use a bank of resonating comb filters to extract the periodicity from the input signal [21, 22, 23, 24] Informally, a resonating comb filter detects the presence of a specific frequency in a signal by summing the signal with a scaled and shifted version of that signal. In this way, the filter is able to resonate at different multiples of the target frequency, thus resulting in a promising method to extract periodicities from a signal. Multiple filters that are tuned to different frequencies, i.e. a bank of filters, is used to detect the most prominent periodicity.

A similar approach is to use autocorrelation with a shifted version of the original signal [25], where the autocorrelation operator computes a self-similarity between all the input time steps. This results in a signal whose peaks correspond to the period of prominent rhythmic groups [25].

Both approaches are effective in identifying frequencies that are repeated within the piece, but encounter difficulties in capturing the near-periodic information that characterises a whole composition,i.e. they neglect the fact that a stable pulse should be assumed for the whole composition. To overcome this issue, the Predominant Local Pulse (PLP) was introduced in Grosche and Müller [20]. A PLP is obtained by sliding sinusoidal kernels all over the signal and accumulating the result. Through the use of kernels with varying frequencies, a mid-level representation that captures local periodic information is obtained. This allows noisy signals that display near-periodic characteristics to be captured by the model. The original PLP proposal [20] uses a short-time Fourier transform to identify periodicities. This makes the method less suited when symbolic annotations are used: the distribution of annotations is uneven time-wise, i.e. some time regions might be very dense of annotations while other regions are much less crowded. Furthermore, it is difficult to identify an optimal trade-off between time and frequency resolution when symbolic annotations are used as input. Our method overcomes the limitation of PLP on symbolic annotations by avoiding the use of the Fourier transform.

Recent solutions implement either the feature extraction step [23, 26, 27, 24] or provide the whole tempo estimation [28, 29, 30, 31] using neural networks.

Despite the accuracy obtained by recent methods [1], they suffer from octave errors [28] (described in Section 4). Extending the time estimation step by using ML algorithms [25] or additional data, such as style [32], has proven to be effective to prevent these errors.

Despite an initial interest in the extraction of tempo estimations from symbolic representations from early methods [5, 33], little interest has been devoted to this task in recent years. This may be due to the representation format itself, since the most popular ones (e.g. MIDI or MusicXML) are designed to represent tempo explicitly. Recent proposals focus on the extraction of more sophisticated rhythmic structures instead, such

as meter detection [14, 34], where global tempo information is assumed to exist.

3. Methodology

We focus on tempo estimation of a musical composition, based on symbolic annotations provided by experts. The underlying assumption in our model is that the tempo of a composition tends to be locally consistent - i.e. neighbouring annotations have a similar tempo. Regardless of tempo fluctuations, we assume that it is possible to identify an over-arching tempo that approximates the neighbourhood of each annotation, similarly to [20].

We test how well each tempo hypothesis explains the annotations by computing the value of a periodic function f at each timestamp, where f is defined as a linear combination of cosine functions $\hat{f}$. The frequency of each cosine function $\hat{f}$ is set such that each peak matches the frequency of the hypothesised beats per minute (BPM) or a multiple of it. In practice, we convert a BPM b into $^{rad}/_{s}$ using the equation

$$\hat{b} = \frac{2\pi\phi b}{60} \tag{1}$$

where $\phi \in \mathbb{N}$ and $\hat{b}$ is b in $^{rad}/_{s}$.

Since textual symbolic annotations are most commonly used for chords and sections (e.g. [10, 11]), they require a low rhythmic resolution, such as whole notes, quarter notes or eighth notes. We assume that the tatum - "the smallest time interval between successive notes in a rhythmic phrase" [35] - corresponds to eighth notes. Depending on the application at hand, other rhythmic figure might be more appropriate to be used as tatum, such as sixteenth [35] or thirty-second notes [34]. We compute f as the linear combination of three cosine functions with $\phi \in [1/2, 1, 2]$ where $\phi = 1$ corresponds to quarter notes, $\phi = 1/2$ to half notes and $\phi = 2$ to eighth notes.

The fitness function f is hence defined as

$$f(t) = \alpha\cos^4(tb_1) + \beta\cos^4(tb_2) + \gamma\cos^4(tb_{\frac{1}{2}}) \tag{2}$$

where b_ϕ is the timing hypothesis converted using Equation (1) and α, β, γ are the coefficients for the linear combination. This approach corresponds to the event rule in the Generative Theory of Tonal Music (GTTM) [18, 19], which states that beats that align with event onsets should be preferred over other beats. Here the peaks of f are the beats and the time of each annotation are the event onsets.

Figure 1 depicts the function f computed for $b = 120$ BPM. The onsets occurring at correct beat positions have higher values when compared to neighbouring positions.

