

# UTILITY SYSTEM FOR CONSTRUCTING DATABASE OF PERFORMANCE DEVIATIONS

*Ken'ichi Toyoda, Kenzi Noike, Haruhiro Katayose*  
Kwansei Gakuin University  
Gakuen, Sanda, 669-1337 JAPAN  
{toyoda, noike, katayose}@ksc.kwansei.ac.jp

## ABSTRACT

Demand for music databases is increasing for the studies of musicology and music informatics. Our goal is to construct databases that contain deviations of tempo, and dynamics, start-timing, and duration of each note. This paper describes a procedure based on hybrid use of DP Matching and HMM that efficiently extracts deviations from MIDI-formatted expressive human performances. The algorithm of quantizing the start-timing of the notes has been successfully tested on a database of ten expressive piano performances. It gives an accuracy of 92.9% when one note per bar is given as the guide. This paper also introduces tools provided so that the public can make use of our database on the web.

**Keywords** Database, HMM, DP matching

## 1. INTRODUCTION

Music databases can contribute to studies of musicology and music informatics. The information that music databases should provide is diverse: from the acoustics sounds of musical instruments to SMF files, etc. As for the acoustics and sounds of musical instruments, a copyright-free database has been provided for academic use by Goto et al. [1].

This paper deals with a database of musical deviation that characterizes music "expressiveness." In ethnomusicology, there are reports that focus on musical deviations. For example, Bell collected musical deviations of Indian music (RAGA) [2]. As for western tonal music, for instance, CCARH at Stanford University is engaged in the development of large databases of musical and textual materials [3]. However, there are few databases of deviations available for academic use. Widmer's project is one of the few that have obtained musical deviations in western music [4]. The researchers have collected the deviations from Horowitz' performances and have been trying to classify the performances into certain styles. The derived devia-

tions are limited to transitions of tempi and general dynamics, and the database is yet to be published.

Our goal is to construct databases that contain deviations regarding tempo and dynamics, start-timing, and duration of each note. This paper describes, as the first step in constructing databases, a method to obtain deviation data from MIDI-formatted music and tools for testing the utility of the database.

The first thing that we have to do to obtain deviations is quantization. Quantization is one of the principal competences in transcribing music. In the 1980's, effective quantization methods were proposed [5][6][7]. Recently, it has been reported that approaches based on statistics are highly effective [8][9][10]. However, it is still difficult to secure accuracies of more than 90% in quantizing expressive music, and such high accuracy is needed to reveal the deviations. Moreover, manual correction to cover the errors is more troublesome.

The other practical way of obtaining error-free data is to utilize score guides. Katayose et al. derived deviations of each note, using dynamic programming matching (DP) between the musical sound and its score [11]. This method is reliable enough to obtain error-free data. However, the input of score data is itself time-consuming.

This paper proposes an efficient procedure for compiling a deviation database, based on hybrid use of a matching procedure and a quantization procedure. A coarse DP matching between the performance and sparse guides given by the user contribute to limit the freedom of tempo changes. A Hidden Markov Model (HMM) is used for assigning the time values to the notes between the spans fixed by the matching procedure.

The next section describes the design of the data format. We then describe the procedures to obtain expression templates and show some experimental results. Finally, we discuss the effectiveness of the database together with some tools for utilizing it.

## 2. MUSICAL EXPRESSION AND DATA FORMAT

### 2.1. How to describe musical expressions

There are various control channels and methods for playing musical instruments. Enumeration of controllers of the delicate nuance of acoustic sound, itself, may be one

```

.....
.....
2.00 BEATTIME 475.436 4
2.00 (0.00 E3 78 3.00 -0.11)
=2
1.00 BEATTIME 468.384 4
1.00 (0.00 C#4 76 0.75 -0.09) (0.04 E1 60 1.00 -0.13)
1.75 (0.10 D4 77 0.25 -0.14)
2.00 BEATTIME 461.538 4
2.00 (0.00 B3 75 1.00 -0.03) (0.00 G#3 56 1.00 0.03)
3.00 BEATTIME 469.851 4
3.00 (0.00 B3 72 1.00 0.00) (0.09 G#3 56 1.00 -0.12) (0.14 D3 57 1.00 -0.21)
=3
1.00 BEATTIME 470.219 4
1.00 (0.00 B3 77 2.00 -0.05) (0.00 G#3 47 2.00 -0.05) (-0.06 D4 57 2.00 -0.32)
2.00 BEATTIME 470.219 4
3.00 BEATTIME 462.606 4
3.00 (0.00 F#4 75 1.00 -0.15) (0.00 D4 54 1.00 0.03)
=4
1.00 BEATTIME 469.851 4
1.00 (0.00 D#4 73 0.75 -0.38) (0.02 C4 65 0.75 -0.08)
.....
.....

