ACCURATE AUDIO-TO-SCORE ALIGNMENT FOR EXPRESSIVE VIOLIN RECORDINGS

Jia-Ling Syue[1*] **Li Su**[2*] **Yi-Ju Lin**[1] **Pei-Ching Li**[1]
Yen-Kuang Lu[1] **Yu-Lin Wang**[1] **Alvin W. Y. Su**[1]
[1] SCREAM Lab., Department of CSIE, National Cheng-Kung University, Taiwan
[2] Music and Culture Technology Lab., IIS, Academia Sinica, Taiwan
P76044457@mail.ncku.edu.tw, lisu@iis.sinica.edu.tw

ABSTRACT

An audio-to-score alignment system adaptive to various playing styles and techniques, and also with high accuracy for onset/offset annotation is the key step toward advanced research on automatic music expression analysis. Technical barriers include the processing of overlapped notes, repeated note sequences, and silence. Most of these characteristics vary with expressions. In this paper, the audio-to-score alignment problem of expressive violin performance is addressed. We propose a two-stage alignment system composed of the dynamic time warping (DTW) algorithm, simulation of overlapped sustain notes, background noise model, silence detection, and refinement process, to better capture the onset. More importantly, we utilize the non-negative matrix factorization (NMF) method for synthesis of the reference signal in order to deal with highly diverse timbre in real-world performance. A dataset of annotated expressive violin recordings in which each piece is played with various expressive musical terms is used. The optimal choice of basic parameters considered in conventional alignment systems, such as features, distance functions in DTW, synthesis methods for the reference signal, and energy ratios, is analyzed. Different settings on different expressions are compared and discussed. Results show that the proposed methods notably improve the conventional DTW-based alignment method.

1. INTRODUCTION

An audio-to-score alignment algorithm captures note-level information of a music performance with the aid of symbolic data such as MIDI. It has numbers of applications in the field of music information retrieval (MIR), such as automatic accompaniment [4], and music retrieval through matching a MIDI file to a polyphonic audio recording [7]. Besides, an effective audio-to-score alignment algorithm is also critical in computational music analysis, specifically in the case of extracting the note-level information in expressive music recordings for music expression analysis. For example, Li *et al.* [10] used an audio-to-score alignment algorithm [19] to annotate the onset and offset positions of each note in a dataset, including violin solo pieces interpreted by professional violinists with 10 expressive musical terms. [1] However, such annotation still needs to be checked and corrected manually as it suffers from low quality when there are overlapped sustain segments and unexpected silence between successive notes in expressive violin performance. Therefore, an improved audio-to-score alignment algorithm for expressive violin performance would be of great help to avoid such a tedious and labor-intensive process.

There have been numbers of audio-to-score alignment algorithms proposed in the past few decades, based on graphical models [2, 15, 16], hidden Markov models (HMM) [3, 6, 13, 18], and DTW [5, 12, 14]. Among these models, an HMM [13] is of better potential in modeling how the states of attack, decay or sustain evolve in a note, but to train the model parameters one needs amounts of correctly annotated data which are, as mentioned, hard to get without an improved audio-to-score alignment algorithm. Therefore, as an attempt to with low-resource data, we opt to use DTW with further processing steps, which can be implemented without the need of a large dataset.

Another challenge of matching a score to a musical recording is the diversity of timbre of the input signals, depending on the performer, instrument, recording environment, etc. Mismatch of spectral features between the MIDI-synthesized reference and the real violin sounds of the same event leads to errors in alignment. Such an issue was addressed by an HMM-based model exploiting the spectral templates adaptive to different recordings [9]. In this paper, we adopt a more straightforward approach,

Attribution: Jia-Ling Syue, Li Su, Yi-Ju Lin, Pei-Ching Li, Yen-Kuang Lu, Yu-Lin Wang, Alvin W. Y. Su. "Accurate audio-to-score alignment for expressive violin recordings", 18th International Society for Music Information Retrieval Conference, Suzhou, China, 2017.

