EXPLOITING INSTRUMENT-WISE PLAYING/NON-PLAYING LABELS FOR SCORE SYNCHRONIZATION OF SYMPHONIC MUSIC

Alessio Bazzica
Delft University of Technology
a.bazzica@tudelft.nl

Cynthia C. S. Liem
Delft University of Technology
c.c.s.liem@tudelft.nl

Alan Hanjalic
Delft University of Technology
a.hanjalic@tudelft.nl

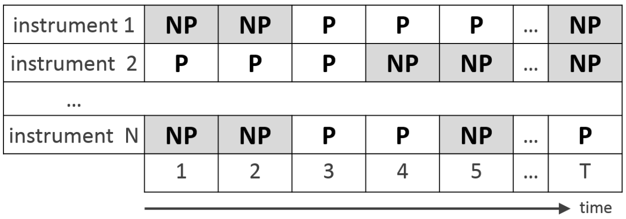
ABSTRACT

Synchronization of a score to an audio-visual music performance recording is usually done by solving an audio-to-MIDI alignment problem. In this paper, we focus on the possibility to represent both the score and the performance using information about which instrument is active at a given time stamp. More specifically, we investigate to what extent instrument-wise "playing" (P) and "non-playing" (NP) labels are informative in the synchronization process and what role the visual channel can have for the extraction of P/NP labels. After introducing the P/NP-based representation of the music piece, both at the score and performance level, we define an efficient way of computing the distance between the two representations, which serves as input for the synchronization step based on dynamic time warping. In parallel with assessing the effectiveness of the proposed representation, we also study its robustness when missing and/or erroneous labels occur. Our experimental results show that P/NP-based music piece representation is informative for performance-to-score synchronization and may benefit the existing audio-only approaches.

1. INTRODUCTION AND RELATED WORK

Synchronizing an audio recording to a symbolic representation of the performed musical score is beneficial to many tasks and applications in the domains of music analysis, indexing and retrieval, like audio source separation [4, 9], automatic accompaniment [2], sheet music-audio identification [6] and music transcription [13]. As stated in [7], "sheet music and audio recordings represent and describe music on different semantic levels" thus making them complementary for the functionalities they serve.

The need for effective and efficient solutions for audio-score synchronization is especially present for genres like symphonic classical music, for which the task remains challenging due to the typically long duration of the pieces and a high number of instruments involved [1]. The existing solutions usually turn this synchronization problem into an audio-to-audio alignment one [11], where the score is rendered in audio form using its MIDI representation.



instrument 1	NP	NP	P	P	P	...	NP
instrument 2	P	P	P	NP	NP	...	NP
...							
instrument N	NP	NP	P	P	NP	...	P
	1	2	3	4	5	...	T

Figure 1: An illustration of the representation of a symphonic music piece using the matrix of playing/non-playing labels.

In this paper, we investigate whether sequences of playing (P) and non-playing (NP) labels, extracted per instrument continuously over time, can alternatively be used to synchronize a recording of a music performance to a MIDI file. At a given time stamp, the P (NP) label is assigned to an instrument if it is (not) being played. If such labels are available, a representation of the music piece as illustrated in Figure 1 can be obtained: a matrix encoding the P/NP "state" for different instruments occurring in the piece at subsequent time stamps. Investigating the potential of this representation for synchronization purposes, we will address the following research questions:

- RQ1: How robust is P/NP-based synchronization in case of erroneous or missing labels?
- RQ2: How does synchronizing P/NP labels behave at different time resolutions?

We are particularly interested in this representation, as P/NP information for orchestra musicians will also be present in the signal information of a recording. While such information will be hard to obtain from the audio channel, it can be obtained from the visual channel. Thus, in case an audio-visual performance is available, using P/NP information opens up possibilities for video-to-score synchronization as a means to solve a score-to-performance synchronization problem.