```

**Figure 1.** Data Format of Deviation Database

of the important study targets. Limited to keyboard instruments represented by piano, the controllers to play the music are start timing, duration, and dynamics (velocity in MIDI) of each note and pedal operation. We designed the NOTE format, in which the deviations from the canonical description were separately described for each event.

## 2.2. Format

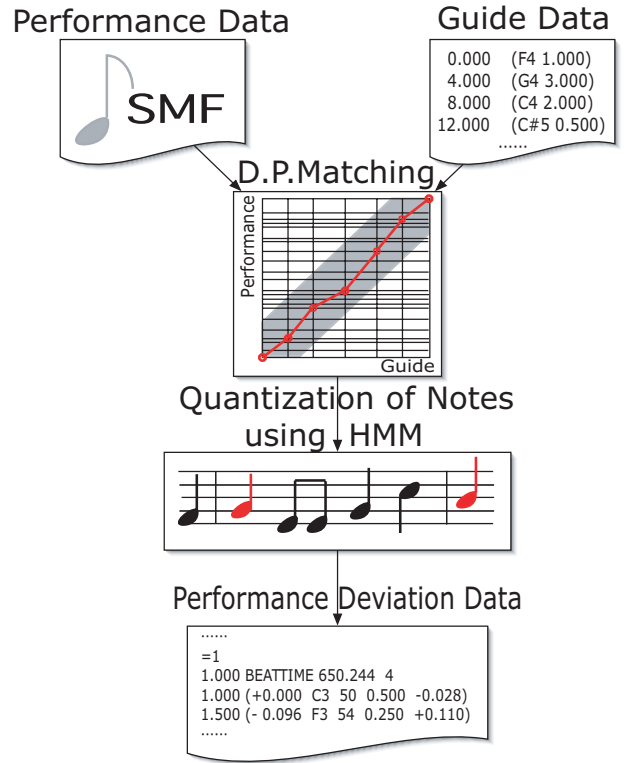
Figure 1 shows a part of the deviation data in the database [12]. The left column represents the start timing of the events. Information about each note, except for the start timing, is placed in brackets. Each bracketed term, in order, represents, the deviation of start time, note name, velocity, duration, and the deviation of duration. In this format, the tempo is described using the descriptor **beattime**. The description is followed by the note length in milliseconds and its note value.

## 3. SYSTEM OVERVIEW

The approach to extracting deviations is as follows (see Figure 2):

1. Prepare performance data recorded in MIDI format and its guide data described in NOTE format. The guide data is translated from SMF (standard MIDI file) or MusicXML data by using the application ‘SMF2note’[13].
2. Match performance notes and guide notes by using DP Matching. In this process, the canonical onset timing of the performance’s notes and local tempi are partly given.
3. Quantize the onset timing of notes between the spans fixed by the matching procedure by using a Hidden Markov Model (HMM) of which hidden states correspond to the canonical onset timing and the outputs represent the fluctuating onset timing.
4. Calculate local tempi<sup>1</sup> deviations of onset timing

<sup>1</sup> The positions where calculated tempi (for example, per downbeat or per bar) can be given by the user.



**Figure 2.** System Overview : Extraction of deviation data with hybrid use of DP matching and HMM quantization

and those of the durations and assign velocities to the identified notes.

5. Output the deviation data

## 4. MATCHING PERFORMANCE NOTES AND GUIDE NOTES

This section describes the method to match the performance notes and guide notes by using DP [14]. This procedure reduces the errors of the following quantization procedure.

### 4.1. DP Matching

Each node  $P(k, l)$  of the trellis represented in the middle of Figure 2 denotes the correspondence of the performance note and guide note.  $P(k, l)$  has a value  $S(k, l)$  which denotes the similarity of the correspondence depending on the route from the origin node to  $P(k, l)$ .  $S(k, l)$  is calculated as:

$$\begin{aligned}
 S(0, 0) &= 0 \\
 S(k, l) &= \max(\text{sim}(i, j, k, l) + S(i, j)) \quad (1) \\
 k - m &\leq i < k, l - n, \leq j < l
 \end{aligned}$$

where  $m, n$  are constants and  $\text{sim}(i, j, k, l)$  denotes the similarity of the correspondence which depends on the route from  $P(i, j)$ , the parent node of  $P(k, l)$ , to  $P(k, l)$ .

$P(k, l)$ 's parent node which maximizes  $S(k, l)$  is also determined by equation (1). In this regard, however, a node denoting the correspondence of two notes having different pitches is not assumed to be the candidate of  $P(k, l)$ 's parent. The most appropriate correspondence of performance and guide notes for the given song is obtained by tracking back parent nodes from the terminal node to the origin node.

