The Sixteenth International Symposium on Artificial Life and Robotics 2011 (AROB 16th '11),
B-Con Plaza, Beppu,Oita, Japan, January 27-29, 2011

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

- A method for searching for similar phrases using DP matching –

Kenichi Koga and Kentaro Minowa and Eiji Hayashi

Department of Mechanical Information Science and Technology
Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology
680-4, Kawazu, Iizuka-City, Fukuoka Prefecture, Japan

Abstract: We have developed a system that allows a piano to perform automatically. In order 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, requiring that an editor spend a huge amount of time to accurately simulate the emotionally expressive performance of a highly skilled pianist. Therefore, we have developed an interactive musical editing system that utilizes a database to edit music more efficiently. We have analyzed MIDI data regarding the performances of highly skilled pianists in order to observe the stylistic tendencies of their performances. Our result showed 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 a musical score and evaluates the style of their performance. The method of searching for similar phrases uses DP (Dynamic Programming) matching. Using the method of searching, we developed a phrase and music search engine. We thought that as long as it is able to search for similar phrases, it would be able to search for a tune including the phrase from among various tunes.

Keywords: automatic piano, knowledge database, computer music, DP matching

I. INTRODUCTION

We have developed a performance system for an automatic piano. In this system, 90 actuators are installed on the 88 keys and the 2 pedals of a grand piano. These actuators operate key strokes and execute pedaling on the piano. (See Figure 1.1)

Reproducing music with the piano is similar in some ways to reproducing music on the computer. Essentially, variations in tempo, dynamics, and so on 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 a score of even a short piece of music, and for this reason an editor must spend an enormous amount of time working with an arrangement in order to simulate the expressions of an actual performance. Therefore, in this research, we have developed an interactive musical editing system to edit music more efficiently [1].

We have analyzed MIDI data from the performances of highly skilled pianists in order to observe the stylistic tendencies of their performances. Our results showed that phrases having similar patterns in the same composition were performed in similar styles. Moreover, we found that the pattern of notes in a score sometimes influences the expression of a piece of music.

In this research we developed a system that searches for similar phrases throughout a musical score and evaluates the style of their performance. We propose a method that uses DP matching as a way to search for similar phrases. This system converts notes into character strings. In addition, the system 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. At the end, we introduce a phrase and music search engine using the method of searching.

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

Figure 1.1: View of the automatic piano

II. Musical Editing Support System

2.1 System Architecture

The structure of the system is shown in Figure 2.1. The user edits music via the user's interface on a computer display. The user can also access a database that has musical grammar, the user's preferences, and so on. As a result, editorial work is reduced and efficient editing becomes possible.

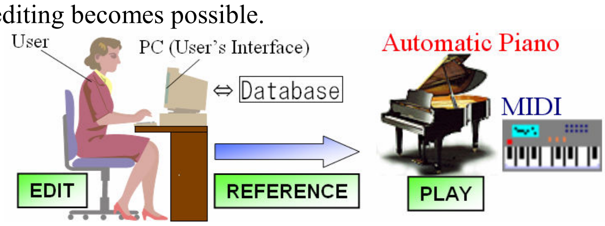


Figure 2.1: Structure of the editing system

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 between notes, 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 Database

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

In addition, the system has Preference Database, which stores the editing characteristic of the user.

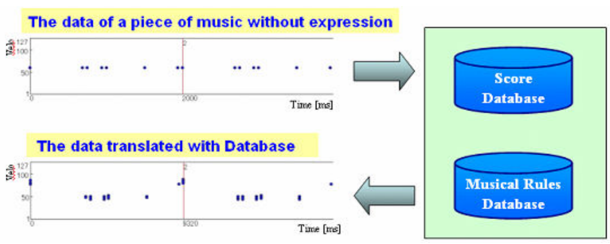


Figure 2.2: Automatic translation with database

2.3.1 Musical Rules Database

This database contains the architecture of musical grammar necessary to interpret symbols in musical notation. It is composed of five tables containing "Dynamics marks", "Articulation marks", "Symbol of Changing Dynamics or Changing Tempo" (symbol that affects the speed of a note or the increase or decrease of the volume), "Time signature", and "Tempo marks".