[1] The SCREAM-MAC-EMT dataset compiled by Li *et al.* contains recordings of 10 different classical music pieces, each of which is interpreted with 5 selected expressions by 11 musicians [10]. It considers in total 10 expressive musical terms, including *Scherzando* (playful), *Tranquillo* (calm), *Con Brio* (bright), *Maestoso* (majestic), *Risoluto* (rigid), *Affettuoso* (affectionate), *Agitato* (agitated), *Cantabile* (like singing), *Grazioso* (graceful), and *Espressivo* (expressive). The experiments in this paper are performed on a subset of this dataset, which contains 50 recordings from randomly selected 3 musicians' performance.

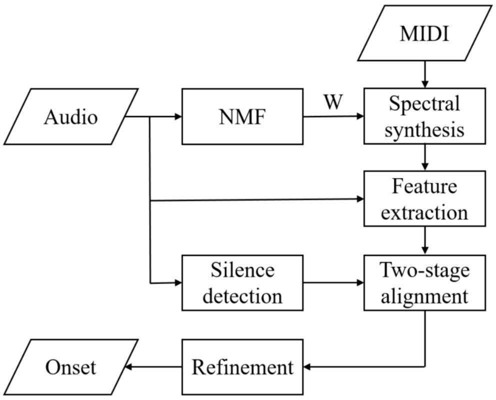

Figure 1. Flowchart of the proposed audio-to-score alignment system for expressive violin performance.

which utilizes the NMF method to directly learn the spectral template for synthesis under low-resource data.

By inspecting the data, we raise four issues which could be seen in aligning an expressive violin recording with its corresponding MIDI. First, the strong sustain of one note could be overlapped with the weak onset of its next note, and this makes the algorithm fail to accurately capture the onset time of the next note. Second, in a repeated note sequence, an algorithm is prone to erroneous onset detection since all notes have the same pitch. Third, the *staccato* technique usually causes unexpected silent segments which are not compatible to the ground-truth MIDI. Lastly, the background noise in a real-world environment may also cause mismatching. These issues are commonly seen; for instance, an inspection shows that in the dataset, 37% of the notes have overlapped sustain with their successive notes, 35% are in a repeated note sequence, 6% have unexpected silence, and 2% follow a rest symbol.

To this end, we add solutions to deal with the four problems: First, we simulate the note overlap in two-stage alignment (cf. Section 2.3). Second, we perform the duration ratio of repeated notes (cf. Section 2.4) and silence detection (cf. Section 2.2) to refine onset positions. Besides, we add a background noise template to model the rest parts in a recording (cf. Section 2.1). These processes are evaluated after a systematic experiment which finds the optimal parameters such as features, timbres for synthesis, distance measures, and energy measures in an audio-to-score algorithm (cf. Section 3.1). Moreover, we also discuss the precision of proposed alignment system within different levels of error tolerance, and draw an insight from analyzing the expression-wise performance (cf. Section 3.2).

2. THIS WORK

Figure 1 shows the diagram of the proposed alignment system, whose goal is to find the onset position accurately in expressive violin performance. The system takes an audio signal and its corresponding MIDI file as input. We adopt the NMF to learn the spectral patterns of the audio input of violin solo for MIDI-to-audio synthesis. Then, either chroma or log-frequency spectral features, which are the basic parameters considered in conventional alignment systems (cf. Section 3.1), are extracted from both the audios. Incorporated with a silence detection process, a two-stage DTW-based audio-to-score alignment and a refinement process are conducted for resolving the discrepancies between audio and MIDI in expressive recordings including overlapped sustain segments, unexpected silence segments, and repeated note sequences.

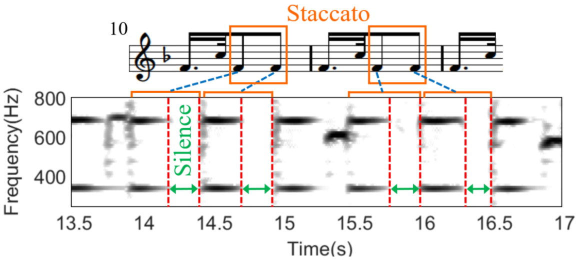

Figure 2. The audio played with *staccato* has extra silence segments (green) that could not be found from the score.