3.1. Estimating tempo

It is possible to optimise f by maximising the sum of f computed at each time step. This requires an additional assumption that reduces the generality of the method: a global BPM must exist for each composition. This method would struggle in those

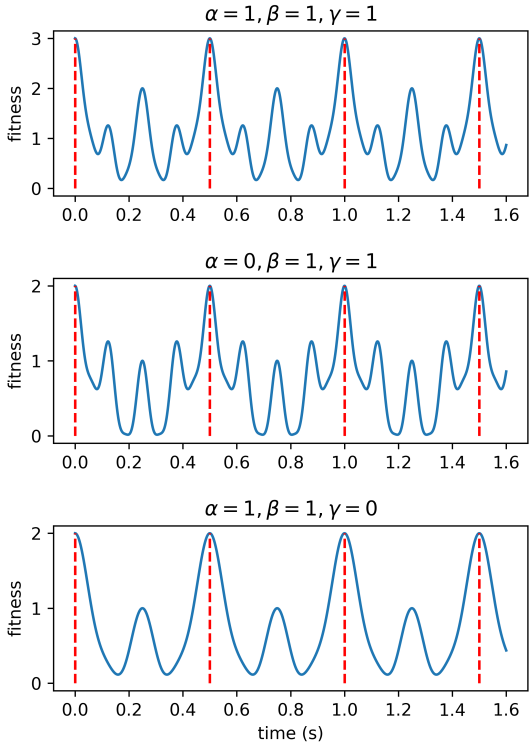


Figure 1: f computed at 120 BPM for different combinations of α, β, γ. The dotted lines represent onset positions while the solid line represents the fitness score of each onset.

cases in which tempo changes throughout the whole piece (e.g. live performances). A straightforward solution is to optimise over sliding windows of a composition. Empirically we observe that this process is too sensitive to the initial tempo hypothesis. While it is possible to obtain an initial hypothesis using a data-driven approach, for instance by using the composition's genre [32], the result would be heavily influenced by the tempo bias on the training data [1].

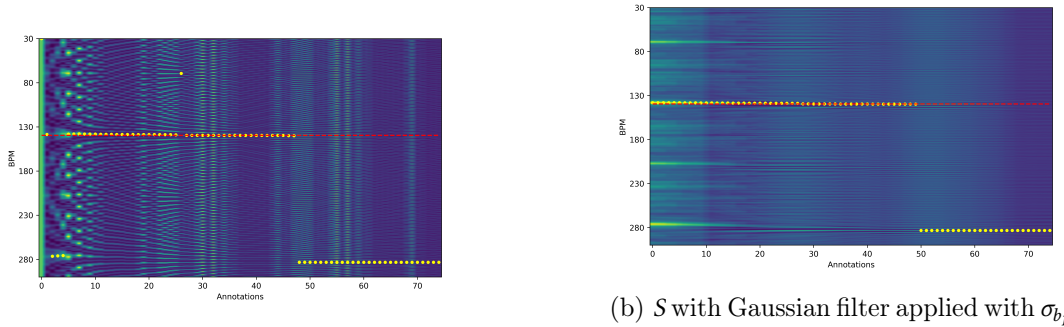
To estimate local tempo, we identify a search space composed of a finite number of hypotheses $H \subseteq [m_{bpm}, M_{bpm}]$ where $|H| \in \mathbb{N}$ and m_{bpm} and M_{bpm} are the bounds of the search space. We identify an hypothesis resolution Δt and sample $\frac{M_{bpm} - m_{bpm}}{\Delta t}$ equally distant points in the search space H. The choice of Δt depends on the trade-off between computational complexity and precision of the solution, since it influences the dimension of the search space. Intuitively, lower values of Δt produce higher resolution results while higher values of Δt result in lower complexity of the search procedure. We investigate the influence of the parameters m_{bpm}, M_{bpm}, and Δt, in Section 4.

For each annotation a we compute the fitness f of each hypothesis $h \in H$. The result is a matrix $S \in \mathscr{R}^{|H| \times |a|}$ where $|a|$ is the number of annotated timestamps in a. S can

be interpreted as an image describing the fitness of each hypothesis at each available timestamp. So far, the described method is similar to PLP [20].

3.2. Local tempo coherency

Differently from PLP and according to our initial assumption, we update S such that neighbouring annotations and neighbouring BPMs influence each other. We use two Gaussian filters [36] over S, one along the rows and one along the columns. A Gaussian filter is obtained by computing the convolution of an image, in our case S, with a Gaussian kernel. Informally, each element $s \in S$ is updated by computing a weighted mean of the neighbouring elements, where the weights are a $2D$ Gaussian distribution centred at s. The standard deviation of the Gaussian distribution, σ, is used to compute the size of the neighbourhood, roughly 3σ. This approach overcomes the limit of the PLP method when applied to symbolic annotations. The application of two distinct filters results in two additional parameters: σ_{bpm} and σ_t. The first (σ_{bpm}) is used to compute the dimension of the kernel along the timing dimension, and the second (σ_t) takes into account the BPM resolution Δt: $\sigma_{bpm} = \sigma_{bpm}/\Delta t$.



(a) Standard S

(b) S with Gaussian filter applied with $\sigma_{bpm} = 1$ and $\sigma_t = 3$.

