PHENICX

D4.2 Methods for automatic alignment of performances to a score representation

Grant Agreement nr	601166
Project title	Performances as Highly Enriched aNd Interactive Concert eXperiences
Project acronym	PHENICX
Start date of project (dur.)	Feb 1st, 2013 (3 years)
Document reference	PHENICX-D-WP4-OFAI-140113-D4.2_ScorePerformanceAlignment-2.1
Report availability	PU - Public
Document due Date	Feb 1st, 2014
Actual date of delivery	Jan 24th, 2014
Leader	OFAI
Reply to	Maarten Grachten (OFAI) (maarten.grachten@ofai.at)
Additional main contributors (authors name / partner acr.)	Gerhard Widmer (OFAI) Martin Gasser (OFAI) Andreas Arzt
Document status	Final

Project funded by ICT-7th Framework Program from the European Commission

PHENICX

Table of Contents

1 Introduction **3**
1.1 Main objective 3
1.2 Executive summary 3

2 Overview of the score-performance alignment method **4**

3 Conclusion **5**

A Published conference paper: Automatic alignment of music performances with structural differences **6**

1 INTRODUCTION

This deliverable reports a method to align audio recordings to the symbolic score representation of the performed piece. This method plays a key role in several of the problems addressed in the PHENICX project. Specifically, accurate score-performance alignments allow for applications such as displaying sheet music synchronously while playing audio/video of recorded performances, and interactively navigating through recorded audio/video, based on the musical score, or annotations/representations derived from the score, such as a structural analysis of the piece. Furthermore, score-performance alignments are the starting point for extracting expressive parameters from recorded performance, such as tempo and loudness curves (see Deliverable 4.1 for more information).

This document is intentionally kept very short, since the essential text describing the method and its evaluation is contained in a peer-reviewed published paper, included under Appendix A. Apart from the goal statement (Section 1.1) and executive summary (Section 1.2), the rest of this document is a brief contextualization of the published work with respect to the deliverable objective (Section 2), and a conclusion, where we point to application areas of the method elsewhere in the PHENICX project, and give some future directions based on this work (Section 3).

1.1 Main objective

The objective of this Deliverable is the development of a method to align audio recordings to the symbolic score representation of the performed piece.

1.2 Executive summary

The purpose of this document is to report a method for automatic alignment of audio recordings of musical performances to a score representation of the performed piece. The alignment method has been evaluated experimentally on a dataset of classical piano music, and was found to yield more accurate alignments than Dynamic Time Warping, an alignment method widely used in the literature. The method provides the basis for several other objectives within the project (specifically in WP4, WP6, and WP7).

2 OVERVIEW OF THE SCORE-PERFORMANCE ALIGNMENT METHOD

The proposed score-performance alignment method was developed as an alternative to traditional alignment methods, in particular dynamic time warping (DTW), as a way to deal with *structural differences* between the data to be aligned. Structural differences between a score and a performance may arise because of performance decisions of the conductor/performer, for instance whether or not to repeat certain parts.

Another important aspect of score-performance alignment is the fact that it involves an alignment of sequential data of distinct categories. In the context of the PHENICX project, the score representation is a symbolic description of the score content, whereas performances are typically only available in the form of audio (and possibly video) recordings. Since it is not straight-forward to define alignment algorithms that deal with such heterogeneous types of data, we have chosen to develop and employ an audio-to-audio alignment method, in combination with an audio rendering of the score representation. An advantage of this route is that the alignment method can also be used to align two performances, even in the absence of score information.

The audio synthesis process that renders score information for a given piece as audio, is documented in Deliverable 4.1. Given the synthesized audio and the audio recording of a real performance, we align both audio files using the method described in the paper. The resulting alignment consists of pairs of time positions, mapping positions in the synthesized audio to positions in the recorded performance. Subsequently, time positions in the synthesized audio are converted to musical time (i.e. offsets in musical beats from the beginning of the piece), such that the alignment can be used both to index the recorded performance by musical time, and to index the score by performance time.

3 CONCLUSION

The presented score-performance alignment method allows for alignments of performance to score representations, even in the presence of structural differences. The experiments on a dataset of classical piano music, reported in the paper, show that our method performs on a par with DTW on data without structural differences, and outperforms DTW when structural differences do occur between the data to be aligned. Due to lack of ground truth data, this result has not yet been confirmed with audio recordings of symphonic music, but informal comparisons of alignment paths look promising.