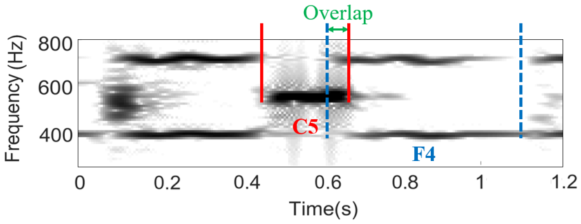

Figure 3. An example of a note with strong energy in sustain segment overlaps with its successive note, which has weak energy in attack segment.

2.1 NMF and Spectral Synthesis

For MIDI-to-audio conversion, the NMF with the Kullback-Leibler divergence [17] is adopted to train the spectral patterns of each pitch in a recording. The NMF decomposes an audio spectrogram V into two matrices W and H, i.e. $V \approx \hat{V} = WH$, where W is a spectral template represented in column, and H is a time-varying energy activation represented in row. Following Joder *et al.* [8], we adopt a Gaussian mixture model to initialize the template matrix W for each pitch with k Gaussian functions centered at the fundamental frequency and the first k-th harmonics of the pitch. Due to the weak energy in the high-frequency range, we take k to be 4, the weight of each Gaussian function to be k^{-2}, and the variance to be 30 cents. Besides, we consider the frequency range from 65 Hz to 4 kHz, which removes the high- and low-frequency noise. The activation matrix H is initialized by a normal distribution with zero mean and unit standard deviation. Moreover, we add an additional *noise template* (NT) with random numbers in [0, 1] to simulate the noise of silence parts in the recorded audio signal. Furthermore, we adopt a preprocessing step of stretching or shortening the reference signal by insertion or deletion of frames so as to make

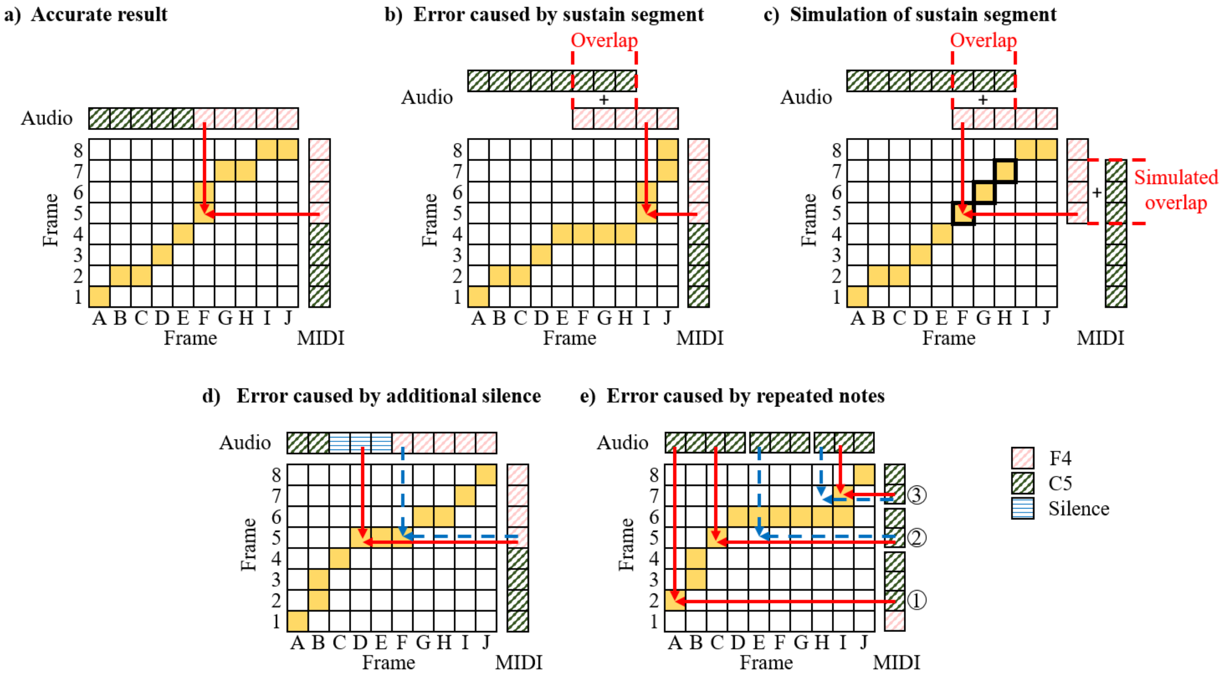
a) Accurate result
b) Error caused by sustain segment
c) Simulation of sustain segment
Overlap
Overlap
Simulated overlap
Audio
Frame
MIDI
A B C D E F G H I J
1 2 3 4 5 6 7 8
+
d) Error caused by additional silence
e) Error caused by repeated notes
①
②
③
F4
C5
Silence

