Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2007, Article ID 42498, 9 pages
doi:10.1155/2007/42498

Research Article

A Transcription System from MusicXML Format to Braille Music Notation

D. Goto,[1] T. Gotoh,[1] R. Minamikawa-Tachino,[2] and N. Tamura[1]

[1] *Graduate School of Environment and Information Science, Yokohama National University, 79-7 Tokiwadai, Hodogaya-ku, Yokohama-shi, Kanagawa 240-8501, Japan*
[2] *The Tokyo Metropolitan Institute of Medical Science, 3-18-22 Honkomagome, Bunkyo-ku, Tokyo 113-8613, Japan*

Received 1 December 2005; Revised 25 September 2006; Accepted 29 September 2006

Recommended by Ichiro Fujinaga

The Internet enables us to freely access music as recorded sound and even music scores. For the visually impaired, music scores must be transcribed from computer-based musical formats to Braille music notation. This paper proposes a transcription system from the MusicXML format to Braille music notation using a structural model of Braille music notation. The resultant Braille scores inspected by volunteer transcribers are up to the international standard. Using this simple and efficient transcription system, it should be possible to provide Braille music scores via the Internet to the visually impaired.

1. INTRODUCTION

The digitization of music is changing our relationship with music. The technologies bring computer-based music composition, computer-based recording studio environments, and more personalized listening environments. Moreover, the Internet enables us to freely access music as recorded sounds and even music scores. The music is saved in various computer-based musical formats.

However, visually impaired people generally learn music using Braille music notation that has signs and notational conventions quite independent of the staff notation. The computer-based musical formats are incompatible with Braille music notation. This incompatibility causes a gap between sighted and visually impaired people to access music scores via the Internet and becomes a barrier to universal access. Programs to transcribe into Braille music notation have been proposed before using computer-based musical formats such as MIDI (Musical Instrument Digital Interface) [1], NIFF (Notation Interchange File Format) [2, 3], and MusicXML (eXtensible Markup Language) released by Musical Plan Ltd. (Japan) (hereinafter abbreviated as MusicXML@Musical Plan) [4].

In 1984, an automatic Braille transcription system using staff notation data for CTS (Computerized Typesetting System) was proposed [5]. This was not in general use because the CTS itself for printing music was not widely available. There remained considerable works to extract the logical structure of music from symbolic music data using the formats of MIDI, NIFF, and MusicXML@Musical Plan [6].

Keeping these pioneering works in mind, we propose here a transcription system based on MusicXML released by Recordare LLC (USA) [7] to produce Braille music scores. The two MusicXML formats are incompatible. The MusicXML@Recordare format is designed to logically represent music as a structured document. Our transcription system consists of three parts of syntactic analysis of MusicXML data, structure transformation from MusicXML format to Braille music notation, and production of Braille music scores.

Dancing Dots, in cooperation with Duxbury Systems as manufacturers of the Duxbury Braille translator, for Windows$^{(R)}$, has developed a Braille music translator GOODFEEL [8]. The GOODFEEL translator ver. 3 supports the MusicXML@Recordare format with the collaboration of a notation program, Lime 8 [9]. In addition, DaCapo is an ongoing project to develop a software package that will allow translating ink-printed notes to Braille notes and vice versa [10].

Section 2 is an outline of Braille music notation. Various computer-based musical formats are outlined in Section 3, with an emphasis on the MusicXML@Recordare format.

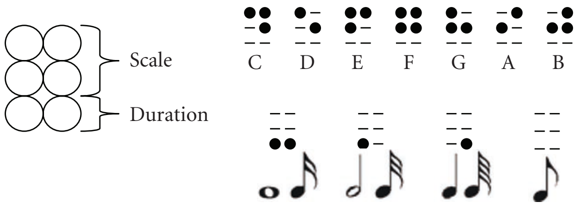


Figure 1: Note notation.

Table 1: Octave signs for pitch.

Pitch	Braille notation	Pitch	Braille notation
C″-B″	⠈⠈	c″-b″	⠨
C′-B′	⠈	c‴-b‴	⠰
C-B	⠘	c⁗-b⁗	⠠
c-b	⠸	c′′′′′-b′′′′′	⠠⠠
c′-b′	⠐		

Structure modeling of Braille music notation, categorization of structure transformation, and the system configuration are described in Section 4. A feasibility study is presented in Section 5, and Section 6 is a summary.

2. BRAILLE MUSIC NOTATION

Braille music notation using the standard Braille cell is used by visually impaired people to read and write music. Braille music scores are produced based on Braille music notation using the same Braille cells as used in their reading and writing sentences. Braille music notation provides, as it was, a language system with standard Braille cells.

The standard Braille cell consists of six embossed dots, three high and two wide. By selecting one or more dots in a cell, 64 different arrangements are possible. There are, however, more than 64 items of notes and music symbols in the staff notation. A cell can be assigned to more than one item. Also, combinations of two or more cells are used for items.