Figure 2: Matrix S computed for *Helter Skelter* by *The Beatles*, with $\Delta t = 0.1, m_{bpm} = 30, M_{bpm} = 300$. In both figures, the correct BPM has been highlighted with a dashed red line while the detected local BPMs are represented with yellow dots. In Figure b it is easier to visually identify high-scoring hypotheses when compared to Figure a. In both images the global BPM is correctly identified, however, the local predictions depicted in Figure b are more stable across annotations.

In Figure 2 a visual comparison between an unfiltered S (a) and a filtered S (b) is shown.

We obtain a local tempo estimation from S by computing the cumulative sum over each column and taking the maximum for each row. Formally, this is defined as

$$score(t) = \underset{j}{\operatorname{argmax}} \sum_{i=1}^{|H|} S_{i,j} \tag{3}$$

with t the annotations in a composition and j the annotations preceding t. We compute the global tempo by taking either the median value of the local estimations or the peak from

Schema	Definition
Uniform	$w(bpm) = 1$
Gaussian	$w(bpm) = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp(-\frac{(bpm-\mu)^2}{2\sigma^2})$
Parncutt [16]	$w(bpm) = \exp(-\frac{1}{2}(\frac{1}{\sigma}\log_{10}(\frac{bpm}{\mu}))^2)$
van Noorden et. al [17]	$w(bpm) = \frac{1}{\sqrt{((\frac{60}{bpm})^2-(\frac{60}{ext})^2)^2-\beta\cdot(\frac{60}{ext})^2}}$

Table 1
Implemented weighting schemas w as function of an input BPM bpm. σ, μ, ext are parameters that need to be tuned.

the histogram of the local estimations, as suggested in [1]. In Section 4 we experiment with both methods.

3.3. Perceptionally-based weight for BPM hypothesis

A common issue among all tempo estimation methods is the octave error, i.e. the estimation of a multiple of the actual BPM. An example can be seen on Figure 2b: the local estimations are coherent only until the 50th annotation, where a BPM twice the correct one is detected.

Octave errors are intrinsic to the tempo estimation task: for example the piece in Figure 2b, Helter Skelter by The Beatles, has a final section which is faster and more upbeat when compared to the previous sections. This can results in a denser time distribution of the annotations that leads to the detection of a faster tempo.

To overcome this problem we update the fitness function f to weight specific hypothesis differently. We experiment with 4 different weighting schemas: a uniform distribution, a Gaussian distribution, the model proposed by Parncutt [16] and the model proposed by van Noorden & Moelants[1] [17]. The fitness function f is hence updated to

$$f_w(t) = f(t) \cdot w(bpm) \tag{4}$$

where $w(bpm)$ represents the weighting schema, normalised in the range $[0, 1]$ using Laplace smoothing [37].

The implemented weighting schemas are described in 1. In Section 4 we experiment with different combinations of parameter values.

3.4. Metric preferences

In the GTTM, Lerdahl and Jackendoff define a rule for rhythmic grouping: longer onsets should align with strong beats [18, 19]. The definition of strong beats depends on the meter of a composition - defined as the hierarchical organisation of beats at different time scales [3]. We take in consideration this rule by extending Equation (4) as follows:

$$\mathbf{f}_{wg}(t, d) = [\ f_w(t) + \cos^2(tb_g) \cdot d\ , \cdots\ ] \tag{5}$$

[1]Our implementation is slightly different from the original proposal. See 1 for more details.

where $\mathbf{f}_{wg}(t, d)$ is a vector whose elements are hypotheses specialised to a metric preference g; d is the length of the annotation at time t, and b_g is the BPM hypothesis converted using Equation (1) with $\phi = g$. In the experiment presented in Section 4 we set $g \in [3, 4]$ to represent ternary and binary meters respectively, but it is trivial to support additional groupings as well. To detect the meter that best fits the composition we maximise g in Equation (5). Formally, this is defined as follows

$$\underset{g}{\operatorname{argmax}} \sum_{t=0}^{|a|} \mathbf{f}_{wg}(t) + \cos^2(tb_1) \cdot d \tag{6}$$

4. Experimental Setup

In this section, we experiment with the different combinations of methods and parameters, described in Section 3. We perform an extensive set of experiments relying on Bayesian Search [38] to find the best combination of parameters in a complete and efficient way. The global tempo estimation methods described in Section 3.1 and the weighting schemas of Section 3.3 are treated as parameters of the model.

Component	Search space
$f(t)$	$\Delta t \in [0.1, 1]$ $m_{bpm} \in [10, 50]$ $M_{bpm} \in [180, 300]$ $\alpha, \beta, \gamma \in [0, 1]$
Smoothing	$\sigma \in [0.5, 10]$
Gaussian weight	$\mu \in [50, 250]$ $\sigma \in [1, 1000]$
Parncutt [16]	$\mu \in [10, 300]$ $\sigma \in [0.1, 10]$
van Noorden et al. [17]	$ext \in [50, 250]$ $\beta \in [0.1, 10]$

Table 2
Search space for the proposed model

2, provides an overview of the identified search spaces.