Figure 4. Examples of accurate and erroneous alignment paths: (a) accurate result, (b) error caused by overlapped sustain segment, (c) simulation of overlapped sustain segment, and error caused by (d) additional silence and (e) repeated notes.

its length similar to the input, in order to reduce the effect of tempo changes in expressive violin performance.

2.2 Silence Detection

Violinists use different playing techniques to interpret distinct expressions. The *staccato* technique might be the one which is most likely to cause errors in DTW alignment among others due to the silence segments caused by articulation of successive notes. Such a silent segment do not have any information in the reference signal but could be found in the audio recording as shown in Figure 2, it results in deterioration in the DTW alignment path. This issue is addressed by introducing an extra silence detection process with energy measurement. From the fast Fourier transform (FFT) [2], the energy curve is computed by summing the spectrum over all the frequency bins and is expressed in dB scale. A silence segment is one which is longer than 100ms and whose energy is less than 12 dB.

2.3 Two-stage Alignment

The main purpose of the proposed two-stage audio-to-score alignment process is to capture the accurate onset for overlapped successive notes where the former note has a *long sustain* and the latter one has a *soft onset*. Such a specific energy characteristic of violin is likely to cause wrong alignment paths. As illustrated in Figure 3, the first note `C5`, which has strong energy in sustain segment, overlaps with the second note `F4`, which has weak energy in the attack, and leads to a distorted alignment path. Figure 4(a) shows the ideal accurate result of this example: the onset of `F4` is at position $(\mathrm{F}, 5)$, while Figure 4(b) shows the actual erroneous result of `F4`, where the onset of `F4` locates at $(\mathrm{I}, 5)$. This is because the first 3 frames of `F4` are submerged in the sustain segment of `C5`.

The two-stage alignment process is proposed to solve this problem. In the first stage, we adopt the conventional DTW-based alignment as our *baseline*, and obtain a rough estimation of the onset of each note. Next, we add the information of the silence segments mentioned in Section 2.2. If there is no silence detected between two successive notes, then the two notes are considered overlapped. For every pair of the overlapped notes, we lengthen the first note of the reference MIDI with a duration of 3 frames, in order to simulate the behavior of overlapped notes. [3] Then, we perform DTW again to align the audio with the modified MIDI. This is the *second-stage alignment* (SA) process. The SA process of the overlapped notes `C5`-`F4` is shown in Figure 4(c). The two-stage alignment is therefore a combination of the baseline process and the SA process.

2.4 Refinement

The refinement process contains two tasks. The first task is to fix the case of *staccato* and *rest*, where the errors of alignment are usually caused by *silence* rather than overlapping between two consecutive notes. Figure 4(d) shows the alignment result of an audio played with *staccato*. The ideal accurate onset of `F4` is at position $(\mathrm{F}, 5)$ as pointed by the dashed-line arrow; however, the actual result of `F4`

[2] This paper uses a 2,048-point Hamming-windowed FFT with each frame staggered by 882 samples (20ms) throughout all the experiments.

[3] We observe that for most of the overlapped notes, the overlapping durations are between 2 and 4 frames (40ms and 80ms). Such an observation gives to an estimation 3 frames.