As stated in the introduction, the automatic score-performance alignment method is an important stepping stone for several other tasks in the PHENICX project. Firstly, it will facilitate the process of extending the database of expressive performance parameters (see Deliverable 4.1). Secondly, WP6 (Task 6.1) involves visualization of a variety of score related aspects of the music, like piece structure, harmony, and tonality. Through score-performance alignment, these visualizations can be shown in synchrony with audio recordings of the performance. Lastly, in WP7 (Task 7.2) promising techniques developed in the context of PHENICX are implemented in the products and services of PHENICX partner *VideoDock*. One of those products is *RCO Editions*, an iPad magazine released on a two-month basis, that contains enriched multimedia material of recorded performances by PHENICX partner the *Royal Concertgebouw Orchestra*. At the time of writing, score-performance alignments produced with the method presented here are being used to integrate scoreviewing functionality into one of the upcoming RCO Editions.

To conclude, we present some possible continuations to extend the score-alignment method and improve its functionality. Firstly, an alignment editor tool (with a graphical user interface) is being developed to manually correct errors that may arise in automatic alignment. Secondly, although the presented alignment method deals with structural differences better than standard DTW, it does not explicitly identify repetitions in a piece (which is a major source of structural differences between score and performance). To alleviate this, the score-performance alignments may be processed further to identify repetitions. Finally, improved score-performance alignments may be obtained by employing predictive models of musical expression (to be developed in a later stage of the project) in the audio rendering of the score representations.

Appendices

A PUBLISHED CONFERENCE PAPER: AUTOMATIC ALIGNMENT OF MUSIC PERFORMANCES WITH STRUCTURAL DIFFERENCES

What follows is the verbatim paper titled “Automatic alignment of music performances with structural differences”, published in the proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013).[1]

[1] http://www.ppgia.pucpr.br/ismir2013/

AUTOMATIC ALIGNMENT OF MUSIC PERFORMANCES WITH STRUCTURAL DIFFERENCES

Maarten Grachten[1] Martin Gasser[1]
[1]Austrian Research Institute for
Artificial Intelligence (OFAI), Vienna, Austria
http://www.ofai.at/~maarten.grachten

Andreas Arzt[2], Gerhard Widmer[1,2]
[2]Dept. of Computational Perception
Johannes Kepler Universität, Linz, Austria

ABSTRACT

Both in interactive music listening, and in music performance research, there is a need for automatic alignment of different recordings of the same musical piece. This task is challenging, because musical pieces often contain parts that may or may not be repeated by the performer, possibly leading to structural differences between performances (or between performance and score). The most common alignment method, dynamic time warping (DTW), cannot handle structural differences adequately, and existing approaches to deal with structural differences explicitly rely on the annotation of "break points" in one of the sequences. We propose a simple extension of the Needleman-Wunsch algorithm to deal effectively with structural differences, without relying on annotations. We evaluate several audio features for alignment, and show how an optimal value can be found for the cost-parameter of the alignment algorithm. A single cost value is demonstrated to be valid across different types of music. We demonstrate that our approach yields roughly equal alignment accuracies compared to DTW in the absence of structural differences, and superior accuracies when structural differences occur.

1. INTRODUCTION AND RELATED WORK

A variety of music processing scenarios involve alignment of music in the form of either symbolic scores, or audio recordings (or both). In some cases, alignment is used to compute a similarity score between instances of a musical piece. This is useful for example in plagiarism detection [7] and cover song identification [3, 19]. In other cases, it is the alignment itself that is of use. Examples are automatic transcription [20], computer assisted music production [14], real-time score-following for automatic page turning [1], and automatic accompaniment [5, 6].

There are several factors that make accurate alignment of music a challenging task. Firstly, in case of audio alignment, the acoustic properties of the recordings may be very different, due to differences in instrumentation, recording, mixing, and mastering. Secondly, interpretations of musical pieces by human performers tend to have expressive variations, causing different interpretations of a piece to diverge in both global and local tempo and dynamics. Thirdly, performance errors may lead to occasional missing, or inserted notes. A fourth complicating factor is the fact that musical pieces are often composed of smaller musical units, where units may be repeated or not, or even left out completely, according to the taste of the musician or conductor. This may lead to what we refer to as *structural differences* between performances of the piece.

The problem of aligning music with structural differences has been addressed in a number of studies. In most of these, the problem setting is score-to-performance alignment, in which a symbolic representation of a musical score is mapped to a performance of that score. In a symbolic score representation, it is relatively easy to mark points where performances are likely to diverge. For example, Fremerey et al. [9] develop a method that relies on explicit annotations of possible jump points in the score where double bar lines occur. A similar approach is taken by Pardo and Birmingham [18].