The rest of the paper is structured as follows. In Section 2, we formulate the performance-to-score synchronization problem in terms of features based on P/NP labels. Then, we explain how the P/NP matrix is constructed to represent the score (Section 3) and we elaborate on the possibilities for extracting the P/NP matrix to represent the analyzed performance (Section 4). In Section 5 we propose an efficient method for solving the synchronization problem. The experimental setup is described in Section 6 and in Section 7 we report the results of our

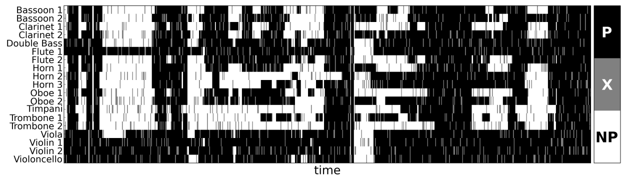


Figure 2: Example of a M_{PNP} matrix with missing labels.

experimental assessment of the proposed synchronization methodology and provide answers to our research questions. The discussion in Section 8 concludes the paper.

2. PROBLEM DEFINITION

Given an audio-visual recording of a performance and a symbolic representation of the performed scores, we address the problem of synchronizing these two resources by exploiting information about the instruments which are active over time.

Let $L = \{-1, 0, 1\}$ be a set encoding the three labels non-playing (NP), missing (X) and playing (P). Let $M_{\mathrm{PNP}} = \{m_{ij}\}$ be a matrix of $N_{\mathrm{I}} \times N_{\mathrm{T}}$ elements where N_{I} is the number of instruments and N_{T} is the number of time points at which the P/NP state is observed. The value of $m_{ij} \in L$ represents the state of the i-th instrument observed at the j-th time point ($1 \leq i \leq N_{\mathrm{I}}$ and $1 \leq j \leq N_{\mathrm{T}}$). An example of M_{PNP} is given in Figure 2.

We now assume that the matrices $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and $M^{\mathrm{S}}_{\mathrm{PNP}}$ are given and represent the P/NP information respectively extracted by the audio-visual recording and the sheet music. The two matrices have the same number of rows and each row is associated to each instrumental part. The number of columns, i.e. observations over time, is in general different. The synchronization problem can be then formulated as the problem of finding a time map $f_{sync} : \{1 \dots N^{\mathrm{AV}}_{\mathrm{T}}\} \rightarrow \{1 \dots N^{\mathrm{S}}_{\mathrm{T}}\}$ linking the observation time points of the two resources.

3. SCORE P/NP REPRESENTATION

For a given piece, we generate one P/NP matrix $M^{\mathrm{S}}_{\mathrm{PNP}}$ for the score relying on the corresponding MIDI file as the information source.

We start generating the representation of the score by parsing the data of each available track in the given MIDI file. Typically, one track per instrument is added and is used as a symbolic representation of the instrumental part's score. More precisely, when there is more than one track for the same instrument (e.g. Violin 1, Violin 2 - which are two different instrumental parts), we keep both tracks as separate. In the second step, we use a sliding window that moves along the MIDI file and derive a P/NP label per track and window position. A track receives a P label if there is at least one note played within the window. We work with the window in order to comply with the fact that a played note has a beginning and end and therefore lasts for an interval of time. In this sense, a played note is registered when there is an overlap between the sliding window and the play interval of that note.

The length of the window is selected such that short rests within a musical phrase do not lead to misleading P-NP-P switches. We namely consider a musician in the "play" mode if she is within the "active" sequence of the piece with respect to her instrumental part's score, independently whether at some time stamps no notes are played. In our experiments, we use a window length of 4 seconds which has been determined by empirical evaluation, and a step-size of 1 second. This process generates one label per track every second.

In order to generalize the parameter setting for window length and offset, we also related them to the internal MIDI file time unit. For this purpose, we set a reference value for the tempo. Once the value is assigned, the sliding window parameters are converted from seconds to beats. The easiest choice is adopting a fixed value of tempo for every performance. Alternatively, when an audio-visual recording is available, the reference tempo can be estimated as the number of beats in the MIDI file divided by the length of the recording expressed in minutes. A detailed investigation of different choices of the tempo is reported in [6].

4. PERFORMANCE P/NP REPRESENTATION

While an automated method could be thought of to extract the P/NP matrix $M^{\mathrm{AV}}_{\mathrm{PNP}}$ from a given audio-visual recording, developing such a method is beyond the scope of this paper. Instead, our core focus is assessing the potential of such a matrix for synchronization purposes, taking into account the fact that labels obtained from real-world data can be noisy or even missing. We therefore deploy two strategies which mimic the automated extraction of the $M^{\mathrm{AV}}_{\mathrm{PNP}}$ matrices. We generate them: (i) artificially, by producing (noisy) variations of the P/NP matrices derived from MIDI files (Section 4.1), and (ii) more realistically, by deriving the labels directly from the visual channel of a recording in a semi-automatic way (Section 4.2).