2.1. Note notation

A cell is assigned to a note which represents the relative duration and scale of sound. The cell is divided into groups of dots as shown in Figure 1. Combinations of the four upper dots in the cell are assigned to each note of the scale in staff notation. The absence or presence of the lower two dots indicates the note value as the relative duration. The possible combinations of the lower dots are 4 as shown in Figure 1, representing the whole, half, quarter, and eighth value. The sixteenth has the same combination as the whole, the thirty-second the same as the half, and the sixty-fourth the same as the quarter.

The pitch is specified by octave signs as shown in Table 1. The octave sign is placed before the cell assigned to the note and its note value. The octave sign can be removed unless information is lost and ambiguity is caused.

2.2. Music notation

Both the key signature and the time signature are placed at the beginning under Braille music notation as shown in Figure 2. Afterward, the notes and various Braille signs are arranged in the order corresponding to those of the music score with staff notation. A crescendo starting sign, an octave sign, a dotted note, a tie sign, and others are placed in the sample case as shown in Figure 2. Then space or carriage return code is placed instead of a bar symbol between measures.

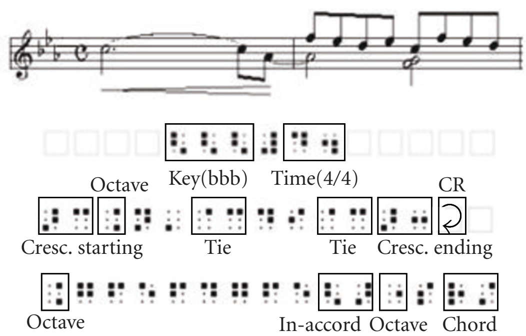


Figure 2: A sample of a Braille music score.

Table 2 is a sample of other Braille signs represented by combinations of dots in one or more Braille cells. There are also signs for accidental, dynamics, tie, and so on. The start signs for music symbols such as crescendo and decrescendo are placed before the affected notes to make those symbols execute because the symbols require time to take effect. The end signs are placed after the notes to make those symbols terminate.

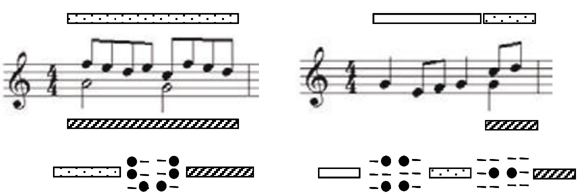
Other signs are generally placed before the affected notes. For dotted notes, the sign is placed immediately after the note.

A chord contains two or more notes of equal value. Therefore, only one note indicates the root of the chord and the other notes are indicated by their interval from the root. In cases of the treble clef appears in music scores, the highest note is indicated, and the other notes are expressed by descending intervals. However, the bass clef appears, the lowest note is indicated and the other notes are expressed by ascending intervals [11]. An arpeggio can be said to be a particular chord in which the notes are consecutively played. The broken chord with individual note values is sounded one after the other. It is applied to representations of two or more parallel voices to sound simultaneously.

When these parallel voices appear in a measure, in-accord and/or measure-division signs are placed among the voices

Table 2: Some of Braille signs for music symbols.

Braille notation	Music symbol
⠼⠙⠲	C Time(4/4)
⠖	tr Trill
⠨⠦	> Accent
⠣	♭ Flat
⠜⠉ / ⠜⠒	Crescendo
⠜⠍⠋	mf Mezzo forte
⠣⠶ / ⠣⠆	Repeat mark
⠣⠅	Double bar



(a) Parallel voices in the entire measure

(b) Parallel voices in the part measure

Figure 3: Braille representation of parallel voices. Each bar (, ,) denotes Braille representation corresponding to a voice in the staff notation. In (a), each voice is consecutively connected by the full-measure in-accord sign (⠣⠜). In (b), parallel voices are specified by the measure-division sign (⠨⠅), and then each voice is connected by the part-measure in-accord sign (⠐⠂).

to consecutively connect each other's voices in Braille music scores [11]. Figure 3 shows conceptually representations of parallel voices in Braille music scores.

2.3. Variations in Braille music scores

Braille music scores are generally produced by volunteer transcribers. Transcribers first have to learn Braille music notation and have to transcribe every detail of the music score according to the rules and syntax of Braille music notation. The transcribed Braille music scores have to be checked by other transcribers. Producing a Braille music score is therefore an extremely time-consuming task, because it is not easy for even an experienced transcriber to produce a Braille music score without errors. As a result, there are not enough Braille music scores to meet the needs of visually impaired people.

Visually impaired musicians must memorize Braille music scores before playing, therefore, there are ways to facilitate the task of remembering, such as using an abbreviation of octave signs or using Braille repeat signs. The beginners, however, may prefer simple Braille music scores that correspond clearly to the course of the music. This also means that various Braille music scores may be produced from one music score.