In performance-performance alignment, as opposed to score-performance alignment, it is generally not possible to rely on such annotations, since there is no score representation involved. Müller and Appelt [16] propose a method to deal with structural differences in performance-performance alignment. This approach uses dynamic time warping (DTW) in combination with pre-processing of the similarity matrix, and post-processing of alignment paths.

In this paper we start from the observation that DTW has shortcomings when dealing with structural differences in music recordings (Section 2). Our intention is to show that other variants of dynamic programming alignment are more effective. In particular, it is beneficial to include skip operations, as well as one-to-many and many-to-one matching, as in the algorithm of Mongeau and Sankoff [13], who use this approach for measuring similarity between melodies as sequences of notes, and Grachten et al. [10], who design alignment operations to capture the semantics of expressive musical behavior, like spontaneous ornamentations of notes in a performance. To our knowledge, such extensions of the classical dynamic programming variants have not been used in the context of audio alignment.

Along with this alternative alignment method (Section 3.1), we propose a method to estimate the optimal value for

the gap penalty, a parameter that controls the behavior of the alignment (Section 3.3). In Section 4, we experimentally determine the utility of various audio features with respect to the effectiveness of the gap penalty across different types of music. Based on the most successful feature, and the corresponding optimal gap penalty, we perform a quantitative evaluation of the alignment accuracy of our proposed approach, in comparison to DTW (Section 4.2).

2. PROBLEM DESCRIPTION

Aligning two sequences requires a distance measure that quantifies how different their elements are. In Section 4.1.1, we will discuss different types of features and distance measures in more detail in the context of audio alignment. For now, let s and t be the sequences to be compared, of lengths M and N respectively. We refer to s and t as the *source* and *target* sequence, respectively. We use $d(i, j)$ to denote the distance between the i-th element of s and the j-th element of t, where $1 \leq i \leq M$ and $1 \leq j \leq N$.

2.1 Dynamic time warping (DTW)

DTW computes the minimal cost of aligning s and t. It can be expressed as $\text{dtw}(M, N)$, where dtw is defined by the recursive equation:

$$\text{dtw}(i,j) = \begin{cases} 0, & \text{if } i = 0, j = 0 \\ \infty, & \text{if } i = 0, j \neq 0 \text{ or } j = 0, i \neq 0 \\ d(i,j) + \min \begin{cases} \text{dtw}(i-1, j-1)) \\ \text{dtw}(i-1, j) \\ \text{dtw}(i, j-1), \end{cases} & \text{otherwise} \end{cases} \tag{1}$$

The alignment that leads to $\text{dtw}(M, N)$ is called the *optimal alignment*, and can be easily recovered by keeping track which argument of the $\min$ operator is selected in (1).

As the name of the algorithm states, dynamic time warping is a method to align sequences that are *time warped* versions of each other. That means that the sequences represent the same order of events, but the duration of events may differ from one sequence to the other. This time warping assumption explains why in DTW, each element of one sequence must be matched to an element of the other sequence. When the sequences are structurally different however, this assumption is violated: the sequences contain elements that are not to be matched to elements in the other sequence. By forcing a match between elements, DTW produces undesired alignments in such cases.

2.2 Needleman-Wunsch alignment (NW)

A solution to this problem is to allow the alignment algorithm to skip unmatchable parts of either sequence. The cost of skipping should not be proportional to the distances between the elements of the sequences, since these distances are not relevant in the case of unmatchable sequences. This type of alignment is achieved by another member of the family of dynamic programming algorithms for optimal sequence alignment – the Needleman-Wunsch algorithm (NW) [17]. This algorithm computes the minimal cost $\text{nw}(M, N)$ of aligning s and t using the equation:

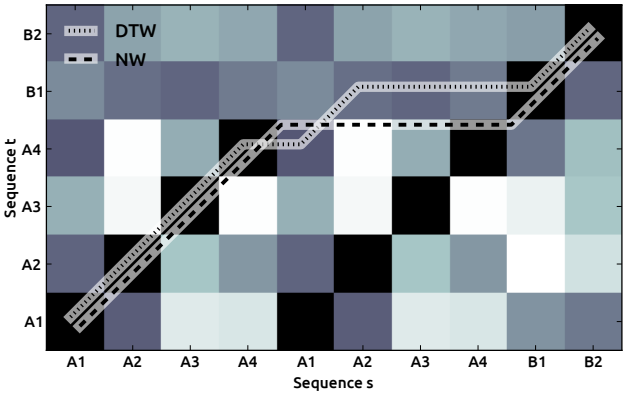

Figure 1. Distance matrix between two structurally different sequences; Dark cells represent low distances, light cells high distances; The DTW path jumps over a repeated section 'uncleanly', the NW path makes a clean jump