We compare our model with the optimisation methods sketched in Section 3. We use the Nelder-Mead algorithm [39] (based on gradients) with the initial solution set to $0.5 * (M_{bpm} - m_{bpm})$, and the Particle Swarm Optimisation (PSO, free from gradients) method [40] to minimise the objective function. To provide a fair comparison, we search for the best parameter ($m_{bpm} \in [10, 50]$, $M_{bpm} \in [180, 300]$ and $\Delta w \in [1, 10]$, with Δw the sliding window size) on the PSO model as well, including the hyper-parameters ($\alpha, \beta, \gamma \in [0, 1]$) in the search space as well.

Finally, we compare our results with other publicly available methods: Böck et al. [23], Böck et al. [41], and Grosche et al. [20]. They use, respectively, comb filters,

autocorrelation and PLP. Each method is either implemented using the madmom [42] or essentia [43] libraries. All the related methods expect a signal representation as input. Given our setting, we construct a signal with sampling rate $f_s = 200Hz$ and manually add peaks at the samples corresponding to each annotation.

Each result is compared using the standard measures of Accuracy and Formal Octave Errors (FOE). Accuracy is divided into two measures, Accuracy 1 and 2, defined as:

$$0.96 \cdot \alpha \cdot est < bpm < 1.04 \cdot \alpha \cdot est \tag{7}$$

where bpm is the correct BPM, est the estimation, and $\alpha = 1$ for Accuracy 1 and $\alpha \in [\frac{1}{3}, \frac{1}{2}, 1, 2, 3]$ for Accuracy 2. Both accuracy measures are binary measures: the estimate is correct if it is within a 4% tolerance with respect to the true BPM. Differently from Accuracy 1, Accuracy 2 considers octave errors as correct estimations. As pointed out by Schreiber et al. in [1], Accuracy 1 and 2 are of difficult interpretation: important information, such as the most common octave errors, are hidden by the binary result. We address this issue by analysing the FOE measures:

$$\begin{aligned} OE_1(est, bpm) &= \log_2(\frac{est}{bpm}) \\ OE_2(est, bpm) &= \operatorname{argmin} OE_1(\alpha \cdot est, bpm) \\ AOE_1(est, bpm) &= |OE_1(est, bpm)| \\ AOE_2(est, bpm) &= |OE_2(est, bpm)| \end{aligned}$$

where α is the same as the one for Accuracy 2. FOE measures are complementary to Accuracy 1 and 2 and are easier to be visually interpreted. The search procedure optimises Accuracy 1 over a subset of the Beatles [10] and RWC Pop [11] datasets provided by the mir_data library [44]. We use 3-fold cross-validation on a subset of the data (80%) randomly sampled and use the remaining data to evaluate the method in Section 5.

5. Results

In Table 3 the best results obtained from the experiments described in Section 4 are described. Our method outperforms existing techniques on Accuracy 1, providing estimations that are less flawed by octave errors. We obtain our best results by estimating global tempo using the median operator. This provides additional evidence that this operator is best suited to estimate global tempo from a list of local tempos, as also argued by others [1]. Interestingly, when it comes to Accuracy 2, the approach from Böck et al. [23] (comb filter-based) achieves good results, outperforming some of our experiments. The lower Accuracy 1 score, however, indicates that it is not a reliable method for symbolic annotations. Using a weighting schema, as described in Section 3.3, correctly biases the method towards octave-correct estimations, as in the case of Gaussian weighting, which largely improves Accuracy 2 results. When using the work

	Weighting schema	α	β	γ	m_{bpm}	M_{bpm}	Δt	σ_{bpm}	σ_t	A1	A2
Histogram	Nelder-Mead$_{\Delta t=7}$	0.18	0.56	0.48	10	271				0.17	0.23
	PSO$_{\Delta t=4,c_1=0.69,c_2=0.45,w=0.63}$	0.93	0.27	0.58	14	184				0.02	0.08
	Gaussian$_{\mu=103,\sigma=168.44}$	0.17	0.82	0.64	27	241	0.11	7.87	0.66	**0.67**	0.85
	Parncutt$_{\mu=93,\sigma=4.12}$	0.40	0.25	0.55	31	279	0.12	6.39	1.57	0.63	0.83
	Resonance$_{\beta=8.76,ext=86}$	0.56	0.46	0.29	50	276	0.15	1.56	6.19	0.60	0.75
	Uniform	0.47	0.35	0.91	48	240	0.13	9.27	9.67	0.48	**0.88**
Median	Nelder-Mead$_{\Delta t=6}$	0.13	0.85	0.58	21	277				0.13	0.15
	PSO$_{\Delta t=3,c_1=0.97,c_2=0.5,w=0.41}$	0.14	0.89	0.79	47	184				0.06	0.06
	Gaussian$_{\mu=99,\sigma=890.26}$	0.86	0.43	1	40	242	0.12	9.79	2.32	0.67	**0.90**
	Parncutt$_{\mu=109,\sigma=7.83}$	0.21	0.54	0.56	41	297	0.10	6.31	6.88	**0.71**	0.85
	Resonance$_{\beta=0.34,ext=96}$	0.70	0.34	0.82	50	296	0.21	1.32	1.83	0.67	0.79
	Uniform	0.01	0.76	0.58	39	240	0.29	8.51	4.75	0.40	0.88
Baseline	Böck et al. [41] (*autocorrelation*)									0.11	0.67
	Böck et al. [23] (*comb filters*)									0.09	0.88
	Grosche et al. [20] (*PLP*)									0.11	0.46