## 4.2. Similarity of correspondence

$sim(i, j, k, l)$  in the equation (1) is calculated as:

$$sim(i, j, k, l) = \sum_{n=1}^3 w_n \cdot s_n(i, j, k, l),$$

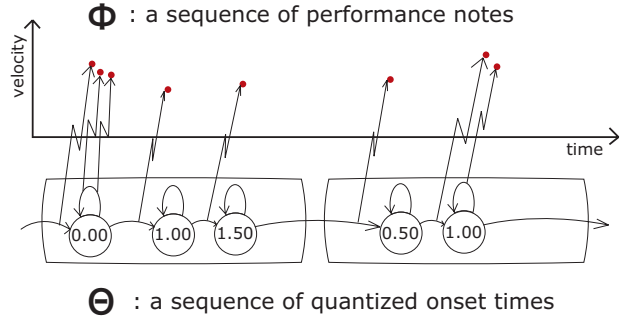
where  $s_1, s_2, s_3$  mean as follows:

- $s_1$  denotes the similarity between the local tempo determined by guide notes and the average tempo of a whole song.  $s_1$  reflects the assumption that, though local tempi does not fluctuate much compared with the fluctuation of the whole tempo transition.
- $s_2$  denotes the similarity between the onset time of a guide note and that of a performance note. Here, both are compared on the millisecond time scale. The onset times of guide notes are converted from the beat scale to the millisecond time scale by using the average tempo of the whole song.
- $s_3$  denotes the likelihood of the velocity of the performance note in terms of the canonical onset timing of the guide note. When the velocity takes on a large value at the downbeat,  $s_3$  also takes on a large value, and vice versa.

$w_n$  denotes the weighting factor of  $s_n$ . We determined  $s_n$  according to pilot studies.

## 5. QUANTIZATION USING HMM

This section describes the quantization procedure of onset timing based on the HMM. The proposed method follows Otsuki's approach in part [8]. Otsuki's algorithm estimates the canonical onset timing of chords (merging chords) as a preprocessing. However, with Otsuki's algorithm, there may be an initial error in the whole quantization procedure. In contrast, our quantization algorithm for chords does not require the preprocessing for merging chords. Our approach represents simultaneous onset timings as auto-regressions of a state in the HMM, and their detection is embedded in the quantizing process. We can find the same idea in the recent work of Takeda et al. [10]. The difference between our algorithm and Takeda's is that ours embeds in the HMM the factors regarding not only IOIs (Inter Onset Intervals) but also their ratios and velocities.



**Figure 3.** A hidden Markov model which has canonical onset timing as hidden states and fluctuated onset timing and velocities as outputs

### 5.1. Formulation of Quantization using HMM

Let  $\Phi = (\phi_1, \phi_2, \dots, \phi_T)$  denote a sequence of feature vectors whose elements are onset timing and velocities of performance notes.  $\Theta = (\theta_1, \theta_2, \dots, \theta_T)^2$  denotes a sequence of canonical onset timings. Here, estimating the intended  $\Theta$  from the observed  $\Phi$  is assumed to be the same as finding the  $\Theta^*$  that maximizes the a posteriori probability,  $P(\Theta|\Phi)$ . According to Bayes theorem,  $\Theta^*$  also maximizes  $P(\Phi|\Theta)P(\Theta)$ . The above description is formulated as follows:

$$\Theta^* = \arg \max_{\Theta} P(\Phi|\Theta)P(\Theta) \quad (2)$$

We assume further that the probability of  $\theta_t$  appearing depends only on the preceding  $\theta_{t-1}$  and that  $\Theta$  could be described as a chain of units called ‘‘Rhythm Words[8]’’ which means small units having states corresponding to the canonical onset timing. Then,  $\Theta^*$  can be rewritten as:

$$\Theta^* = \arg \max_{\Theta \in G_w} \prod_{t=1}^T a_{\theta_{t-1}\theta_t} b_{\theta_t}(\phi_t) \quad (3)$$

where  $G_w$  denotes the whole set whose elements are chains of Rhythm Words obtained from 77 musical scores made in advance,  $a_{\theta_{t-1}\theta_t}$  denotes the transition probability from  $\theta_{t-1}$  to  $\theta_t$  and  $b_{\theta_t}(\phi_t)$  denotes the probability that  $\phi_t$  emerges from  $\theta_t$ . The sequence of intended onsets  $\Theta^*$  is estimated based on equation (3), where, giving credit to the preceding matching process, any  $\Theta$  conflicting with the guide's onset timing is excluded from the candidates.

### 5.2. Factors regarding performance fluctuations

The IOI, ratio of IOIs, and velocity were characterized as follows:

- IOI  
 $b_{q_t}$  denotes the probability that a canonical IOI  $q_{\theta_t}(= \theta_{t+1} - \theta_t)$  is observed as an IOI  $q_{\phi_t}(= \phi_{t+1} - \phi_t)$ .

