The Thirteenth International Symposium on Artificial Life and Robotics 2008(AROB 13th '08),
B-Con Plaza, Beppu, Oita, Japan, January 31-February 2, 2008

Interactive musical editing system to support human errors and offer personal preferences for an automatic piano

- Method of searching for similar phrases with DP matching and inferring performance expression-

Yuichi Chigoi

Graduate School of Computer Science
and System Engineering
Kyushu Institute of Technology
680-4, Iizuka City, Fukuoka, Japan, 820-8502
hiki@mmcs.mse.kyutech.ac.jp

Eiji Hayashi

Department of Mechanical Information
Science and Technology
Kyushu Institute of Technology
680-4, Iizuka City, Fukuoka, Japan, 820-8502
haya@mse.kyutech.ac.jp

Abstract

We have developed a system that allows a piano to perform automatically. To get the piano to play music in the manner of a live pianist, we must add expression to the piano's performance. In the case of piano music, there are often 1000 or more notes in the score of even a short piece of music, and the automatic editor must spend a huge amount of time to accurately simulate the emotionally expressive performance of a highly skilled pianist. To shorten this editorial time, we have developed an interactive musical editing system to edit music more efficiently by utilizing a database.

We have analyzed MIDI data regarding the performances of highly skilled pianists in order to observe the stylistic tendencies of their performances. We found that phrases having similar patterns in the same composition were performed in similar styles.

Therefore, we developed a system that searches for similar phrases throughout the musical score and infers the style of their performance. Here, we propose a method using DP (Dynamic Programming) matching as a way to search for similar phrases and a method for inferring performance expression.

Key words: *automatic piano, knowledge database, computer music, DP matching*

1. Introduction

We have developed a performance system for an automatic piano. In this system, 90 actuators are installed in the 88 keys and 2 pedals of a grand piano. Those actuators operate key strokes and pedaling to be executed on the piano. (See Figure 1)

Reproducing music with the piano is similar in some ways to reproducing music on a computer. Essentially, variations in tempo, dynamics, and on the like are needed to arrange the respective tones in the desired way. However, in the case of piano music, there are 1000 or more notes in the score of even a short piece of music, and the automatic editor must spend an enormous amount of time working with the arrangement in order to simulate the expressions of an actual performance. To shorten this editorial time, we have developed an interactive musical editing system to edit music more efficiently [1].

We have analyzed MIDI data regarding the performances of highly skilled pianists in order to observe the stylistic tendencies of their performances. We found that phrases having similar patterns in the same composition were performed in similar styles. Moreover, we found that the pattern of notes in the score sometimes influences the way the pianist expresses the music.

We have developed a system that searches for similar phrases throughout a musical score and infers the style of the performance. We propose a method using DP matching as a way to search for similar phrases. This system converts notes into character strings, and it runs DP matching using character strings and calculates the degree of disagreement between these strings. We use these calculations as an index to determine whether the strings resemble each other. Moreover, we designed a method for inferring the performance expression of similar phrases found by DP matching.

In this paper, we describe the results of searching for similar phrases using DP matching and inferring performance expression.

Figure 1: View of the automatic piano

2. Musical Editing Support System

2.1 System Architecture

The structure of the system is shown in Figure 2. The user edits music via the user's interface on the computer display. The user can also access a database that has musical grammar, the user's preferences, and related topics. As a result, editorial work is reduced and efficient editing becomes possible.

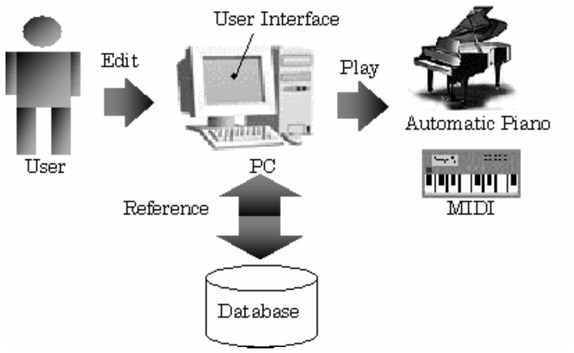


Figure 2.1 Structure of the editing system

2.2 Format of Performance Information

The parameters of performance information are shown in Tables 1 and 2.