4.1 Generating synthetic P/NP matrices

The first strategy produces synthetic P/NP matrices by analyzing MIDI files as follows. Similarly to the process of generating a P/NP matrix for the score, we apply a sliding window to the MIDI file and extract labels per instrumental track at each window position. This time, however, time is randomly warped, i.e. the sliding window moves over time with non-constant velocity. More specifically, we generate random time-warping functions by randomly changing slope every 3 minutes and by adding a certain amount of random noise in order to avoid perfect piecewise linear functions. In a real audio-visual recording analysis pipeline, we expect that erroneous and missing P/NP labels will occur. Missing labels may occur if musicians cannot be detected, e.g. because of occlusion or leaving the camera's angle of view in case of camera movement. In order to simulate such sources of noise, we modify the generated P/NP tracks by randomly flipping and/or deleting predetermined amounts of labels at random positions of the P/NP matrices.

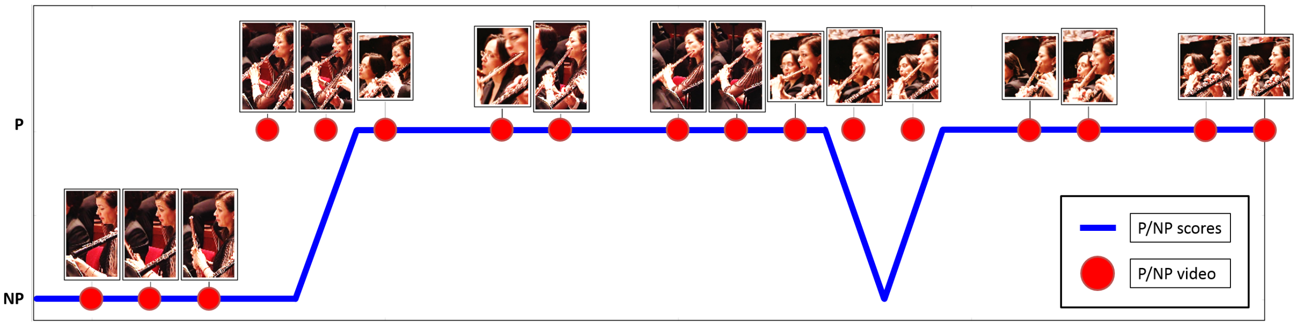


Figure 3: Example of P/NP labels extracted from the visual channel (red dots) and compared to labels extracted by the score (blue line).

4.2 Obtaining P/NP matrices from a video recording

The second strategy more closely mimics the actual video analysis process and involves a simple, but effective method that we introduce for this purpose. In this method, we build on the fact that video recordings of a symphonic music piece are typically characterized by regular close-up shots of different musicians. From the key frames representing these shots, as illustrated by the examples in Figure 4, it can be inferred whether they are using their instrument at that time stamp or not, for instance by investigating their body pose [14].

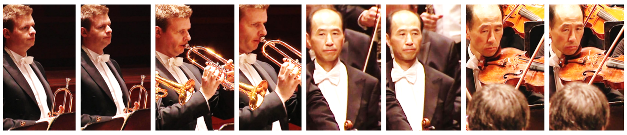


Figure 4: Examples of body poses indicating playing/non-playing state of a musician.

In the first step, a key frame is extracted every second in order to produce one label per second, as in the case of the scores. Faces are detected via off-the-shelf face detectors and upper-body images are extracted by extending the bounding box's areas of face detector outputs. We cluster the obtained images using low-level global features encoding color, shape and texture information. Clustering is done using k-means with the goal to isolate images of different musicians. In order to obtain high precision, we choose a large value for k. As a result, we obtain clusters mostly containing images of the same musician, but also multiple clusters for the same musician. Noisy clusters (those not dominated by a single musician) are discarded, while the remaining are labeled by linking them to the correspondent track of the MIDI file (according to the musician's instrument and position in the orchestra, i.e. the instrumental part). In order to label the upper-body images as P/NP, we generate sub-clusters using the same features as those extracted in the previous (clustering) step. Using once again k-means, but now with k equal to 3 (one cluster meant for P labels, one for NP and one extra label for possible outliers), we build sub-clusters which we label as either playing (P), non-playing (NP) or undefined (X). Once the labels for every musician are obtained, they are aggregated by instrumental part (e.g. the labels from all the Violin 2 players are combined by majority voting). An example of a P/NP subsequence extracted by visual analysis is given in Figure 3.