<sup>2</sup> In what follows,  $\theta_t$  denotes both the value and the corresponding state.

**Table 1.** Results of quantizing onset timing : The top row represents the amount of guide notes. For example, ‘per 1 bar’ represents that the note whose onset timing is the earliest in each bar is given as a guide.

title	start and end notes		1 note per 2 bars		1 note per 1 bar		1 note per 1 beat		1 note per 1 chord	
	threshold	HMM	threshold	HMM	threshold	HMM	threshold	HMM	threshold	HMM
<i>Minuet</i>	8.1	99.0	84.4	99.8	93.8	99.8	99.0	99.8	99.5	99.8
<i>Prelude</i>	26.4	96.9	97.4	96.9	97.8	96.9	99.1	98.7	100	99.3
<i>Gavotte</i>	11.7	91.5	92.9	93.9	99.4	99.4	99.4	99.7	99.6	99.7
<i>Waltz No.7</i>	8.3	86.2	57.1	96.8	74.0	96.8	97.8	96.8	100	100
<i>Cuckoo</i>	19.4	34.6	88.3	85.7	98.0	94.0	98.0	97.7	99.6	99.4
<i>Little serenade</i>	10.4	57.2	58.4	86.6	95.7	100	99.7	100	99.7	100
<i>für Elise</i>	11.6	23.1	65.6	77.8	78.1	81.7	96.3	97.8	99.7	100
<i>Troimerai</i>	7.8	55.6	63.8	80.3	76.8	95.3	98.0	98.4	98.9	100
<i>K.331</i>	7.6	43.5	46.4	78.5	67.1	95.4	98.3	98.7	99.6	99.6
<i>Raindrop</i>	19.5	27.4	38.1	48.5	54.5	69.5	85.7	86.0	99.6	99.9
<i>average</i>	13.1	61.7	69.2	84.5	83.5	92.9	97.1	97.4	99.6	99.8

In this case,  $b_q$  is assumed to follow a Gaussian distribution whose mean is  $q_{\theta_t}$ .

- ratio of IOIs

Let  $r_{\theta_t}$  denote the ‘‘canonical’’ ratio of adjacent two IOIs, that is  $q_{\theta_t}$  to  $q_{\theta_{t-1}}$ .  $b_{r_t}$  denotes the probability that a canonical  $r_{\theta_t}$  is observed as an  $r_{\phi_t}$ . In this case,  $b_{r_t}$  is also assumed to follow a Gaussian distribution whose mean is  $r_{\theta_t}$ .

- velocity

$b_{v_t}$  denotes the probability that the  $t_{th}$  note’s velocity is observed as  $v_{\theta_t}$ .  $v_{max}$  denotes the maximum velocity in the several notes in the vicinity of the  $t_{th}$  note. If the  $t_{th}$  note corresponds to the down-beat,  $b_{v_t}$  is assumed to follow a Gaussian distribution whose mean is  $v_{max}$ .

The variances in the distributions of the above probabilities were statistically determined by using 45 deviation data made in advance. Here, we assumed that  $b_{\theta_t}(\phi_t)$  approximates the multiplication of these probabilities, as follows:

$$b_{\theta_t}(\phi_t) \approx b_{q_t} \cdot b_{r_t} \cdot b_{v_t}$$

## 6. EXPERIMENTAL EVALUATION

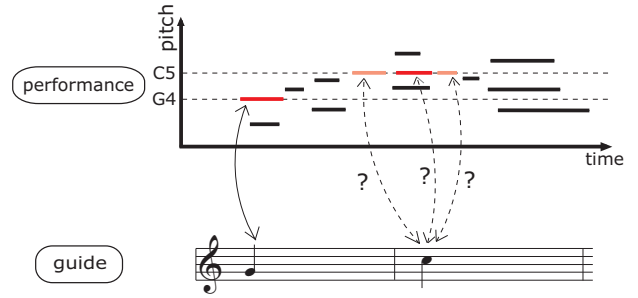
The algorithm for quantizing the start-timing has been tested on a database of ten MIDI-formatted expressive piano performances. These test samples contain both rhythmically complex pieces (e.g. ‘Troimerai’) and simple pieces (e.g. ‘Menuett G dur’).

### 6.1. Experiments of Quantizing onset timing

Let the percentage of correct answers  $Cor_{on}$  be:

$$Cor_{on} = \frac{N - n_{sub}}{N} \times 100$$

where  $N$ ,  $n_{sub}$  are defined as follows:



**Figure 4.** An example of mistakable cases in DP matching

- $N$  : the number of performance notes that exist in the answer data <sup>3</sup>
- $n_{sub}$  : the number of notes in  $N$  for which the onset timings are not equal to the answer.