$$\text{nw}(i,j) = \begin{cases} 0, & \text{if } i = 0, j = 0 \\ \gamma + \text{nw}(i, j-1), & \text{if } i = 0, j \neq 0 \\ \gamma + \text{nw}(i-1, j), & \text{if } j = 0, i \neq 0 \\ \min \begin{cases} d(i,j) + \text{nw}(i-1, j-1) \\ \gamma + \text{nw}(i-1, j) \\ \gamma + \text{nw}(i, j-1), \end{cases} & \text{otherwise} \end{cases} \tag{2}$$

where γ is a constant referred to as *gap penalty*. (2) shows that the distance $d(i, j)$ between two elements i and j is only relevant to the alignment when it is sufficiently low. As soon as $d(i, j) > \gamma$, the algorithm will favor an insertion or deletion to a match (the final decision for an insertion or a deletion will only be made after the algorithm has processed the sequences entirely).

The difference between DTW and NW is illustrated in Figure 1, displaying the distances between the elements of two artificial sequences s and t, and the optimal DTW and NP alignments. The rows and columns are labeled to clarify the structure of s and t. In particular, s consists of two repetitions of a 4-tuple A, plus a 2-tuple B. Sequence t is the concatenation of one instance of A, and B. The DTW path aligns the elements of the second A in s partly to element $A4$ in t, and partly to element $B1$, where the exact alignment depends on the distances between those (non-matching) elements. The NW path (computed with a suitable value for γ) favors the deletion of the elements of the second occurrence of A over a sequence of poor matches. Note that this yields a clean and intuitive jump of the NW path across the second A in s.

2.3 Two problems of NW alignment

The use of Needleman-Wunsch for aligning music recordings introduces two problems. The first is that although NW handles structural differences, it does not handle *time*

warping. Since elements in the sequences can be either matched to a single other element, or skipped entirely, there is no way to deal with the fact that the music of the two recordings may be played at different tempos. Fortunately, a simple extension of the Needleman-Wunsch algorithm, described in Section 3.1, remedies this shortcoming.

The second problem is that – unlike DTW in its basic form [1] – NW involves a parameter γ, and the quality of the alignment will depend on the value of γ. Which value of γ gives good alignments will depend on the audio content and the features used to represent that content. In Section 3.3, we propose a method to estimate the optimal value for γ based on empirical data. In Section 4, we use this method to evaluate different features on various types of music.

3. PROPOSED SOLUTION

In this section we propose solutions to the two problems of NW alignment described above. Firstly, we propose an extension of the NW algorithm to deal with the time warping aspects of aligning music performances. Secondly, we describe a method to estimate the gap penalty γ.

3.1 Needleman-Wunsch time warping (NWTW)

DTW handles time warping by matching multiple elements of one sequence to a single element in the other sequence. Although this is not possible in the original NW algorithm, it is easy to add further arguments to the $\min$ operator, that represent many-to-one and one-to-many operations. In the following equation, which is a revision of (2), a 1-to-2 and a 2-to-1 operation have been included:

$$\mathrm{nw}(i,j) = \begin{cases} 0, & \text{if } i=0,\, j=0 \\ \gamma + \mathrm{nw}(i, j-1), & \text{if } i=0,\, j\neq 0 \\ \gamma + \mathrm{nw}(i-1, j), & \text{if } j=0,\, i\neq 0 \\ \min \begin{cases} d(i,j) + \mathrm{nw}(i-1, j-1) \\ d(i,j) + d(i,j-1) + \mathrm{nw}(i-1, j-2) \\ d(i,j) + d(i-1,j) + \mathrm{nw}(i-2, j-1) \\ \gamma + \mathrm{nw}(i-1, j) \\ \gamma + \mathrm{nw}(i, j-1), \end{cases} & \text{otherwise} \end{cases} \tag{3}$$

Appropriate names for these operations are *lengthen* and *shorten*, respectively, since the first is cost-effective when the music in t is up to two times slower than the music in s, and the second is cost-effective when it is (up to two times) faster. In case there is only a slight difference in tempo between s and t, shorten and lengthen operations occur only occasionally among a majority of *match* operations. Additional operations may be defined to handle even greater tempo differences, but such differences rarely occur in practice.

3.2 Algorithmic complexity

The NW algorithm – like DTW – requires the computation of a full matrix of intermediate results which are assembled into the final result in a backtracking step. This implies time and space requirements of order $O(MN)$, where M and N are the lengths of the two sequences. Compared to NW, our extension NWTW introduces a higher per-cell cost during the construction of the dynamic programming matrix, since we add *lengthen* and *shorten* operations that have to be taken into consideration when finding the optimal operation in (3). However, as this cost is constant and not dependent on M and N, it does not change the overall complexity of the algorithm.