Table 3
Results of the proposed methods and their best parameters found. Results from related works are also reported. The best results for each global BPM estimation method (histogram and median) are represented in bold. The best results overall are also underlined. A1 and A2 are used to refer to Accuracy 1 and Accuracy 2. *Parncutt* weighting schema refers to the work of Parncutt [16] while *Resonance* to the work of van Noorden et al. [17].

from van Noorden et al. [17] and Parncutt [16], Accuracy 1 improvements over a uniform distribution also result in a degraded Accuracy 2 score. This might happen because these methods bias the tested hypotheses in a more aggressive way. A possible approach to overcome this issue is to dampen the amount of added bias through an additional parameter.

In general, we consider the method that uses Parncutt weighting and the median operator to be our best-performing experiment, given the Accuracy 1 result of 0.71. We remark that regardless of the use of a numerical method (Nelder-Mead) or a meta-heuristic one (PSO), the use of optimisation methods show worse performance in comparison to other approaches.

In Figure 3 the measures describing octave errors are reported. The distribution of OE_1 in the best model is centred towards 0 when compared to other models. Analogously in OE_2, the distribution is more accurate when octave errors are also considered correct. From the OE_1 graph, it can be seen that related works are completely skewed towards octave errors since the plot distribution is distributed along all the x-axis, while all of our models are more prone to estimate BPMs that are 1/2 of the target BPM, since denser clusters can be identified around the point -1 [1].

In Figure 4 the accuracy of our best methods, represented in Table 3, is plotted as a function of the tolerance (4% in Equation Equation 7). The median approach converges much quicker to the best results, providing further evidence that it is best suited to

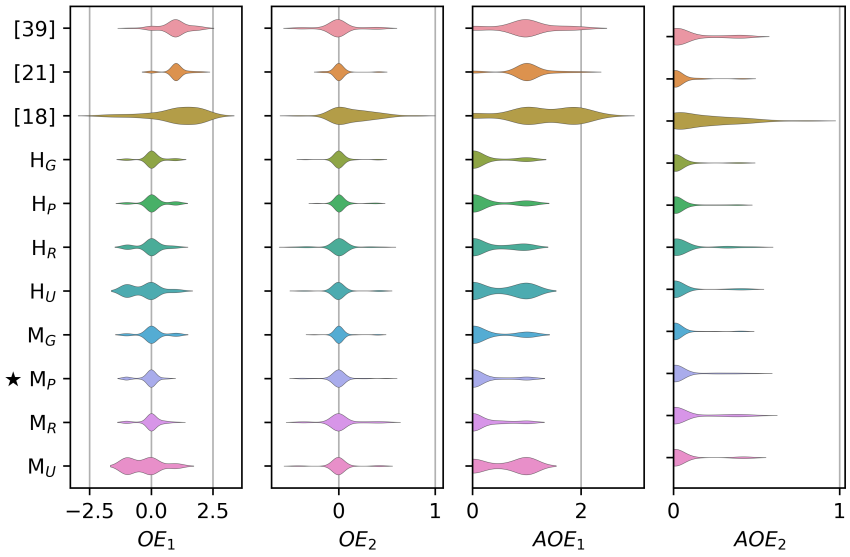


Figure 3: Octave errors from the experiments on the proposed method in Table 3, visualised using Kernel Density Estimation. The best result is highlighted using a star. Each model is in a different row, labelled as E_W where E is the global tempo estimation method (M for median and H for histogram) and W is the weighting schema (G for Gaussian, U for Uniform, P for the Parncutt model and R for the van Noorden et al. model).

extract a global tempo estimation.

6. Conclusion

We propose a novel method for tempo estimation of music composition, based on symbolic text annotations, as input. The core idea is to exploit the linear combination of periodic functions and techniques from the computer vision field, to find a BPM that best explains the annotations. By relying on existing works from computational musicology and cognitive perception of music, we devise a methodology that reaches an accuracy of 71%, outperforming all existing approaches, applicable to this task.

In Section 3 we propose a variation of our method using optimisation techniques to obtain a tempo estimation. Even though the results are not comparable with our other approaches, we argue that framing the tempo estimation task as an optimisation and carefully designing an objective function can lead to robust and accurate methods.