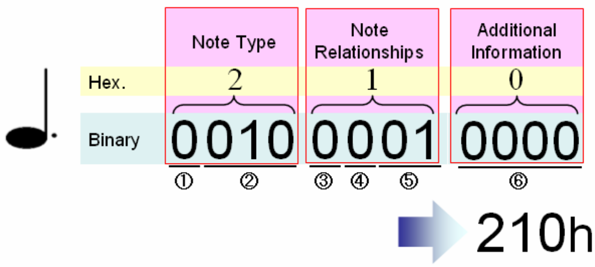
Analyzing a music symbol according to its usage allows efficient information processing by the system.

2.3.2 Score Database

This database has symbols including time signatures, notes, rests and so on in standard musical notation. Symbols were pulled together in order of bars, and bar symbols were arranged in a time series. Performance expression in itself is only information such as pitch, strength, and length and concerns only the enumeration of a sound. Because the identification of each sound is difficult, editing of the performance expression is difficult. By adding the Score Database's information to performance expression, we can connect each note to its enumeration. In doing so, it becomes 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, which are shown in Figure 2.3.



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:010) etc…

Figure 2.3: Note Value

III. Searching for Similar Phrases

As a result of the analysis, it was found that phrases of the same pattern existing in the same tune are performed in a similar expression. This time, we used DP matching to search for similar phrases.

3.1 DP matching

DP matching is a technique used widely in the field of speech recognition, bioinformatics and so on. It has a feature that can calculate the similarity between two words that are different in a number of characters from each other.

In Figure 3.1, the route of minimum cost in each point is taken, and the route with the lowest cost is assumed finally to be the optimal path. 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 cost moves up or to the right, then it is increased by 1. If it moves to the upper right, then it does not increase. Also, if the characters do not correspond in each point, then the cost is increased by 5.

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

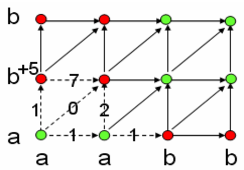


Figure 3.1: DP matching

3.2 Searching with DP matching

In this passage, we describe a method of searching with DP matching. We had to convert a musical score into character strings (a Note Pattern) before searching for similar phrases. This process is explained below.

3.2.1 Note Pattern

Our system converted a score into a Note Pattern using Note Values (See Passage 2.3) in order to perform DP matching. Of the three columns of Note Values, we used the two columns on the left. The system replaces numbers in the second column with letters of the alphabet (from G) because a letter, being a different notation than that used in the first column, allows the expression of one note with two columns. An example of a Note Pattern conversion is shown in Figure 3.2.

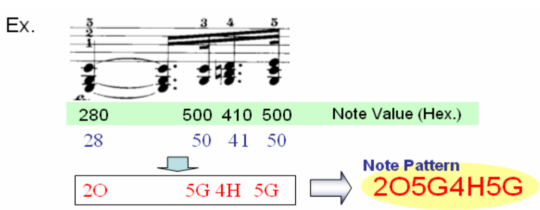


Figure 3.2: Example of Note Pattern

3.2.2 The Method of Searching

The flow of the similar phrase search is shown in Figure 3.3.

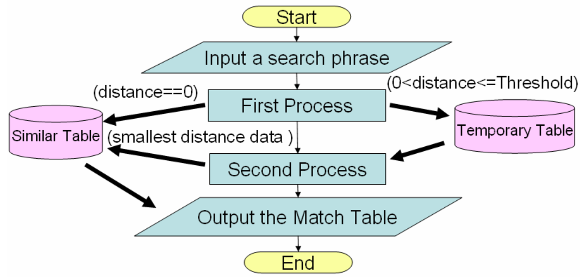


Figure 3.3 The flow of the similar phrase search

Essentially, three rounds of search processing are performed. The first processing round narrows down the points to those having a resemblance in all search ranges. A pattern with the same number as the search phrase is pulled out, and the distance between the two patterns is calculated using DP matching. If they are in complete accord (distance = 0), then the phrase is stored in the Similar Table. If the distance is lower than the threshold, then the phrase is stored in the Temporary Table. (See Figure 3.4)