The experimental results are shown in Table 1. Each number in the table represents the average recognition rates for 10 sample pieces. The results gotten by the proposed algorithm are in the ‘‘HMM’’ columns. For comparison, the results of experiments where the quantization resolution is fixed to a sixteenth note are also shown.

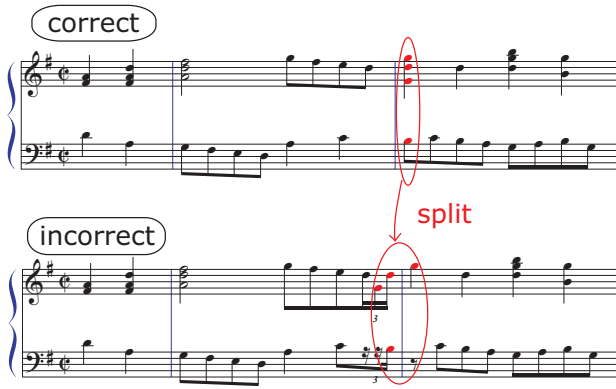
The results show that the proposed algorithm is better than a simple threshold-based quantization processing. When 1 note per bar is given as the guide, the average recognition rate reached 92.9%<sup>4</sup>. This result could be said to meet the purpose of making deviation data efficiently and precisely. The proposed method will especially be useful to those untrained in making score data, who use notation software such as MakeMusic!’s FINALE.

The errors in the quantization are classified into

- ones caused by mismatching performance notes to guide notes (see Figure 4)
- ones splitting the notes’ intended simultaneous onset timings (see Figure 5)

<sup>3</sup> The answer data don’t contain grace notes.

<sup>4</sup> Except for one piece whose recognition rate was the worst in the dataset, the average recognition rate reached 95.5%



**Figure 5.** Splitting the notes intended simultaneous onset timing

We may resolve the former errors by giving guide notes whose pitches are not identical when the same notes are repeated. The latter error can be reduced in part by adjusting the probabilities concerning the auto-regressions of the state in the HMM, to the style of the music to be analyzed (see 5.2).

## 6.2. Experiments on quantizing durations

Here, we mention briefly the tendencies of durations in musical scores:

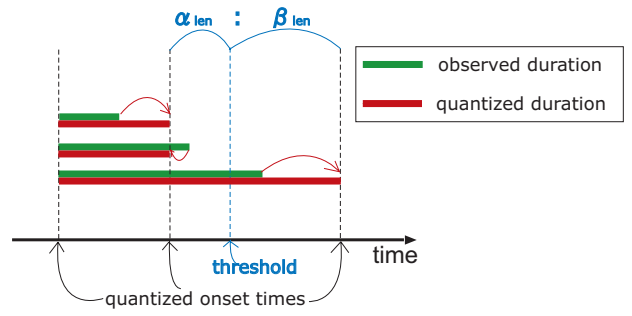
- The durations of performance notes do not always reflect the canonical note values described in scores. Granted that no expression mark is written in the scores, players can consciously or unconsciously change canonical note values into various durations in their performances.
- When the pedal is used to lengthen the note duration, the observed durations, the intervals between onset timing and offset timings in MIDI data, tend to be shorter than intended in the score.

These facts make it very hard to obtain canonical note values from the durations of performance notes.

Against such a background, we tried to quantize the durations of performance notes by which the canonical note values in the musical scores are the criterion for quantization. We used a method reflecting the following custom of musical notation:

- In musical scores, the offset timing of a note tends to be described as the onset timing of subsequent notes
- In musical scores, the note durations tend to be longer than the intervals between the onset and offset timings in MIDI data.

After quantizing the onset timing by using the HMM-based method, the durations were quantized according to the following rules (see Figure 6):



**Figure 6.** Quantizing durations

- When the offset timing of a note is earlier than its next onset timing, the offset timing is identified with the next onset timing.
- When the offset timing of a note is later than its next onset timing, the offset timing is quantized according to the threshold that divides the area whose ends are the nearest neighbour onset timings of the offset timing.

$\alpha_{len}$  and  $\beta_{len}$  ( $\alpha_{len} < \beta_{len}$ ) in Figure 6 were determined from pilot studies.

Let the recognition rate  $Cor_{on,len}$  be:

$$Cor_{on,len} = \frac{N - n'_{sub}}{N} \times 100$$

where  $N$ ,  $n'_{sub}$  are defined as follows:

- $N$ : the numbers of performance notes that exist in the answer data<sup>5</sup>
- $n'_{sub}$ : the number of notes in  $N$  for which the onset timing or durations are not equal to the answer.