The automatic piano that we have developed uses a music data structure that is similar to MIDI. We defined performance information, dividing it into two categories, the notes and the pedals.

The note information is comprised of the six parameters involved in producing a tone: "Key (note)", "Velo (velocity)", "Gate", "Step", "Bar", and "Time". "Velo" is the dynamics, given by the value of 1～127. "Gate" is the duration of the note in milliseconds. "Step" is the interval of time until the next note, and it also exhibits tempo. "Bar" is the vertical line placed on the staff to divide the music into measures.

The pedal information is comprised of four parameters: "Key (indicating the kind of pedal, "Damper" or "Shifting")", "Velo (the pedaling quantity)", "Time (the duration for which the pedal is applied)", and "Bar".

2.3 Editing Support Process with Databases

Our system can automatically apply a rough version of performance expression using a Musical Rules Database and Score Database. (See Figure 2.2)

In addition, this system has a Preference Database for storing away the editing characteristic of the user.

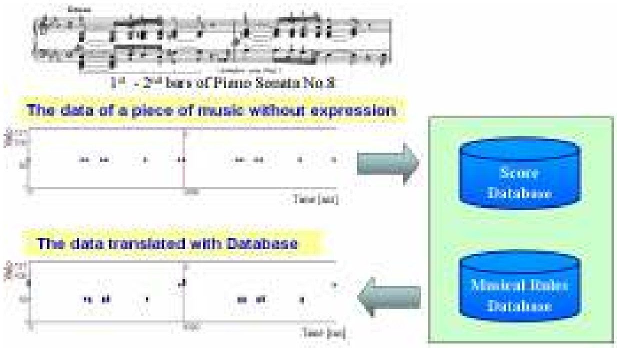


Figure 2.2: Automatic translation with database

2.3.1 Musical Rules Database

The Musical Rules Database contains the architecture of musical grammar necessary to interpret symbols in musical notation. This database is composed of five tables containing dynamics marks, articulation marks, symbols of changing dynamics or changing tempo (symbols that affect the speed of a note or the increase or decrease of the volume), time signature, and tempo marks.

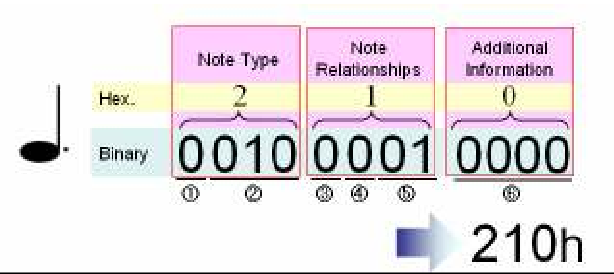
Analyzing a music symbol according to its usage allows efficient information processing by the system.

2.3.2 Score Database

The Score Database has symbols including notes, time signatures, rests, and the like in standard musical notation. The symbols are pulled together in the order in which they occur within the bars, and the bar symbols are arranged in a time series. Performance expression in itself is only information such as pitch and strength, the length and is over only in the enumeration of the sound. Because the identification of each sound is difficult, editing of the performance expression is difficult. By adding information from the Score Database to performance expression, we can connect each note to the note(s) before and after it. In doing so it, it is easy to edit each phrase.

This database consists of three tables, the element table (showing the position of the note and the composition of the chord), the symbol table (showing the position of the music symbol), and the "same" table (showing the position of the repetition of the phrase).

The element table contains the field "Note Value". Data in this field indicates the type of note, e.g., a quarter note, a triplet, and so on. "Note Value" is expressed by three hexadecimal numbers. The numbers of "Note Value" are shown in Figure 4.