5. SYNCHRONIZATION METHODOLOGY

In this section, we describe the synchronization strategy used in our experiments. The general idea is to compare configurations of P/NP labels for every pair of performance-score time points and produce a distance matrix. The latter can then serve as input into a synchronization algorithm, for which we adopt the well-known dynamic time warping (DTW) principle. This implies we will not be able to handle undefined amounts of repeats of parts of the score. However, this is a general issue for DTW also holding for existing synchronization approaches, which we consider out of the scope of this paper.

In order to find the time map between performance and score, we need to solve the problem of finding time links between the given $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and $M^{\mathrm{S}}_{\mathrm{PNP}}$ matrices. To this end, we use a state-of-the-art DTW algorithm [12].

5.1 Computing the distance matrix

Ten Holt et. al. [12] compute the distance matrix through the following steps: (i) both dimensions of the matrices are normalized to have zero mean and unit variance, (ii) optionally a Gaussian filter is applied, and (iii) pairs of vectors are compared using the city block distance. In our case, we take advantage of the fact that our matrices contain values belonging to the finite set of 3 different integers, namely the set L introduced in Section 2. This enables us to propose an alternative, just as effective, but more efficient method to compute the distance matrix.

Let $\boldsymbol{m}^{\mathrm{AV}}_{j}$ and $\boldsymbol{m}^{\mathrm{S}}_{k}$ be two column vectors respectively belonging to M^{AV}_{PNP} and M^{S}_{PNP}. To measure how (dis-)similar those two vectors are, we define a *correlation score* s_{jk} as follows:

$$s_{jk} = \mathrm{corr}(\boldsymbol{m}^{\mathrm{AV}}_{j}, \boldsymbol{m}^{\mathrm{S}}_{k}) = \sum_{i=1}^{N_{\mathrm{I}}} m^{\mathrm{AV}}_{ij} \cdot m^{\mathrm{S}}_{ik}$$

From such definition, it follows that a pair of observed matching labels add a positive unitary contribution. If the observed labels do not match, the added contribution is unitary and negative. Finally, if one or both labels are not observed (i.e. at least one of them is X), the contribution is 0. Hence, it also holds $-N_{\mathrm{I}} \leq s_{jk} \leq +N_{\mathrm{I}}$. The maximum is reached only if the two vectors are equal. Correlation scores can be efficiently computed as dot-product of the given P/NP matrices, namely as $(M^{\mathrm{AV}}_{\mathrm{PNP}})^{\top} M^{\mathrm{S}}_{\mathrm{PNP}}$.

The distance matrix $D = \{d_{jk}\}$, whose values are zero when the compared vectors are equal, can now be computed as $d_{jk} = N_{\mathrm{I}} - s_{jk}$. As a result, D will have $N^{\mathrm{AV}}_{\mathrm{T}}$

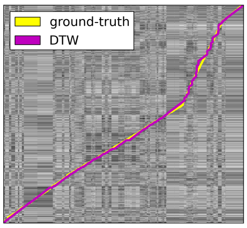
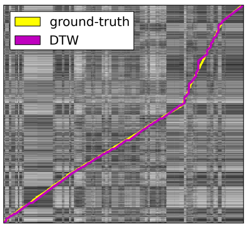
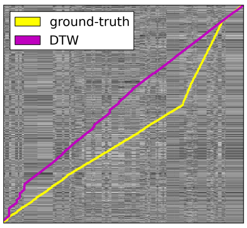
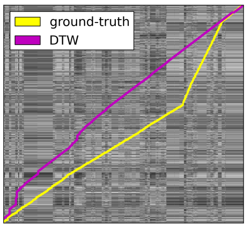
noisy M_{PNP}		very noisy M_{PNP}	
			
Ten Holt et. al.	our method	Ten Holt et. al.	our method

Table 1: Comparing our distance matrix definition to Ten Holt et. al. [12]. By visual inspection, we observe comparable alignment performances. However, the computation of our distance matrix is much faster.

rows and $N_{\mathrm{T}}^{\mathrm{S}}$ columns. When the correlation is the highest, namely equal to N_{I}, the distance will be zero.