The experimental results are shown in Table 2. Each number in the table represents the average recognition rates for ten sample pieces, which are the same as used in the experiment on quantizing onset timing. The results gotten by the proposed method reflecting the custom in scoring are shown in Table 2 in the “Custom” column. For comparison, the results of an experiment where the quantization resolution is fixed to a sixteenth note are also shown.

The results show that the proposed algorithm is better than a simple threshold-based quantization processing. It seems reasonable to conclude that the proposed algorithm reflects the customary form of a score, to some extent. However, the obtained recognition rates themselves are not satisfactory. The difficulties mentioned above, which may be the cause of the errors, have not been fully examined. We are going to discuss them again at the end of this section.

<sup>5</sup> The answer data don't contain grace notes.

**Table 2.** Results of quantizing durations

Guide	start and end	per 2 bars	per 1 bar	per downbeat	per chord
Threshold	28.2	41.5	44.1	61.0	68.8
Custom	41.6	57.8	67.7	75.7	84.9

**Table 3.** The amount of time required to make guide data. productivity is given by the 2nd. column value divided by the 1st. column value

	whole notes	1 note per a chord	productivity
<i>expert of Finale</i>	10min. 18sec.	2min. 16sec.	22.0%
<i>beginner of Finale</i>	1hour 9min. 49sec.	12min. 20sec.	17.7%

### 6.3. Measurement of productivity

The primary goal of the system described so far is to reduce the load when trying to obtain deviation data from MIDI-formatted music. To assess the utility of the system, we compared the times for inputting the score data by using *FINALE*.

Table 3 compares the times to input *Gavotte* composed by Chopin. The data show that we can save around 80% of the workload to input the score while obtaining a 99.8% accuracy (onset timing) deviation database. It thus seems that the proposed method is effective, especially for those who are interested in music expression but are not accustomed to using notation software.

### 6.4. Discussion

Let us summarize the points that have been made in this section:

1. The proposed method exhibits a high capability in quantizing the onset timing.
2. There is still room for improving the quantizing durations, ie., giving a note value to a note.
3. The proposed method is an effective tool for making a deviation database.

Although we might conclude that the proposed method satisfies the primary goal, some people who regard the customary form of a score as crucial to making the music database may think the second point could be a fatal shortcoming. We confirmed the effectiveness of the following heuristics to solve this problem: *Rests seldom appear at the process of phrase*. Applying this heuristic limited the quantizing duration to that of the human MIDI performances, *When a pianist presses the pedal down beyond a certain depth and if the note is a one whose position is far from other notes, the note duration is interpreted to be longer than the duration indicated in the MIDI data*.

Note values of a score do not always represent the acoustically “correct” feature. The score’s note values are expedient in transcribing music. Quantizing duration seems to be less important than quantization of onset timing from

a cognitive viewpoint. A suggestion to cope with this situation is to gather many performances and to make average deviation data from their individual deviations. The proposed method, at least, can extract the average of the performances. It is also able to segregate the canonical part from the deviations of a performance, most of which contain mis-touches (extra notes or missing notes).

## 7. DATABASE AND TOOLS

The database is to be released on the web [13].

### 7.1. Database

The sources of the deviation data were pieces recorded in SMF which were played by pianists or MIDI files which were in the public domain. The database contains about 40 pieces in “Sonatinen Album” and 20 well-known classical pieces, for example, “für Elise”, “Turkish March”, and so on. One set of deviation data was made for each of the above pieces. Additionally, we prepared several deviation data for Mozart’s “K.331” and Chopin’s “Etude Op.10, No.3” and “Walzer Op.64, No.2”, for the studies on performance rendering.

As for above pieces, not reflecting the purpose to make efficient deviation data, we also made deviation data whose durations were accurately canonicalized to the durations represented in the scores by using guide scores containing all notes of the pieces.

### 7.2. Tools

We here introduce some tools for utilizing the database.

#### 7.2.1. Sequencer *flower*

*flower* is a sequencer to play deviation data. The user is able to assign the degree of deviation in the onset timing, duration and local tempi. For example, if all degrees are 0%, the consequent performance will have no deviation. In turn, if all of the degrees are assigned 100%, *flower* reproduces the performance as it is. That is, it enables comparison between a mechanical performance and an expressive performance. Of course, any degree between 0%