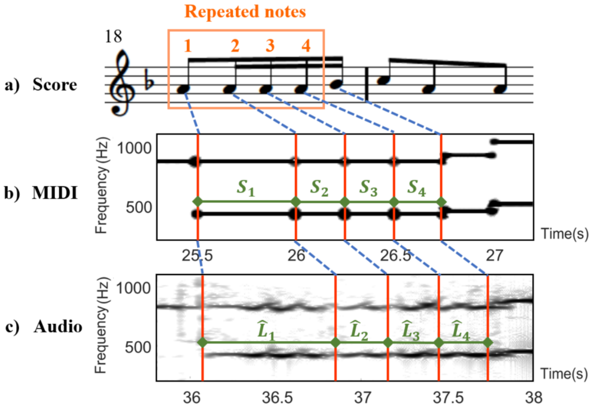

Figure 5. An example of repeated notes, whose spectrum is highly similar to each other. Their onsets are calibrated via the duration ratio estimated from the reference signal.

locates at $(\mathrm{D}, 5)$, a position within the silence segment, as pointed by the solid-line arrow. To address this issue, the *refinement of silence* (RS) is implemented as follows: if the onset of an aligned note locates in a silence segment of the audio, then it will be shifted to the correct position, that is, the frame right after such a silence segment.

The second task is to adjust the duration of repeated notes, where the alignment path is hard to estimate because of high similarity among the note spectra. This issue is illustrated in Figure 4(e), where one can see that except for the first C5, the onset results of the other notes are at the wrong positions, $(\mathrm{C}, 5)$ and $(\mathrm{I}, 7)$, respectively, as pointed by the solid-line arrows. The correct onsets of the second and the third C5 are the positions pointed by the dashed-line arrows. Figure 5 shows a real-world example of repeated notes in a more detailed manner: the spectra of the four repeated notes A4 are highly similar, especially for the last three notes with the same note type. It is hard to figure out the accurate onset of each repeated note when performing DTW alignment.

We therefore come up with a strategy to deal with this problem. Our assumption is that both audio and its corresponding MIDI file have approximately similar duration ratio of repeated notes. In other words, we can modify the onsets of repeated notes by referring to the duration ratio of such notes in the reference signal. Given a sequence of repeated notes, $r = (r_1, r_2, ..., r_m)$, the *refinement of repeated notes* (RRN) is realized as follows:

STEP 1 Computing the duration of each repeated note, $S = (S_1, S_2, ..., S_m)$ and $L = (L_1, L_2, ..., L_m)$, according to the reference signal and the onset results of the two-stage alignment, respectively.

STEP 2 Calculating the duration ratio for each repeated note, i.e. $\mathrm{ratio}_k = S_k / \sum_{k=1}^{m} S_k$.

STEP 3 Estimating the predicted duration for such notes in the audio recording, $\hat{L} = (\hat{L}_1, \hat{L}_2, ..., \hat{L}_m)$, via the calculation of $\hat{L}_k = L_k \times \mathrm{ratio}_k$.

Timbre	MIDI synthesizer		NMF
Feature	12-D chroma	84-D linear-log spectrum	
Precision	72.68	81.37	95.12

Table 1. Comparison of precision (in %) using chroma and the 84-D spectrum feature. Since the 84-D spectrum performs better, the timbre synthesis via MIDI synthesizer or NMF is considered.

STEP 4 Adjusting the duration of the first $m-1$ repeated notes according to the criterion of $|\hat{L}_k - L_k| > \theta$. If $\hat{L}_k > L_k$ then the onset of the $(k+1)$-th note is shifted backward by $|\hat{L}_k - L_k| - \theta$ ms. Besides, L_{k+1} is also updated by the shifted value. On the contrary, it is updated by shifting forward $|\hat{L}_k - L_k| - \theta$ ms, and so does the L_{k+1}.

Our pilot study shows that choosing $\theta = 60$ ms gives better performance.

3. EVALUATION

The experiments are separated into two parts. The first part is to find the optimal setting used in the conventional DTW-based alignment. The second part is the results of the proposed system, including the performance of baseline process, proposed processes, and expression-wise. The test dataset contains 10 expressions, each with 5 classical music pieces, totaling 50 excerpts (2,925 notes). We use precision as our evaluation method, which is the percentage of the number of correct onsets among all the excerpts. A correct onset is defined as the difference between aligned onset time and its corresponding ground-truth onset time, being less than 100ms.

3.1 Factors Experiment

We consider four types of factors: feature, timbre, distance function, and energy. The brief introduction and results of each factor are described one by one as follows.