Our approach has two properties that make the computation of D fast: D is computed via the dot product and it contains integer values only (as opposed to standard methods based on real-valued distances). As shown in Table 1, both the distance matrix proposed in [12] and using our definition produce comparable results. Since our method allows significantly faster computation (up to 40 times faster), we adopt it in our experiments.

5.2 Dynamic Time Warping

Once the distance matrix D is computed, the time map between $M_{\mathrm{PNP}}^{\mathrm{AV}}$ and $M_{\mathrm{PNP}}^{\mathrm{S}}$ is determined by solving the optimization problem: $P^{\star} = \arg\min_{P} \mathrm{cost}(D, P)$ where $P = \{(p_{\ell} \rightsquigarrow p_{\ell+1})\}$ is a path through the items of D having a cost defined by the function $\mathrm{cost}(D, P)$. More specifically, $p_{\ell} = (i_{\ell}^{\mathrm{AV}}, i_{\ell}^{\mathrm{S}})$ is a coordinate of an element in D. The cost function is defined as $\mathrm{cost}(D, P) = \sum_{\ell=1}^{|P|} d_{i_{\ell}^{\mathrm{AV}}, i_{\ell}^{\mathrm{S}}}$ The aforementioned problem is efficiently solved via dynamic programing using the well-known dynamic time warping (DTW) algorithm. Examples of $P^{\star}$ paths computed via DTW are shown in the figures of Table 1.

Once $P^{\star}$ is found, the time map f_{sync} is computed through the linear interpolation of the correspondences in $P^{\star}$, i.e. the set of coordinates $\{p_{\ell}^{\star} = (i_{\ell}^{\mathrm{AV}}, i_{\ell}^{\mathrm{S}})\}$. This map allows to define correspondences between the two matrices, as shown in the example of Figure 5.

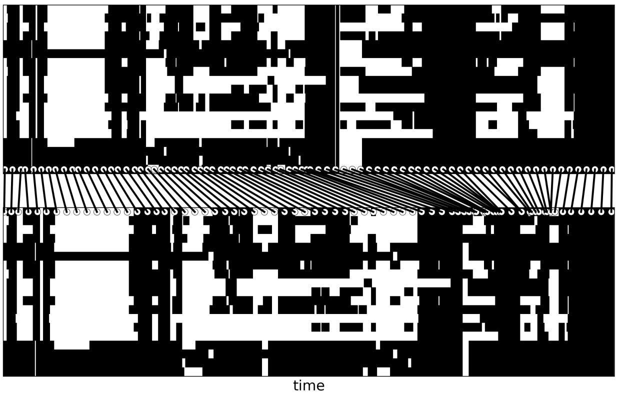


Figure 5: Example of produced alignment between two fully-observed M_{PNP} matrices.

6. EXPERIMENTAL SETUP

In this section, we describe our experimental setup including details about the dataset. In order to ensure the reproducibility of the experiments, we release the code and share the URLs of the analyzed freely available MIDI files [1].

We evaluate the performances of our method on a set of 29 symphonic pieces composed by Beethoven, Mahler, Mozart and Schubert. The dataset consists of 114 MIDI files. Each MIDI file contains a number of tracks corresponding to different parts performed in a symphonic piece. For instance, first and second violins are typically encoded in two different parts (e.g. "Violin 1" and "Violin 2"). In such a case, we keep both tracks separate since musicians in the visual channel can be labeled according to the score which they perform (and not just by their instrument). We ensured that the MIDI files contain tracks which are mutually synchronized (i.e. MIDI files of type 1). The number of instrumental parts, or MIDI tracks, ranges between 7 and 31 and is distributed as shown in Figure 7.