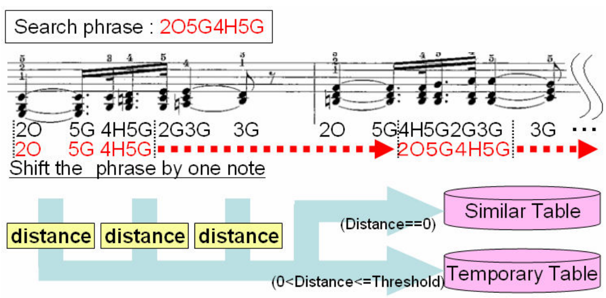


Figure 3.4: The first process

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 phrases. The threshold of this system is decided by trial and error. (See Figure 3.5)

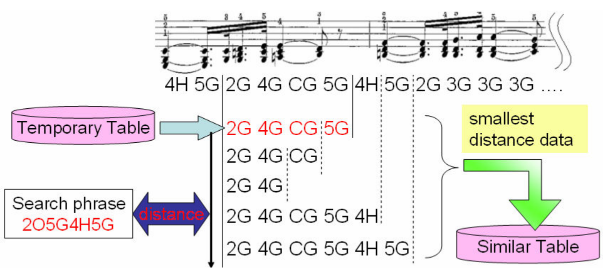


Figure 3.5: The second process

VI. Phrase and music search engine

We can use this system to search for similar phrases in a piece of music. Thereupon, we thought that as long as it is able to search for similar phrases, it would be able to search for a tune including the phrase from among various tunes.

Therefore, we developed a phrase and music search engine. It searches a tune from various tunes with one phrase of voice data. The voice is first inputted; then the interval between the sounds of the inputted voice data is converted back to notes according to the tempo of music. As a result, we thought would be able to search for sequence similarity to the phrase rhythm and notes. The search strategy uses 3.2.2 The Method of Searching.

4.1 Processing that inputs voice

It is shown that voice is input in the shape of waves as shown in Fig. 1. The vertical axis of Fig. 4.1 shows loudness (mdB), and the horizontal axis shows time. By setting a threshold on the vertical axis, the presence of voice can be recognized. The interval of this sound is measured as the time. In other words, as shown in Fig. 1, the interval of sound is the period between the initial iterative peak and the next time the sound iterative peak. The shapes of the waves tell us the note values (See section 2.3) of the phrase.

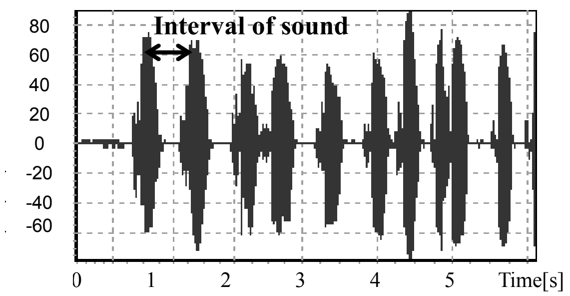


Figure 4.1: Result of sound input

4.2 Search Experiments and Discussion

We searched for music using the phrase and music search engine. The search phrase was the phrase from fourth sound of bar 24 to first sound of bar 27 of "asita ga arusa" ULFULS. Results were set to output phrases with a distance of 20 or less. Table 4.1 summarizes the results. Looking at the results, "asita ga arusa" has the smallest average distance. Thus, the system can find "asita ga arusa" to select smallest average distance sound.

Table 4.1: Search results

Name of sound	Number of searches for phrases	The average distance between the phrase
Sekai ni hitotu dake no hana	46	17.20
Asita ga arusa	128	16.98
Agehatyou	9	18.33
Asuhe no tobira	26	18.42
Can you keep a secret	135	17.12
Haruka	83	17.01
Pieces of a dream	47	17.91
Tenntai kannsoku	13	19.62
sakura	1	20.00

V. CONCLUSION

We designed methods of searching for similar phrases using DP matching and combined these functions into a single system.

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

In the phrase and music search engine, we develop a system to search for tune by inputting the data of a voice performing the tune. The interval between the sounds of the inputted voice data is converted it into a note according to the tempo of music. By this, we can search for sequence similarity to the phrase rhythm and notes.

In this study, we were able to perform similar phrase searches and searching for tune by voice. In our future research we will perform evaluations with different pieces of music and will evaluate the existing system. In addition, we will develop the voice-input search system to perform a search for changes in pitch.

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.