①	Note or Rest (0: Note, 1: Rest)
②	Note Value (000: A whole note, 001: A half note …etc)
③	Tie (If the note has a tie then this number is 1.)
④	Ornament (If the note has an ornament then this number is 1.)
⑤	The number of dots (Exceptionally, if it is "11" then the note is tuplet.)
⑥	Additional Information The number of tuplets. (Triplets: 0011) The type of an ornament. (Trill: 0101) etc…

Figure 4: Note Value

3 Searching for Similar Phrases

Through analysis, we found that phrases of the same pattern existing in the same tune are performed with similar expression. We used DP matching to search for similar phrases within a musical score.

3.1 DP Matching

DP matching is a technique that works well in the fields of speech recognition, bioinformatics, and other language fields. It has a feature that can calculate similarity between two words that are different in a number of characters from each other.

In Figure 5, the route of minimum cost in each point is taken and the route with the lowest cost is assumed to be an optimal path finally. The cost at that time is defined as the distance between patterns. In this system, this distance is handled as a threshold to judge whether the phrases are similar to each other. For example, if the pointer moves up or to the right then the cost is increased by 1. If it moves to the upper right then the cost is not increased. And, if the characters do not correspond in each point then the cost is increased by 5.

$$g(i,j)=\min\begin{Bmatrix} g(i-1,j) \\ g(i,j) \\ g(i,j-1) \end{Bmatrix}+d(i,j)$$

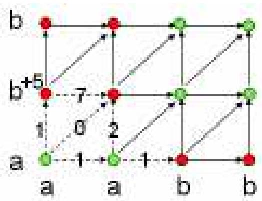


Figure3.1 DP matching

3.2 Searching with DP Matching

To search with DP matching, we had to convert a musical score into character strings (with the series of strings known as a Note Pattern) before searching for similar phrases. This process is explained below.

3.2.1 Note Pattern

Our system converts a score into a Note Pattern using Note Value (see Figure 4) to perform DP matching. Of the three columns of Note Value, we used the two leftmost columns. The system replaces the second column with letters of the alphabet (from G) because a letter should not be the same as the first column with the second column since it expresses one note with two columns. An example of a Note Pattern conversion is shown in Figure 3.2.

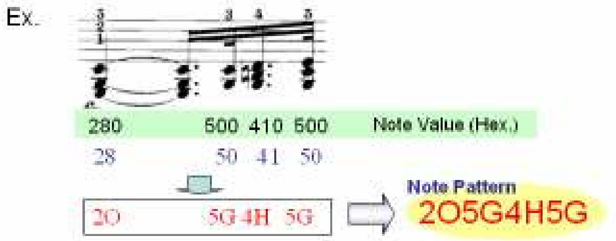


Figure 3.2 Example of a Note

3.2.2 The Method of Searching

The flow of the similar phrase search is shown in Figure 7.

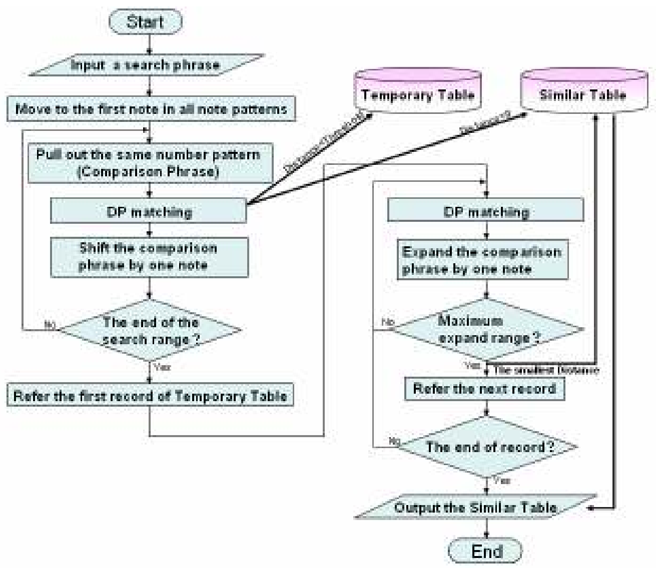


Figure 3.3 The flow of the similar phrase search
(Search range and expand range are set beforehand.)