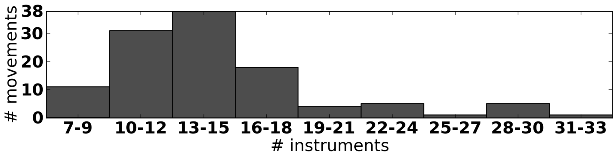
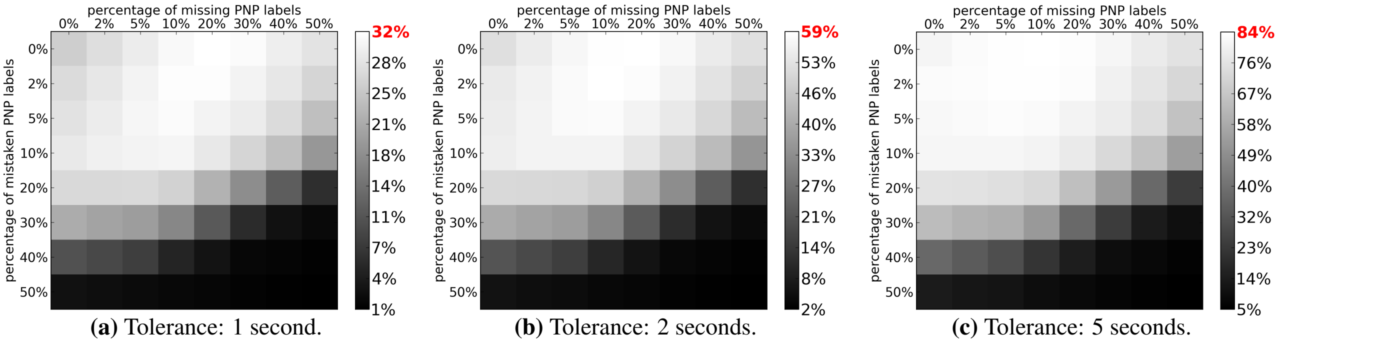


Figure 7: Distribution of the number of instrumental parts across performances in the data set.

For each MIDI file, we perform the following steps. First, we generate one $M_{\mathrm{PNP}}^{\mathrm{S}}$ matrix using a fixed reference tempo of 100 BPM. The reason why we use the same value for every piece is that we evaluate our method on artificial warping paths, hence we do not need to adapt the sliding window parameters to any actual performance. Then we generate one random time-warping function which has two functions: (i) it is used as ground-truth when evaluating the alignment performance, and (ii) it is used to make one time-warped P/NP matrix $M_{\mathrm{PNP}}^{\mathrm{AV}}$. The latter is used as template to build noisy copies of $M_{\mathrm{PNP}}^{\mathrm{AV}}$ and evaluate the robustness of our method. Each template P/NP matrix is used to generate a set of noisy P/NP matrices which are affected by different pre-determined amounts of noise. We consider two sources of noise: mistaken and missing labels. For both sources, we generate

[1] http://homepage.tudelft.nl/f8j6a/ISMIR2014baz.zip



(a) Tolerance: 1 second. **(b)** Tolerance: 2 seconds. **(c)** Tolerance: 5 seconds.

Figure 6: Average matching rates as a function of the percentage of mistaken and/or missing labels at different tolerance thresholds.

the following percentages of noisy labels: 0% (noiseless), 2%, 5%, 10%, 20%, 30%, 40% and 50%. For every pair of noise percentages, e.g. 5% mistaken + 10% missing, we create 5 different noisy versions of the original P/NP matrix [2]. Therefore, for each MIDI file, the final set of matrices has the size $1 + (8 \times 8 - 1) \times 5 = 316$. Overall, we evaluate the temporal alignment of $316 \times 114 = 36024$ P/NP sequences.

For each pair of M_{PNP} matrices to be aligned, we compute the matching rate by sampling f_{sync} and measuring the distance from the true alignment. A match occurs when the distance between linked time points is below a threshold. In our experiments, we evaluate the matching rate using three different threshold values: 1, 2 and 5 seconds.

Finally, we apply the video-based P/NP label extraction strategy described in Section 4.2 to a multiple camera video recording of the 4th movement of Symphony no. 3 op. 55 of Beethoven performed by the Royal Concertgebouw Orchestra (The Netherlands). For this performance, in which 54 musicians play 19 instrumental parts, we use the MIDI file and the correspondent performance-score temporal alignment file which are shared by the authors of [8]. The latter is used as ground truth when evaluating the synchronization performance.