Regardless of the method used to obtain local tempo estimations, our results provide additional evidence that the median operator is the best way to estimate global tempo from a list of local estimations. The histogram method, however, can still be incorporated into our approach when estimating local tempo. Instead of solving the maximisation problem formulated in Equation (6), the histogram operator can be used to retrieve a list of top-k candidates for each time step. We will investigate this option in future works.

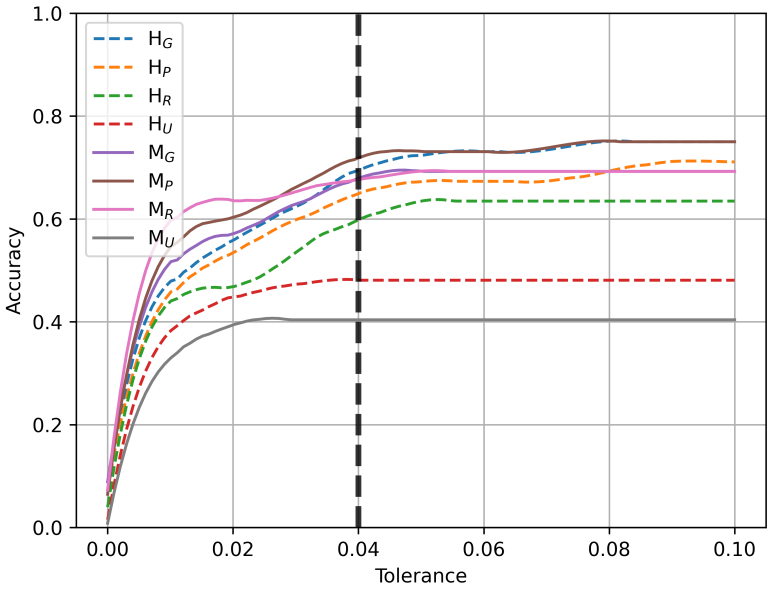


Figure 4: Plot of Accuracy 1 with varying tolerance $\in [0, 0.1]$. Model names are the same of Figure 3

Finally, given the promising results from Böck et al. [23] in Table 3, an interesting approach to explore is the combination of comb filters with our proposed method, to enhance the performance on audio-representation as well.

7. Acknowledgments

This project has received funding from the FAIR Future Artificial Intelligence Research foundation as part of the grant agreement MUR n. 341.

References

[1] H. Schreiber, J. Urbano, M. Müller, Music tempo estimation: Are we done yet?, Trans. Int. Soc. Music. Inf. Retr. 3 (2020) 111. URL: https://doi.org/10.5334/tismir.43. doi:10.5334/tismir.43.

[2] F. Gouyon, S. Dixon, Dance music classification: A tempo-based approach, in: ISMIR 2004, 5th International Conference on Music Information Retrieval, Barcelona, Spain, October 10-14, 2004, Proceedings, 2004. URL: http://ismir2004.ismir.net/proceedings/p091-page-501-paper151.pdf.

[3] J. D. McAuley, Tempo and Rhythm, University of California Press, 2010, pp. 165–199. doi:10.1007/978-1-4419-6114-3_6.

[4] D. T. Bishop, M. J. Wright, C. I. Karageorghis, Tempo and intensity of pre-task

music modulate neural activity during reactive task performance, Psychology of Music 42 (2014) 714–727. URL: https://doi.org/10.1177/0305735613490595. doi:10.1177/0305735613490595. arXiv:https://doi.org/10.1177/0305735613490595.

[5] S. Dixon, Automatic extraction of tempo and beat from expressive performances, Journal of New Music Research 30 (2001) 39–58. URL: https://www.tandfonline.com/doi/abs/10.1076/jnmr.30.1.39.7119. doi:10.1076/jnmr.30.1.39.7119. arXiv:https://www.tandfonline.com/doi/pdf/10.1076/jnmr.30.1.39.7119.

[6] G. Peeters, Time variable tempo detection and beat marking, in: Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005, Michigan Publishing, 2005. URL: https://hdl.handle.net/2027/spo.bbp2372.2005.186.

[7] T.-P. Chen, L. Su, Attend to chords: Improving harmonic analysis of symbolic music using transformer-based models, Transactions of the International Society for Music Information Retrieval (2021). doi:10.5334/tismir.65.

[8] M. Zeng, X. Tan, R. Wang, Z. Ju, T. Qin, T.-Y. Liu, MusicBERT: Symbolic music understanding with large-scale pre-training, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 791–800. URL: https://aclanthology.org/2021.findings-acl.70. doi:10.18653/v1/2021.findings-acl.70.

[9] D. von Rütte, L. Biggio, Y. Kilcher, T. Hofmann, FIGARO: controllable music generation using learned and expert features, in: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, OpenReview.net, 2023. URL: https://openreview.net/pdf?id=NyR8OZFHw6i.

[10] M. Mauch, C. Cannam, M. Davies, S. Dixon, C. Harte, S. Kolozali, D. Tidhar, M. Sandler, Omras2 metadata project 2009, in: 12th International Society for Music Information Retrieval Conference, ISMIR, 2009.