In practice, NWTW based on fully computing the dynamic programming matrix is feasible on current desktop computers for audio files up to about 15 minutes. For longer audio files, we use multi-step dynamic programming [15], where full dynamic programming is used for downsampled feature vectors. Subsequent alignments for higher resolution feature vectors are computed for a band of fixed width around the previously computed (coarse) alignment path.

3.3 Estimation of optimal gap penalty γ

The gap penalty γ value serves as an upper bound on the distance between pairs of elements that are considered to match: if the distance is larger than γ, the alignment will favor skipping one of the elements. This means that the choice of γ is essentially a binary classification problem, in which pairs of elements are to be classified as *match* or *non-match*, based on their distance. Let $p(x|match)$ denote the distribution of distances between matching elements, and $p(x|non\ match)$ the distribution of distances between non-matching elements, then the optimal value $\hat{\gamma}$ can be defined as the value of γ that minimizes the expected classification error:

$$\begin{aligned} \hat{\gamma} &= \operatorname*{argmin}_{\gamma} \int_0^{\gamma} p(x|non\ match)\, dx \; + \int_{\gamma}^{1} p(x|match)\, dx \\ &= \operatorname*{argmin}_{\gamma} \int_0^{\gamma} p(x|non\ match) - p(x|match)\; dx \end{aligned} \tag{4}$$

Figure 2 shows $p(x|match)$, and $p(x|non\ match)$ for imaginary data, together with the corresponding optimal value of γ. Since DTW reliably finds correct alignments between recordings in the absence of structural differences [8], the alignments it produces on such recordings provide samples from the population of matching audio features, allowing us to estimate $p(x|match)$. By sampling randomly from the distance matrix (excluding cells on the DTW path), we obtain samples from $p(x|non\ match)$. With these distributions (which can often be well approximated by *beta-distributions*), we can obtain $\hat{\gamma}$ from data using a numerical approximation of (4).

Obviously, the actual form of these two distributions will depend on the musical content, the audio features, and the distance function used for alignment. In this context, the best combination of audio features and distance function is that which maximizes the divergence between the two distributions across musical content, since it facilitates distinguishing matching audio from non-matching audio.

[1] Extensions of DTW that include weights for operations are discussed in [15]

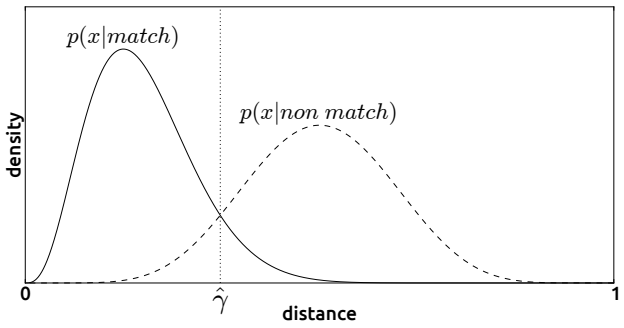

Figure 2. Schematic diagram of distance distributions between pairs of matching elements (solid), and non-matching elements (dashed); the optimal value of γ is indicated with a dotted vertical line

4. EXPERIMENTAL EVALUATION

In this section, we evaluate our proposed method NWTW by comparing it to DTW, on recordings both with and without structural differences. Before we do this, we assess different audio features by looking at how well they allow for separating matching from non-matching audio, as described in Section 3.3. Based on the results of that evaluation, we choose an audio feature, and choose the optimal γ for that feature. With this value of γ, we instantiate the NWTW algorithm, and perform a quantitative comparison of NWTW and DTW.

4.1 Choice of audio features and γ: Method and data

The purpose of this experiment is to find audio features for which the values in the distance matrix are as low as possible when they lie on the correct alignment path, and as high as possible otherwise. More specifically, we are interested in the features that maximize the divergence between the distance distributions of those two classes. The rationale for this is that with increasing divergence, the separation of the two classes by the gap penalty parameter will be more successful.

The pairs of audio recordings used for the evaluation are manually selected such that no structural differences occur. For each of the pairs, the optimal DTW path is computed to align the audio. From this alignment, the two distance distributions are computed.

4.1.1 Features

The features we evaluate are all known from the literature, including both standard features, such as MFCC and CQT coefficients, and more special purpose features such as PSD [8] and LNSO/NC [2]. Except for the LNSO/NC features (which are deliberately chosen to be used in conjunction with an adaptive distance function), all features are used in combination with the cosine distance measure, normalized to the interval $[0, 1]$.