Essentially two rounds of search processing are performed. The first processing round narrows down the point that may be the point of resemblance in all search ranges. A pattern of the same number as the search phrase is pulled out, and the distance between two patterns is calculated using DP matching. If they are in complete accord (distance = 0), then the phrase is stored in the "same" table. If the distance is lower than the threshold then the phrase is stored in the temporary table. In the second round of processing, DP matching is performed again using the phrases in the temporary table while increasing the number of characters. In other words, the system looks for the most similar phrases in the surrounding phrase. The threshold of this system has been decided by trial and error.

4 Inferring Phrase Expression System

The performance expression of similar phrases identified by the method described in Section 3 is inferred using the expression of the performance of the search phrase.

4.1 Method of inferring phrase expression

The correspondence of the note patterns is revealed when the distance is minimized in DP matching. Next, the ratio of Velo to the previous sound is calculated in each sound of the searching phrase. Velo of the associated phrase is inferred using a ratio, R_i, as follows:

$$R_i = \frac{V_{i+1}}{V_i}$$

V is the Velo of each search phrase. W, which is the Velo of a similar phrase, is calculated in R as follows:

$$W_{i+1} = W_i \times R_i$$

Thus, the Velo of a similar phrase is inferred. The value of Time uses the value decided by Automatic translation (see Section 2.3).

4.2 Searching & Inference Result

The result of the searching and inference using this inference system is shown in Figure 4. The phrase of the seventh bar of Beethoven's Sonata No. 8 was inferred from the first bar, which Gerhard Oppitz performed. The figure shows the reasoning result of the phrase that has distance = 0 , 20 from the search phrase. The inferred phrase was similar to the expression of the performance of the search phrase.

In addition, even the phrase which distance leaves, we understand that the reasoning that resembled closely to some extent is made.

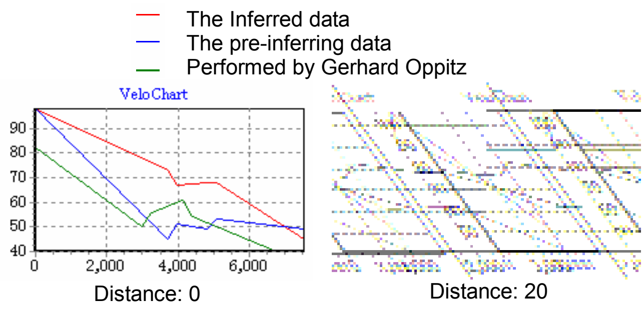


Figure 4 Searching & Inference Result

5 Conclusion

We designed a method of searching for similar phrases and inferring the performance expression using DP matching and combined these into one system.

In the similar phrase search, the system was able to search for similar phrases using DP matching in a short time, and it was even possible to search for phrases whose resemblance might not be immediately apparent.

In the inference of the performance expression, the system was able to infer the performance by using the best association based on DP matching.

The performance expression of some similar phrases that exist in the tune can be inferred at the same time using this system. The work efficiency increases by this automatic editing in contrast to editing the piece by hand from the beginning, without sacrificing quality.

In this study, We were able to perform only a similar phrase search and the reasoning of the performance expression in the same music. We will perform the reasoning with different music and will evaluate the existing system in the future.

References

[1] Hayashi, E. et al, "Behavior of piano-action in a grand piano, I", Journal of Acoustical Society of America, Vol. 105, pp. 3534-3544, 1999.

[2] Hayashi, E et al, "Interactive musical editing system for supporting human errors and offering personal preferences for an automatic piano", Proc. of the 7th International Symposium on Artificial Life and Robotics, Vol. 2, pp. 513-516, 2002.

[3] Hikisaka, Y., Hayashi, E., et al, "Interactive musical editing system for supporting human error and offering personal preferences for an automatic piano –Method of searching for similar phrases with DP matching and inferring performance expression-", Proc. of the 12th International Symposium on Artificial Life and Robotics, GS4-3, 2007.