[11] T. Cho, J. P. Bello, A feature smoothing method for chord recognition using recurrence plots, in: 12th International Society for Music Information Retrieval Conference, ISMIR, 2011.

[12] J. de Berardinis, A. Meroño-Peñuela, A. Poltronieri, V. Presutti, Choco: a chord corpus and a data transformation workflow for musical harmony knowledge graphs, Scientific Data 10 (2023) 641. URL: https://doi.org/10.1038/s41597-023-02410-w. doi:10.1038/s41597-023-02410-w.

[13] A. Volk, The study of syncopation using inner metric analysis: Linking theoretical and experimental analysis of metre in music, Journal of New Music Research 37 (2008) 259–273. URL: https://doi.org/10.1080/09298210802680758. doi:10.1080/09298210802680758. arXiv:https://doi.org/10.1080/09298210802680758.

[14] W. B. de Haas, A. Volk, Meter detection in symbolic music using inner metric analysis, in: M. I. Mandel, J. Devaney, D. Turnbull, G. Tzanetakis (Eds.), Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016, New York City, United States, August 7-11, 2016, 2016, pp. 441–447. URL: https://wp.nyu.edu/ismir2016/wp-content/uploads/sites/2294/2016/07/033_Paper.pdf.

[15] T. De Clercq, D. Temperley, A corpus analysis of rock harmony, Popular Music 30 (2011) 47–70.

[16] R. Parncutt, A Perceptual Model of Pulse Salience and Metrical Accent in Musical Rhythms, Music Perception 11 (1994) 409–464. URL: https://doi.org/10.2307/40285633. doi:10.2307/40285633. arXiv:https://online.ucpress.edu/mp/article-pdf/11/4/409/145282/40285633.pdf.

[17] L. van Noorden, D. Moelants, Resonance in the perception of musical pulse, Journal of New Music Research 28 (1999) 43–66. URL: https://www.tandfonline.com/doi/abs/10.1076/jnmr.28.1.43.3122. doi:10.1076/jnmr.28.1.43.3122. arXiv:https://www.tandfonline.com/doi/pdf/10.1076/jnmr.28.1.43.3122.

[18] D. Temperley, D. D. Sleator, Modeling meter and harmony: A preference-rule approach, Comput. Music. J. 23 (1999) 10–27. URL: https://doi.org/10.1162/014892699559616. doi:10.1162/014892699559616.

[19] F. Lerdahl, R. S. Jackendoff, A Generative Theory of Tonal Music, The MIT Press, 1996. URL: https://doi.org/10.7551/mitpress/12513.001.0001. doi:10.7551/mitpress/12513.001.0001.

[20] P. Grosche, M. Müller, A mid-level representation for capturing dominant tempo and pulse information in music recordings, in: K. Hirata, G. Tzanetakis, K. Yoshii (Eds.), Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe International Conference Center, Kobe, Japan, October 26-30, 2009, International Society for Music Information Retrieval, 2009, pp. 189–194. URL: http://ismir2009.ismir.net/proceedings/OS2-3.pdf.

[21] E. D. Scheirer, Tempo and beat analysis of acoustic musical signals, The Journal of the Acoustical Society of America 103 (1998) 588–601.

[22] A. Klapuri, A. J. Eronen, J. Astola, Analysis of the meter of acoustic musical signals, IEEE Trans. Speech Audio Process. 14 (2006) 342–355. URL: https://doi.org/10.1109/TSA.2005.854090. doi:10.1109/TSA.2005.854090.

[23] S. Böck, F. Krebs, G. Widmer, Accurate tempo estimation based on recurrent neural networks and resonating comb filters, in: M. Müller, F. Wiering (Eds.), Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, Spain, October 26-30, 2015, 2015, pp. 625–631. URL: http://ismir2015.uma.es/articles/196_Paper.pdf.

[24] S. Böck, M. E. P. Davies, Deconstruct, analyse, reconstruct: How to improve tempo, beat, and downbeat estimation, in: J. Cumming, J. H. Lee, B. McFee, M. Schedl, J. Devaney, C. McKay, E. Zangerle, T. de Reuse (Eds.), Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020, Montreal, Canada, October 11-16, 2020, 2020, pp. 574–582. URL: http://archives.ismir.net/ismir2020/paper/000223.pdf.

[25] G. Percival, G. Tzanetakis, Streamlined tempo estimation based on autocorrelation and cross-correlation with pulses, IEEE ACM Trans. Audio Speech Lang. Process. 22 (2014) 1765–1776. URL: https://doi.org/10.1109/TASLP.2014.2348916. doi:10.1109/TASLP.2014.2348916.

[26] S. Böck, M. E. P. Davies, P. Knees, Multi-task learning of tempo and beat: Learning one to improve the other, in: A. Flexer, G. Peeters, J. Urbano, A. Volk (Eds.), Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019, 2019, pp.

486–493. URL: http://archives.ismir.net/ismir2019/paper/000058.pdf.