Mel frequency cepstral coefficients (MFCC). Mel Frequency Cepstral Coefficients [12] are an STFT based audio representation well known in speech and music processing. MFCC's are a compressed version of the spectral envelope of a short-term section of an audio signal, and they are especially useful for capturing the timbre and the formant structure of speech/music signals. It is common to ignore the first MFCC coefficient, and to take only the first *n* coefficients. Here we evaluate both $n = 13$ and $n = 50$. In addition to that, we use two FFT sizes: 46ms, and 372ms.

Constant Q transform (CQT). The Constant Q transform [4] is a time-frequency transform, but unlike the STFT – which implements a constant bin width and therefore yields a non-constant bin center frequency to bin width ratio (the Q value) — it forces the Q value to stay constant and modifies the bin widths accordingly. The CQT is suitable for representing musical audio signals since its structure resembles the diatonic scale: all octaves are an equal number of bins apart.

Positive spectral difference (PSD). This feature was proposed in [8], and is designed to capture onset information for performance-to-performance alignment. It is based on a Short Time Fourier Transform with the frequency bins mapped to a musically meaningful scale. The PSD feature is computed as the half-wave rectification of the energy difference per frequency bin from one audio frame to the next.

Locally adaptive features/distance (LNSO/NC). For the purpose of audio alignment, Arzt et al. [2] propose to compute a weighted sum of distances of two features. The first, Locally Normalized Semitone Onset (LNSO) is an adaptation of PSD, and responds strongly to onsets. In absence of onsets, the distance is dominated by a second feature, Normalized Chroma (NC), capturing harmonic information.

4.1.2 Data

In order to ensure generality of the results beyond a single type of acoustic signals, we use three different classes of recordings (all obtained from commercial cd's):

Symphony orchestra: 7 Pieces from 5 different Beethoven symphonies, by 10 different conductors, amounting to 148 comparisons between 59 recordings (4.4 hours of music).

Solo guitar: The complete guitar works of Villa-Lobos (23 pieces), by 5 performers, amounting to 103 comparisons between 83 recordings (4.6h).

Solo piano: 14 Movements from 6 different piano sonatas by Mozart, by 7 performers, amounting to 720 comparisons between 308 recordings (6.4h).

4.2 Comparison of NWTW and DTW: Method and data

In this part of the experimentation we evaluate alignment accuracies quantitatively using manual annotations of the beat in a set of recordings of Mozart piano sonatas. We use the evaluation procedure used in [8], in which for each annotated beat the alignment error is the Manhattan distance

(in frames) to the closest point on the computed alignment. We compare DTW and NWTW, both on pairs of recording with and without structural differences. When structural differences occur, the alignment error for a given beat is the minimum error among the instances of that beat in the repetitions. Informally, the error criterion does not penalize the alignment for passing through one repetition of a section rather than through another.

4.2.1 Data

The recordings we use for this are from the same solo piano data set as the data described in Section 4.1.2, for which manual beat annotations are available (details on the annotation process can be found in [21]). We take pairs or recordings from this set such that each pair is a recording of the same piece by a different performer. Of these pairs, 74 are without structural differences. This set involves 6 performers, playing 41 movements from 20 sonatas. In addition, we take pairs of recordings with structural differences. This set consists of 133 pairs, and involves 8 performers, and 56 movements from 26 sonatas.

4.3 Results and discussion

Figure 3 shows the distance distributions between matching and non-matching audio, computed on the various data sets, using the various features. In general, the distributions vary more strongly across features than across the different types of audio. The pitch-oriented features with high frequency resolution (most notably MFCC50 / FFT.372s, and CQT) tend to be those with highest Jensen-Shannon divergence (JSD, shown in the plots). That said, the solo guitar data set in combination with the MFCC features shows a substantial reduction in JSD. This could be a consequence of the sensitivity of the MFCC features to the guitar tuning. Note also that this leads to a rightward shift of the optimal γ value, shown as dotted vertical lines in the plots. Although this discrepancy between optimal γ values across data sets is principally undesirable, the MFCC50 / FFT.372s feature still yields the highest JSD when over the joint data set (bottom row in Figure 3). For this reason, we use the MFCC50 / FFT.372s feature, and the corresponding optimal parameter value $\hat{\gamma} = 0.346$, for the subsequent quantitative evaluation of the DTW and NWTW methods.

The success of MFCC features for alignment is at odds with the findings of [11], even if they only evaluate the features indirectly through a retrieval task. An explanation for this may be the number of MFCC's selected: we obtain best results with 50 MFCC's, which is substantially more than the first 13 MFCC's typically used.