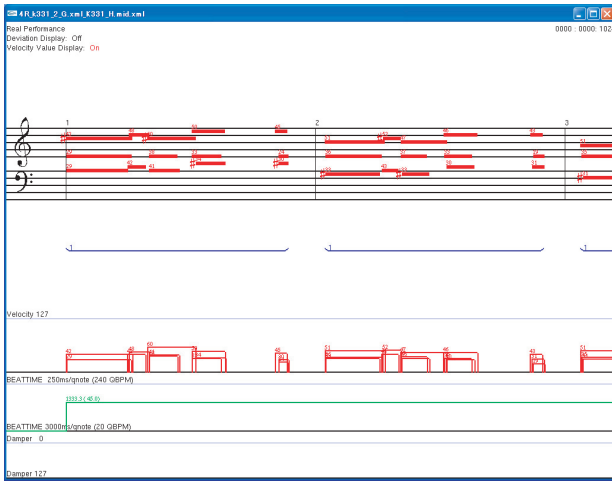


Figure 7. K.331 Henle Edition

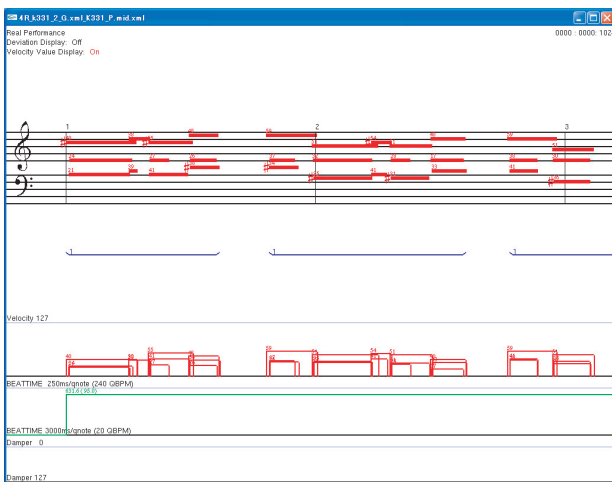


Figure 8. K.331 Peters Edition

and 100% can be assigned. Moreover, *flower* can accept the degrees more than 100% in order to emphasize deviations.

Also, *flower* has functions as follows:

- Play only the specified voice
- Play according to the default velocity

### 7.2.2. Visualizing Tool *nov*

*nov* provides a piano-roll-type representation of the deviation data. Figure 7 and 8 indicate the deviations of “K.331” Henle edition and those of “K.331” Peters edition. *nov* enables easy visual comparison of several deviation data.

### 7.2.3. Interactive Performance System *iFP*

It would be useful to exploit a deviation database for entertainment purposes rather than just research purposes.

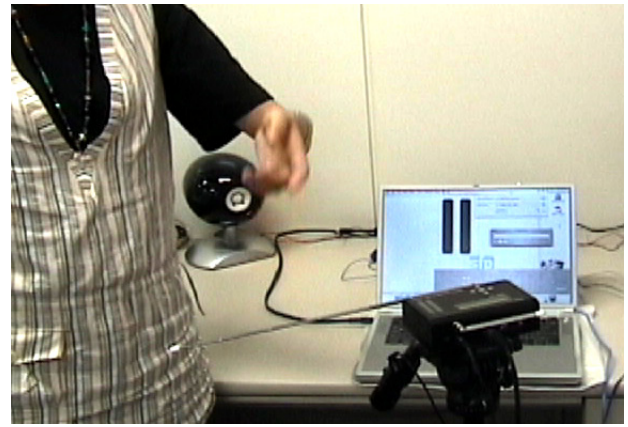


Figure 9. A Conducting Interface using capacity transducer

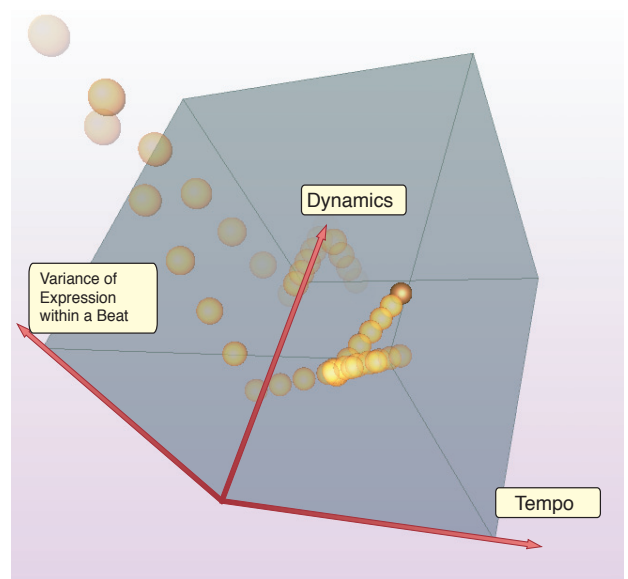


Figure 10. Example of real-time visualization of a performance. K.331 played by S. Bunin.

*iFP* [15] is an interactive performance system that enables one to play expressive music with a conducting interface as shown in Figure 9. *iFP* uses the deviation data as a performance template. By using the template as a skills-complement, a player can play music expressively over and under beat level. The scheduler of *iFP* allows the player to mix her/his own intention and the expressiveness written in the deviation data. It also provides real-time visualization (feedback) of expressiveness as shown in Figure 10.