Two features are considered: chroma [1] and linear-log frequency spectrum [5]. The chroma is a 12-dimensional vector representing the energy of the 12 pitch classes (i.e. C, C#, ..., B). The linear-log frequency spectrum is a spectral feature with reduced dimension, performed by a 84-D filterbank, which is in linear scale in the low frequencies and in logarithm scale in the high frequencies [5]. Such a feature simulates the linear-log frequency sensitivity in human auditory systems. Table 1 compares the two features according to the averaged precision (in %) of all the 50 excerpts. The results indicate that the linear-log spectrum is better than the chroma. We therefore select the 84-D spectrum for the feature factor.

Then, we compare two methods of synthesizing the reference signal from MIDI, where the one is directly through a MIDI synthesizer, and the other uses the NMF to learn the spectral features from the original audio recording, a similar strategy of the HMM-based timbre learning method

E_a/E_m	13 dB	10 dB	0 dB	-10 dB	-13 dB
Cosine	96.10	96.10	**96.14**	96.00	96.00
Euclidean	77.81	88.62	95.12	78.39	60.62
SKL	95.73	95.86	95.41	96.85	96.75

Table 2. Comparison of precision (in %) using three types of distance functions for DTW with five different levels of energy ratios of audio recording to reference signal.

Process	Precision
Baseline	96.14
Baseline+NT	97.03
Baseline+NT+SA	97.64
Baseline+NT+SA+RS	97.88
Baseline+NT+SA+RS+RRN (‘Proposed’)	**98.43**

Table 3. Performance (in %) of the baseline and the proposed system. NT: noise template; SA: second stage alignment; RS: refinement of silence; RRN: refinement of repeated notes.

Process		NT	SA	RS	RRN	Other
# Notes		45	1094	173	1015	598
# Errors	Baseline	15	42	6	40	10
	Proposed	2	26	1	10	7

Table 4. Comparison of the number of error notes between the baseline and the proposed system based on the four raised issues.

by Joder *et al.* [9]. To simulate the silence parts, we simply apply zero values for silence segments, which is the same means used in the MIDI synthesizer. Results in Table 1 also shows that using NMF for timbre synthesis yields much better precision (95.12%) than using a MIDI synthesizer (81.37%), since a common MIDI synthesizer can not well resemble the wide variety of timbre in expressive violin performance. We therefore take the NMF-based synthesis method for the following experiments.

Moreover, since the dynamics of notes vary largely in expressive violin performance, the distance functions in DTW and the frame-level energy are also essential factors in the alignment process. We compare three types of distance functions in the DTW algorithm: cosine similarity [11], Euclidean distance [5], and symmetric Kullback-Leibler (SKL) divergence [9]. Cosine similarity is the normalized inner product of two non-zero vectors. Since we would like to find the minimal value of the cost function, the inner product is subtracted by 1. Besides, Euclidean distance calculates the straight-line distance of two feature vectors. Further, SKL divergence is defined as: $d_{SKL}(i,j) = d_{KL}(i \parallel j) + d_{KL}(j \parallel i)$, where i and j are two n-dimensional vectors. The performance of a distance function is highly related to the effect of *energy ratios*, i.e. the ratio of the energy levels of the audio recording (E_a) to the reference signal (E_m). Table 2 presents the averaged precision values using the three distance functions for DTW with five different levels of energy ratio from -13 dB to 13 dB. All the three distance functions have similar performance when audio and reference signals have similar levels of energy. However, when the energy ratio exceeds 10 dB, performance degrades significantly for Euclidean distance, while the cosine similarity turns out to be the most stable distance function among all levels of energy ratio (STD=0.06%). Therefore, we opt to use cosine similarity in the following experiments.

In short, the optimal setting of the conventional DTW-based alignment algorithm (i.e. the *baseline*) encompasses linear-log spectral features, the reference signal synthesized with NMF on the input signal, and cosine similarity as a distance function. We will use these settings in the following experiments if not mentioned.