Table 1 shows the alignment accuracies for DTW and NWTW. When no structural differences occur between recordings, no jumps are required. In that case, the accuracy of NWTW alignment is very similar to that of DTW. When differences do occur, DTW tends to align parts of non-matching audio segments (as illustrated schematically in Figure 1), leading to higher alignment errors. The straight jumps that NWTW tends to make, ensure that the alignment path is always close to a matching position in either one or the other of a section that is repeated in only one of the recordings.

Error $\leq$ (ms)	0	20	40	60	80	100	200	500	1000
	alignment of performances *without* structural differences								
DTW	**47.1**	72.6	84.5	90.6	93.6	95.4	98.3	**99.6**	99.9
NWTW	46.3	**73.3**	**85.5**	**91.5**	**94.5**	**96.1**	**98.6**	**99.6**	**100.0**
	alignment of performances *with* structural differences								
DTW	37.0	60.6	73.1	80.2	83.6	85.6	89.2	91.5	92.9
NWTW	**38.1**	**66.0**	**79.5**	**86.7**	**90.1**	**91.7**	**94.5**	**96.4**	**97.3**

Table 1. Alignment accuracies for DTW and NWTW, for pairs of recordings without (top) and with (bottom) structural differences; The values represent the percentages of annotated beats with a Manhattan distance less or equal to the corresponding times in the top row; For example, using NWTW in structurally different audio, 96,4% of the beats are aligned no more than 500ms apart (91.5% for DTW)

5. CONCLUSIONS

In this paper we propose Needleman-Wunsch time warping (NWTW), a pure dynamic programming method to align music recordings that contain structural differences, and propose a way to estimate the optimal value for the gap penalty parameter γ. Experiments show that audio features with high frequency resolution allow for the most effective use of the gap penalty parameter. Moreover, a single value for γ is (close to) optimal for different types of music, including both solo instruments, and symphonic orchestra.

The advantage of our method over classical dynamic time warping is that it handles structural differences better, and the advantage over the original NW algorithm is that it handles tempo discrepancies between different recordings.

A limitation of the method in its current form is that it does not prefer jumps at the beginnings or ends of structural units over jumps at intermediate positions. Although this is not problematic for application scenarios that only require a matching position in one recording for each position in the other, jumps at intermediate positions in a structural unit are counter-intuitive from a musical point of view. We are currently investigating a further extension of the method to resolve this.

6. ACKNOWLEDGMENTS

This research is supported by the European Union Seventh Framework Programme FP7 / 2007-2013, through the PHENICX project (grant agreement no. 601166).

7. REFERENCES

[1] A. Arzt, G. Widmer, and S. Dixon. Automatic page turning for musicians via real-time machine listening. In *ECAI*, pages 241–245, 2008.

[2] A. Arzt, G. Widmer, and S. Dixon. Adaptive distance normalization for real-time music tracking. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 2689–2693. IEEE, 2012.