### 7.2.4. Converter *note2xml*

Presently, we are preparing *note2xml* which converts the deviation data into MusicXML(4R) format. This format, developed by Rencon WG, is based on Recordare MusicXML and has the purposes of distributing datasets

for training and evaluating performance rendering systems [16]. MusicXML(4R) is able to preserve deviation data written in NOTE format.

## 8. CONCLUSIONS

This paper proposed a method to extract performance deviation data efficiently from MIDI piano performances, based on DP Matching and an HMM. We gave a brief overview of the database and related tools. The experimental results showed that we can efficiently construct databases of performance deviations. We are going to expand the database by using the proposed method.

Databases of performance deviations should help to foster studies in music informatics and musicology. The databases could become indispensable, especially because they can accelerate studies on performance rendering. Education and entertainment fields could also make good use of such databases.

We would like to improve the algorithm, so it can extract deviations from acoustic sounds. We also would like to construct a computational tapping model, which may give the rational position of tempo change terms **beat-time**, as a future work.

## 9. ACKNOWLEDGEMENT

The authors would like to thank Ms. Mitsuyo Hashida, and Mr. Keita Okudaira for their contributions to the study. Prof. Hiroshi Hoshina and Mr. Yoshihiro Takeuchi made valuable comments, and they were very helpful. This research was supported by PRESTO, JST, JAPAN.

## 10. REFERENCES

- [1] Goto, M., Hashiguchi, H., Nishimura, T. and Oka, R. "RWC Music Database: Music Genre Database and Musical Instrument Sound Database", *Proceedings of the 4th International Conference on Music Information Retrieval*, pp.229-230, 2003.
- [2] Bell, B. "Raga: approches conceptuelles et experimentales", *MIM-CRSM-Sud Musiques - CCSTI Marseille*, 1988.  
<http://www.lpl.univ-aix.fr/lpl/presentation/publications/docs/bel/raga.pdf>
- [3] CCARH Homepage:  
<http://www.ccarh.org/>
- [4] Widmer, G., Dixon, S., Goebel, W., Pampalk, E. and Tobudic, A. "In Research of the Horowitz Factor", *AI Magazine*, FALL 2003, pp.111-130, 2003.
- [5] Schloss, W. Andrew. *On the Automatic Transcription of Percussive Music - from Acoustical Signal to High-Level Analysis*. Ph.D. thesis, CCRMA - Stanford University, 1985.
- [6] Desain, P. and Honing, H. "The Quantization of Musical Time: A Connectionist Approach", MIT press, *Computer Music Journal*, Vol. 13, No. 3, pp. 56-66, 1989.
- [7] Katayose, H., Imai, M. and Inokuchi, S. "Sentiment Extraction in Music", *Proceedings of the International Conference on Pattern Recognition*, pp.1083-1087, 1988.
- [8] Otsuki, T., Saito, N., Nakai, M., Shimodaira, H. and Sagayama, S. "Musical Rhythm Recognition Using Hidden Markov Model", *Transaction of Information Processing Society of Japan*, Vol.43, No.2, pp. 245-255, 2002 (in Japanese).
- [9] Hamanaka, M., Goto, M., Asoh, H. and Otsu, N. "Learning-Based Quantization: Estimation of onset timing in a Musical Score", *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics (SCI2001)*, Vol.X, pp. 374-379, 2001.
- [10] Takeda, H., Nishimoto, T. and Sagayama, S. "Estimation of Tempo and Rhythm from MIDI Performance Data based on Rhythm Vocabulary HMMs", *IPSJ SIGNotes 2004-MUS-54*, pp. 51-56, 2004 (in Japanese).
- [11] Katayose, H., Fukuoka, T., Takami, K. and Inokuchi, S. "Expression extraction in virtuoso music performances", *Proceedings of the Tenth International Conference on Pattern Recognition*, pp.780-784, 1989.
- [12] NOTE format Specification:  
<http://ist.ksc.kwansei.ac.jp/~katayose/Download/Document/>
- [13] Katayose Lab. Download Page:  
<http://ist.ksc.kwansei.ac.jp/~katayose/Download/frameset.html>
- [14] Bellman, R. *Dynamic Programming*, Princeton University Press, Princeton, 1957.
- [15] Katayose, H. and Okudaira, K. "Using an Expressive Performance Template in Music Conducting Interface", *Proceeding of the New Interfaces for Musical Expression (NIME04)*, pp.124-129, 2004.
- [16] Hirata, K., Noike K. and Katayose, H. "Proposal for a Performance Data Format", *Working Notes of IJCAI-03 Workshop on methods for automatic music performance and their applications in a public rendering contest*, 2003.