[27] H. F. Aarabi, G. Peeters, Deep-rhythm for global tempo estimation in music, in: A. Flexer, G. Peeters, J. Urbano, A. Volk (Eds.), Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019, 2019, pp. 636–643. URL: http://archives.ismir.net/ismir2019/paper/000077.pdf.

[28] H. Schreiber, M. Müller, A single-step approach to musical tempo estimation using a convolutional neural network, in: E. Gómez, X. Hu, E. Humphrey, E. Benetos (Eds.), Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018, 2018, pp. 98–105. URL: http://ismir2018.ircam.fr/doc/pdfs/141_Paper.pdf.

[29] H. Schreiber, M. Müller, Musical tempo and key estimation using convolutional neural networks with directional filters, CoRR abs/1903.10839 (2019). URL: http://arxiv.org/abs/1903.10839. arXiv:1903.10839.

[30] M. S. de Oliveira de Souza, P. N. de Souza Moura, J. Briot, Music tempo estimation via neural networks - A comparative analysis, CoRR abs/2107.09208 (2021). URL: https://arxiv.org/abs/2107.09208. arXiv:2107.09208.

[31] X. Sun, Q. He, Y. Gao, W. Li, Musical tempo estimation using a multi-scale network, in: J. H. Lee, A. Lerch, Z. Duan, J. Nam, P. Rao, P. van Kranenburg, A. Srinivasamurthy (Eds.), Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR 2021, Online, November 7-12, 2021, 2021, pp. 682–689. URL: https://archives.ismir.net/ismir2021/paper/000085.pdf.

[32] F. Hörschläger, R. Vogl, S. Böck, P. Knees, Addressing tempo estimation octave errors in electronic music by incorporating style information extracted from wikipedia, in: Proceedings of the Sound and Music Computing Conference (SMC), Maynooth, Ireland, 2015.

[33] H. Takeda, T. Nishimoto, S. Sagayama, Rhythm and tempo analysis toward automatic music transcription, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2007, Honolulu, Hawaii, USA, April 15-20, 2007, IEEE, 2007, pp. 1317–1320. URL: https://doi.org/10.1109/ICASSP.2007.367320. doi:10.1109/ICASSP.2007.367320.

[34] A. McLeod, M. Steedman, Meter detection and alignment of MIDI performance, in: E. Gómez, X. Hu, E. Humphrey, E. Benetos (Eds.), Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018, 2018, pp. 113–119. URL: http://ismir2018.ircam.fr/doc/pdfs/136_Paper.pdf.

[35] A. Klapuri, Musical meter estimation and music transcription, in: Cambridge Music Processing Colloquium, Citeseer, 2003, pp. 40–45. doi:https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.77.8559.

[36] L. G. Shapiro, G. C. Stockman, Computer vision, Pearson, 2001.

[37] C. D. Manning, P. Raghavan, H. Schütze, Introduction to information retrieval, Cambridge University Press, 2008. URL: https://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf. doi:10.1017/CBO9780511809071.

[38] J. Snoek, H. Larochelle, R. P. Adams, Practical bayesian optimization of ma-

chine learning algorithms, in: P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States, 2012, pp. 2960–2968. URL: https://proceedings.neurips.cc/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html.

[39] F. Gao, L. Han, Implementing the nelder-mead simplex algorithm with adaptive parameters, Comput. Optim. Appl. 51 (2012) 259–277. URL: https://doi.org/10.1007/s10589-010-9329-3. doi:10.1007/s10589-010-9329-3.

[40] K. Hussain, M. N. M. Salleh, S. Cheng, Y. Shi, Metaheuristic research: a comprehensive survey, Artif. Intell. Rev. 52 (2019) 2191–2233. URL: https://doi.org/10.1007/s10462-017-9605-z. doi:10.1007/s10462-017-9605-z.

[41] S. Böck, M. Schedl, Enhanced beat tracking with context-aware neural networks, in: Proc. Int. Conf. Digital Audio Effects, 2011, pp. 135–139.

[42] S. Böck, F. Korzeniowski, J. Schlüter, F. Krebs, G. Widmer, madmom: a new Python Audio and Music Signal Processing Library, in: Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 2016, pp. 1174–1178. doi:10.1145/2964284.2973795.

[43] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. R. Zapata, X. Serra, ESSENTIA: an open-source library for sound and music analysis, in: A. Jaimes, N. Sebe, N. Boujemaa, D. Gatica-Perez, D. A. Shamma, M. Worring, R. Zimmermann (Eds.), ACM Multimedia Conference, MM '13, Barcelona, Spain, October 21-25, 2013, ACM, 2013, pp. 855–858. URL: https://doi.org/10.1145/2502081.2502229. doi:10.1145/2502081.2502229.

[44] R. M. Bittner, M. Fuentes, D. Rubinstein, A. Jansson, K. Choi, T. Kell, mirdata: Software for reproducible usage of datasets, in: A. Flexer, G. Peeters, J. Urbano, A. Volk (Eds.), Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019, Delft, The Netherlands, November 4-8, 2019, 2019, pp. 99–106. URL: http://archives.ismir.net/ismir2019/paper/000009.pdf.