3.2 System Experiment

3.2.1 Overview

Table 3 lists the performance of the proposed system and a comparison to the individual building blocks mentioned in Section 2, i.e. one-stage DTW only (Baseline), noise template (NT), second-stage alignment (SA), refinement of silence (RS), and refinement of repeated notes (RRN). Results show that the baseline achieves a precision of 96.14%. Its performance is then increased by 0.89% after adding NT. A further improvement of 0.61% is seen after adding SA and addressing the issue of overlapped sustain notes. Finally, the RS and RRN also give a slight improvement of 0.24% and 0.55% subsequently. As a result, the averaged precision of the proposed system comes to 98.43%, showing a significant improvement from the baseline system as validated by a two-tailed t-test ($p <$ 0.05, d.f.=98).

Table 4 gives a more in-depth comparison of the number of error notes between the baseline and the proposed system according to the four raised issues. [4] We find that every process reduces the number of error notes of their corresponding types, and the RRN process leads to the greatest improvement: 40 error notes within repeated note sequences are reduced to 10 notes.

Table 5 shows the precision of the proposed system within different levels of error tolerance values not only at 100ms but ranged from 20ms (1 frame) to 700ms (35 frames). We find that the performance is over 90% when using the tolerance with 60ms (3 frames). In addition, the maximal erroneous time of onset is within 700ms.

3.2.2 Expression-wise Performance

Table 6 presents the averaged precision for the 10 violin expressions based on the Baseline process with two distinct timbre synthesis methods, MIDI synthesizer and NMF, and the proposed system, respectively. Comparing the MIDI

[4] An NT refers to a note which follows a rest symbol; SA counts the note that overlaps its successive notes over 60ms; RS includes the notes played with *staccato*; RRN contains the notes belonging to a sequence of repeated notes; the remaining ones are marked as‘Other’.

Error ≤		Proposed
Frames	Seconds	
1	0.02	58.60
2	0.04	85.61
3	0.06	94.84
5	0.1	98.43
10	0.2	99.62
15	0.3	99.79
20	0.4	99.83
25	0.5	99.93
30	0.6	99.97
35	0.7	100.00

Table 5. Performance (in %) of the proposed system within different levels of error tolerance values.

Expression	Baseline		Proposed
	MIDI	NMF	
Scherzando	78.25	96.37	96.98
Tranquillo	69.65	93.06	98.55
Con Brio	87.19	98.44	98.75
Maestoso	82.73	97.48	98.56
Risoluto	76.86	96.92	99.49
Affettuoso	87.41	96.67	100.00
Agitato	88.78	97.96	96.94
Cantabile	86.43	93.97	95.98
Grazioso	86.44	95.25	99.66
Espressivo	78.07	95.35	98.00

Table 6. Performance (in %) of the ten violin expressions via the baseline process with two distinct timbre synthesis methods and the proposed system.

synthesizer (i.e. the second column) to NMF-based synthesis from audio recording (i.e. the third column), a significant improvement can be observed (p < 0.005, d.f.=18), especially for *Tranquillo* and *Risoluto*, where the improvement is over 20% for both cases. Besides, the proposed system (i.e. the fourth column) has significant improvement from both the Baseline cases (p < 0.05, d.f.=18). Particularly, *Tranquillo*, *Grazioso*, and *Affettuoso* are improved the most; this implies that the proposed system can enhance the onset precision for such violin performance with plentiful expression and intense vibrato. For *Risoluto*, the expression played with *staccato* technique mostly, the proposed system also gives excellent result. Furthermore, we see that the improvement of *Con Brio* and *Scherzando* is limited, probably due to their intense characteristics of performance such as clear attack of energy envelope.

3.3 Discussion

According to the expression-wise performance as illustrated in Table 6, we find that *Agitato* is the only one expression which has degraded precision via the proposed system. The reason is possibly that the energy of sustain segment might be weak such that the simulation of sustain perhaps cause additional errors. Except for *Agitato*, the proposed system has improvement for other expressions.

Although we use a refinement process to deal with the unexpected silence segments caused by the *staccato* technique, this process actually could be merged into the two-stage alignment. For example, we can adopt similar means which is used in the simulation of overlapped sustain notes, by inserting additional frames in a reference signal based on the information of silence segments. Thereby, the system will be made more succinct.

In this paper, we only consider a subset of the violin expression dataset, which includes 50 solo recordings from randomly selected 3 musicians' performance. In order to obtain more reliable performance and to develop a robust alignment system, the test data needs to be expanded such as using the recordings from other musicians in the SCREAM-MAC-EMT dataset as well as data of polyphonic recordings, where the latter suggest a future work of constructing a new dataset for expression analysis of violin solo in polyphonic music.