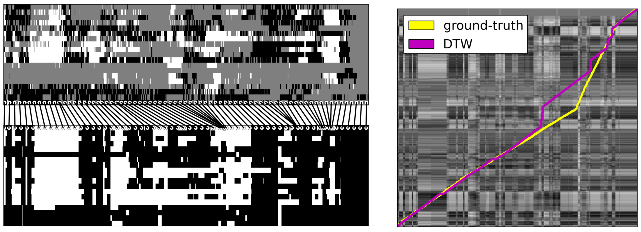
7. RESULTS

In this section, we present the obtained results and provide answers to the research questions posed in Section 1. We start by presenting in Figure 6 the computed matching rates in 3 distinct matrices, one for each threshold value. Given a threshold, the overall matching rates are reported in an 8×8 matrix since we separately compute the average matching rate for each pair of mistaken-missing noise rates. Overall, we see two expected effects: (i) the average matching rate decreases for larger amounts of noise, and (ii) the performance increases with the increasing threshold. What was not expected is the fact that the best performance is not obtained in the noiseless case. For instance, when the threshold is 5 seconds, we obtained an average matching rate of 81.7% in the noiseless case and 85.0% in the case of 0% mistaken and 10% missing labels. One possible explanation is that 10% missing labels could give more “freedom” to the DTW algorithm than the noiseless case. Such freedom may lead to a better global optimization. In order to fully understand the reported outcome, however, further investigation is needed, which we leave for future work.

As for our first research question, we conclude that the alignment through P/NP sequences is more robust to missing labels than to mistaken ones. We show this by the fact that the performance for 0% mistaken and 50% missing labels are higher than in the opposite case, namely for 50% mistaken and 0% missing labels. In general the best performance is obtained for up to 10% mistaken and 30% missing labels.

In the second research question we address the behavior at different time resolutions. Since labels are sampled every second, it is clear why acceptable matching rates are only obtained at coarse resolution (namely for a threshold of 5 seconds).

Finally, we comment on the results obtained when synchronizing through the P/NP labels assigned via visual analysis. The P/NP matrix, shown in Figure 8a, is affected by noise as follows: there are 53.95% missing and 8.65% mistaken labels.



(a) $M_{\mathrm{PNP}}^{\mathrm{AV}}$ and $M_{\mathrm{PNP}}^{\mathrm{S}}$ **(b)** DTW

Figure 8: Real data example: P/NP labels by analysis of video

We immediately notice the large amount of missing labels. This is mainly caused by the inability to infer a P/NP label at those time points when all the musicians of a certain instrumental part are not recorded. Additionally, some of the image clusters generated as described in Section 4.2 are not pure and hence labeled as X.

The obtained synchronization performance at 1, 2 and 5 seconds of tolerance are respectively 18.74%, 34.49% and 60.70%. This is in line with the results obtained with synthetic data whose performance at 10% of mistaken labels and 50% of missing for the three different tolerances are 24.3%, 44.2% and 65.9%. Carrying out the second exper-

[2] We do not add extra copies for the pair (0%,0%), i.e. the template matrix.

iment was also useful to get insight about the distribution of missing labels. By inspecting Figure 8a, we notice that such a type of noise is not randomly distributed. Some musicians are sparsely observed over time hence leading to missing labels patterns which differ from uniform distributed random noise.

8. DISCUSSION

In this paper, we presented a novel method to synchronize score information of a symphonic piece to a performance of this piece. In doing this, we used a simple feature (the act of playing or not) which trivially is encoded in the score, and feasibly can be obtained from the visual channel of an audio-visual recording of the performance. Unique about our approach is that both for the score and the performance, we start from measuring individual musician contributions, and only then aggregate up to the full ensemble level to perform synchronization. This makes a case for using the visual channel of an audio-visual recording. In the audio channel, which so far has predominantly been considered for score-to-performance synchronization, even if separate microphones are used per instrument, different instruments will never be fully isolated from each other in a realistic playing setting. Furthermore, audio source separation for polyphonic orchestral music is far from being solved. However, in the visual channel, different players are separated by default, up to the point that a first clarinet player can be distinguished from a second clarinet player, and individual contributions can be measured for both.

Our method still works at a rough time resolution, and lacks the temporal sub-second precision of typical audio-score synchronization methods. However, it is computationally inexpensive, and thus can quickly provide a rough synchronization, in which individual instrumental part contributions are automatically marked over time. Consequently, interesting follow-up approaches could be devised, in which cross- or multi- modal approaches might lead to stronger solutions, as already argued in [3, 10].