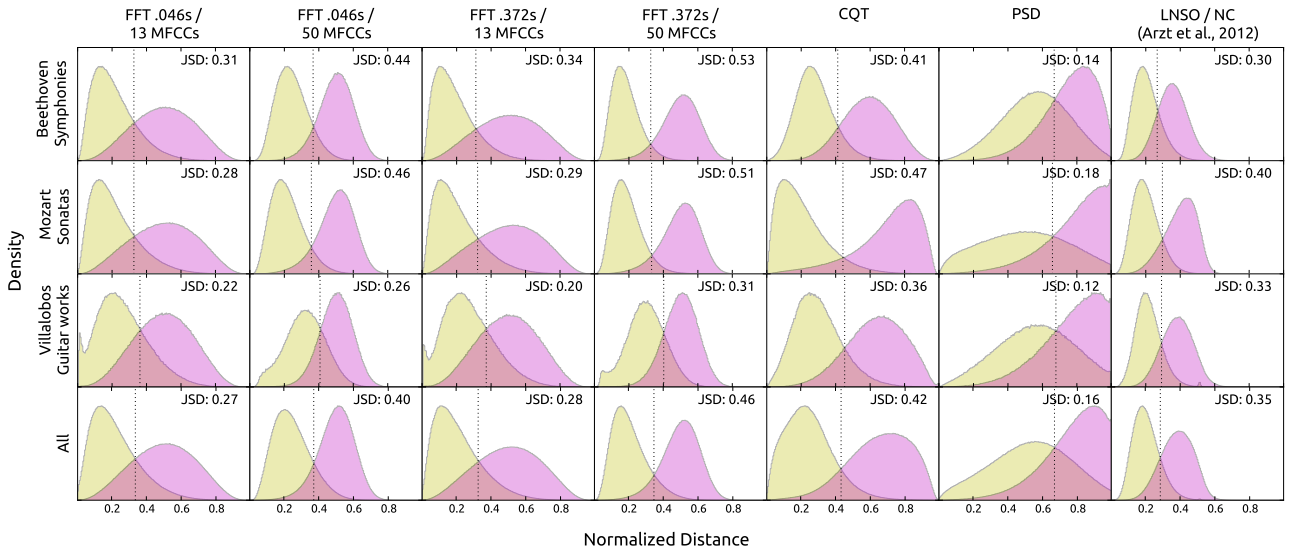

Figure 3. Distance distributions per feature for different types of music; In each subplot, the yellow distribution (left) is the distribution of distances between features of matching audio, the magenta distribution (right) is the distribution of distances between features of non-matching audio; The dashed vertical lines indicates the optimal value for γ; The value indicated with JSD is the Jensen-Shannon divergence between the two distributions

[3] J. P. Bello. Audio-based cover song retrieval using approximate chord sequences: Testing shifts, gaps, swaps and beats. In *Proc. of the 8th International Conference on Music Information Retrieval*, pages 239–244, Vienna, Austria, September 23-27 2007.

[4] J. C. Brown and M. S. Puckette. An efficient algorithm for the calculation of a constant q transform. *The Journal of the Acoustical Society of America*, 92:2698, 1992.

[5] R. Dannenberg. An on-line algorithm for real-time accompaniment. In *Proceedings of the 1984 International Computer Music Conference*. International Computer Music Association, 1984.

[6] R. B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. *Commun. ACM*, 49(8):38–43, 2006.

[7] C. Dittmar, K. Hildebrand, D. Gaertner, M. Winges, F. Muller, and P. Aichroth. Audio forensics meets music information retrieval; a toolbox for inspection of music plagiarism. In *Proc. of the 20th European Signal Processing Conference (EUSIPCO)*, pages 1249–1253, 2012.

[8] S. Dixon and G. Widmer. Match: A music alignment tool chest. In *Proceedings of the 6th International Conference on Music Information Retrieval*, London, UK, September 11-15 2005.

[9] C. Fremerey, M. Müller, and M. Clausen. Handling repeats and jumps in score-performance synchronization. In *Proc. of the 11th International Society for Music Information Retrieval Conference*, pages 243–248, Utrecht, The Netherlands, August 9-13 2010.

[10] M. Grachten, J. L. Arcos, and R. López de Mántaras. A case based approach to expressivity-aware tempo transformation. *Machine Learning*, 65(2–3):411–437, 2006.

[11] N. Hu, R. B. Dannenberg, and G. Tzanetakis. Polyphonic audio matching and alignment for music retrieval. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 185–188. IEEE, 2003.

[12] B. Logan. Mel frequency cepstral coefficients for music modeling. In *Proceedings of the 1st International Conference on Music Information Retrieval*, Plymouth, Massachusetts, 2000.

[13] M. Mongeau and D. Sankoff. Comparison of musical sequences. *Computers and the Humanities*, 24:161–175, 1990.

[14] N. Montecchio and A. Cont. Accelerating the mixing phase in studio recording productions by automatic audio alignment. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, pages 627–632, Miami (Florida), USA, 2011.

[15] M. Müller. *Information Retrieval for Music and Motion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[16] M. Müller and D. Appelt. Path-constrained partial music synchronization. In *Proceedings of the 34th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 65–68, Las Vegas, Nevada, USA, Apr. 2008.

[17] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443 – 453, 1970.

[18] B. Pardo and W. Birmingham. Modeling form for on-line following of musical performances. In *Proceedings of the 20th national conference on Artificial intelligence - Volume 2*, AAAI'05, pages 1018–1023. AAAI Press, 2005.

[19] J. Serrà, E. Gómez, P. Herrera, and X. Serra. Chroma binary similarity and local alignment applied to cover song identification. *IEEE Transactions on Audio, Speech and Language Processing*, 16:1138–1151, 2008.

[20] R. J. Turetsky and D. P. W. Ellis. Ground-truth transcriptions of real music from force-aligned midi syntheses. In *Proceedings of the Fourth International Conference on Music Information Retrieval*, Baltimore (Maryland), USA, 2003.

[21] G. Widmer, S. Dixon, W. Goebl, E. Pampalk, and A. Tobudic. In search of the Horowitz factor. *AI Magazine*, 24(3):111–130, 2003.