Moreover, this study only considers the accuracy of onset-only alignment. Another important task for music expression analysis of notes is offset alignment, which is still a challenging problem. An extension of the proposed alignment system such as to cover the offset alignment issue is also left as future work.

4. CONCLUSION

To have better expression analysis of violin recordings, it is desired to have the precise onset information of each note. The conventional DTW algorithm is modified for accurate audio-to-score alignment for the violin dataset, including the simulation of sustain notes, silence detection, refinement of duration ratio of repeated notes, and background noise model, which are used to deal with the four common issues usually seen in violin recordings. Experiments show that high precision is achieved if instrumental timbre and the 84-dimensional spectral feature vector are used. Cosine similarity is adopted as our distance formula for its robustness to various violin playing techniques. The proposed two-stage alignment system obtains significant improvement, not only for the addressed issues but also for the distinct expressions, from the baseline process.

5. ACKNOWLEDGEMENTS

The authors would like to thank the Ministry of Science and Technology of Taiwan for its financial support of this work, under contract MOST 105-2221-E-006-148.

6. REFERENCES

[1] M. A. Bartsch and G. H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on multimedia*, pages 96–104, 2005.

[2] B. Catteau, J. P. Martens, and M. Leman. A probabilistic framework for audio-based tonal key and chord

recognition. *Advances in Data Analysis*, pages 637–644, 2007.

[3] A. Cont. Realtime audio to score alignment for polyphonic music instruments, using sparse non-negative constraints and hierarchical HMMs. In *ICASSP*, pages 245–248, 2006.

[4] R. B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. *Communications of the ACM*, pages 38–43, 2006.

[5] S. Dixon. Live tracking of musical performances using on-line time warping. In *DAFx*, pages 92–97, 2005.

[6] Z. Duan and B. Pardo. A state space model for online polyphonic audio-score alignment. In *ICASSP*, pages 197–200, 2011.

[7] N. Hu, R. B. Dannenberg, and G. Tzanetakis. Polyphonic audio matching and alignment for music retrieval. In *WASPAA*, pages 185–188, 2003.

[8] C. Joder, S. Essid, and G. Richard. Optimizing the mapping from a symbolic to an audio representation for music-to-score alignment. In *WASPAA*, pages 121–124, 2011.

[9] C. Joder and B. Schuller. Off-line refinement of audio-to-score alignment by observation template adaptation. In *ICASSP*, pages 206–210, 2013.

[10] P.-C. Li, L. Su, Y.-H. Yang, and A. W. Y. Su. Analysis of expressive musical terms in violin using score-informed and expression-based audio features. In *ISMIR*, pages 809–815, 2015.

[11] R. Macrae and S. Dixon. Accurate real-time windowed time warping. In *ISMIR*, pages 423–428, 2010.

[12] B. Niedermayer and G. Widmer. A multi-pass algorithm for accurate audio-to-score alignment. In *ISMIR*, pages 417–422, 2010.

[13] N. Orio and F. Déchelle. Score following using spectral analysis and hidden markov models. In *ICMC*, pages 151–154, 2001.

[14] C. Raffel and D. P. W. Ellis. Optimizing DTW-based audio-to-midi alignment and matching. In *ICASSP*, pages 81–85, 2016.

[15] C. Raphael. A hybrid graphical model for aligning polyphonic audio with musical scores. In *ISMIR*, pages 387–394, 2004.

[16] C. Raphael. Aligning music audio with symbolic scores using a hybrid graphical model. *Machine learning*, pages 389–409, 2006.

[17] T. Virtanen. Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria. *IEEE Transactions on Audio, Speech, and Language Processing*, pages 1066–1074, 2007.

[18] S. Wang, S. Ewert, and S. Dixon. Robust and efficient joint alignment of multiple musical performances. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pages 2132–2145, 2016.

[19] T.-M. Wang, P.-Y. Tsai, and A. W. Y. Su. Note-based alignment using score-driven non-negative matrix factorisation for audio recordings. *IET Signal Processing*, pages 1–9, 2014.