For the problem of score synchronization, a logical next step is to combine our analysis with typical audio-score synchronization approaches, or approaches generally relying on multiple synchronization methods, such as [5], to investigate whether a combination of methods improves the precision and efficiency of the synchronization procedure. Our added visual information layer can further be useful for e.g. devising structural performance characteristics, e.g. the occurrence of repeats. Our general synchronization results will also be useful for source separation procedures, since the obtained P/NP annotations indicate active sound-producing sources over time. Furthermore, results of our method can serve applications focusing on studying and learning about musical performances. We can easily output an activity map or multidimensional time-scrolling bar, visualizing which orchestra parts are active over time in a performance. Information about expected musical activity across sections can also help directing the focus of an audience member towards dedicated players or the full ensemble.

Finally, it will be interesting to investigate points where P/NP information in the visual and score channel clearly disagree. For example, in Figure 3, some time after the flutist starts playing, there is a moment where the score indicates a non-playing interval, while the flutist keeps a playing pose. We hypothesize that this indicates that, while a (long) rest is notated, the musical discourse actually still continues. While this also will need further investigation, this opens up new possibilities for research in performance analysis and musical phrasing, broadening the potential impact of this work even further.

Acknowledgements The research leading to these results has received funding from the European Union Seventh Framework Programme FP7 / 2007–2013 through the PHENICX project under Grant Agreement no. 601166.

9. REFERENCES

[1] A. D'Aguanno and G. Vercellesi. Automatic Music Synchronization Using Partial Score Representation Based on IEEE 1599. *Journal of Multimedia*, 4(1), 2009.

[2] R.B. Dannenberg and C. Raphael. Music Score Alignment and Computer Accompaniment. *Communications of the ACM*, 49(8):38–43, 2006.

[3] S. Essid and G. Richard. Fusion of Multimodal Information in Music Content Analysis. *Multimodal Music Processing*, 3:37–52, 2012.

[4] S. Ewert and M. Muller. Using Score-informed Constraints for NMF-based Source Separation. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 129–132. IEEE, 2012.

[5] S. Ewert, M. Müller, and R.B. Dannenberg. Towards Reliable Partial Music Alignments Using Multiple Synchronization Strategies. In *Adaptive Multimedia Retrieval. Understanding Media and Adapting to the User*, pages 35–48. Springer, 2011.

[6] C. Fremerey, M. Clausen, S. Ewert, and M. Müller. Sheet Music-Audio Identification. In *ISMIR*, pages 645–650, 2009.

[7] C. Fremerey, M. Müller, and M. Clausen. Towards Bridging the Gap between Sheet Music and Audio. *Knowledge Representation for Intelligent Music Processing*, (09051), 2009.

[8] M. Grachten, M. Gasser, A. Arzt, and G. Widmer. Automatic Alignment of Music Performances with Structural Differences. In *ISMIR*, pages 607–612, 2013.

[9] Y. Han and C. Raphael. Informed Source Separation of Orchestra and Soloist Using Masking and Unmasking. In *ISCA-SAPA Tutorial and Research Workshop, Makuhari, Japan*, 2010.

[10] C.C.S. Liem, M. Müller, D. Eck, G. Tzanetakis, and A. Hanjalic. The Need for Music Information Retrieval with User-centered and Multimodal Strategies. In *Proceedings of the 1st international ACM workshop MIRUM*, pages 1–6. ACM, 2011.

[11] M. Müller, F. Kurth, and T. Röder. Towards an Efficient Algorithm for Automatic Score-to-Audio Synchronization. In *ISMIR*, 2004.

[12] G.A. Ten Holt, M.J.T. Reinders, and E.A. Hendriks. Multi-dimensional Dynamic Time Warping for Gesture Recognition. In *13th annual conference of the Advanced School for Computing and Imaging*, volume 119, 2007.

[13] R.J. Turetsky and D.P.W. Ellis. Ground-truth Transcriptions of Real Music from Force-aligned MIDI Syntheses. *ISMIR 2003*, pages 135–141, 2003.

[14] B. Yao, J. Ma, and L. Fei-Fei. Discovering Object Functionality. In *ICCV*, pages 2512–2519, 2013.