In addition, there appears to be a variety of minor differences in Braille music notation, depending on where in the world it is transcribed. Some countries have a different sign for interval or staff notation and use different signs for less-common music notations.

Table 3: Computer-based musical formats.

Format	Contents	Remarks
WAVE	A signal-oriented format as digital audio waveform	Music notation independent format
MIDI	A signal-oriented format for communication of music information between electronic musical instruments	Music notation independent format
NIFF	A graphic-oriented format for interchange between editing and scanning programs for music scores	Weak musicological information format
MusicXML@ Musical Plan Ltd.	An XML-based music notation format with geometrical information on music sheet	Weak musicological information format
MusicXML@ Recordare LLC	An XML-based music notation format with musicological information	Musicological information format

@ is used to mean "released by."

3. MUSIC NOTATION FOR INTERCHANGE

3.1. Notation for musical computing

Music is perceived through various media, such as sound for listening, music scores for reading, and Braille music scores for touching and reading. The media are categorized into three types to express the music with signals, graphs, and language. Moreover, modern computer-based environments convey the music saved in various computer-based formats which conform to the three types.

As shown in Table 3, the computer-based musical formats can be roughly categorized into WAVE as signal type, MIDI as intermediate between signal type and language type, NIFF as graph type, MusicXML@Musical Plan as graph type, and MusicXML@Recordare as language type. Note that transcription from the MusicXML@Recordare format to Braille music notation is between language types.

The WAVE format is a signal-oriented format for storing digital audio data. It supports a variety of bit resolutions, sample rates, and channels of audio. This format is very popular on PC platforms, and is widely used in professional programs that process digital audio waveforms.

The MIDI format is a signal-oriented format based on a sequence of numbers assigned to command to control musical hardware or software for electronic musical instruments, which are generally instructions to generate sound. It does not represent repeat, slur, or other music notations. In transcription using the MIDI format, it is difficult to decide how a sequence of numbers in the MIDI data is exactly assigned to Braille signs [1].

The NIFF format is a graphic-oriented format to interchange music data between editing programs and scanning programs which have an interactive graphical environment for editing symbolic music data extracted from scanned music scores. NIFF transcription to Braille music scores showed that the NIFF format did not lend itself to the logical structure of music [2].

MusicXML technology has been developed using XML-based music language and enables us easily to share and interchange music scores via the Internet. The MusicXML formats have been independently created by Musical Plan Ltd. [6] and Recordare LLC [7] and are incompatible. The MusicXML@Musical Plan is designed to describe the staff notation graphically using the relative positions of notes and music symbols on music sheets. We developed a transcription system using the MusicXML@Musical Plan format [4]. This system showed that the logical structure of music could not be always extracted from music data with the MusicXML@Musical Plan format.

The MusicXML@Recordare format is designed to represent music data as a structured document with musicological information so that people can print downloaded music data and play it back in its proprietary player.

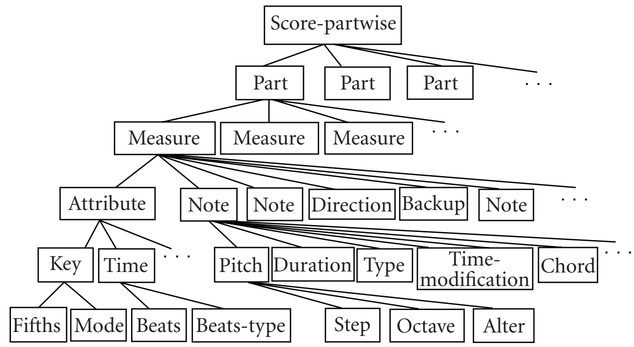


Figure 4: A sample tree structure of elements with the MusicXML format.

Table 4: Frequently used elements.

Element
<part id ="x"></part>
<measure number="y"></measure>
<attribute></attribute>
<key></key>
<time></time>
<note></note>
<chord/>
<pitch></pitch>
<step></step>
<octave></octave>
<rest/>
<duration></duration>
<backup></backup>
<direction></direction>

3.2. Structure of MusicXML format

Hereafter, we will use MusicXML to mean MusicXML@Recordare. The basis of MusicXML is XML, which is a general-purpose markup language to share data across various systems connected to the Internet. XML provides a text-based way to apply a tree structure to information and to describe the information using a hierarchy of elements and their attributes. There are two styles of structuring these elements with the MusicXML format, which are measures within parts and parts within measures.

A tree structure of measures within parts is shown in Figure 4. As the root is the score-partwise element here, the part element is the primary bifurcation.

Part elements are broken down into measure elements and their attribute is the number as time sequence. Measure elements contain nested elements such as attribute, note, and backup elements.

The attribute element contains a key element and a time element. The note element contains elements such as pitch, duration, and type elements. Moreover, these elements contain elements such as fifths, mode, and step elements, depending on the content of those elements. Table 4 shows elements that are frequently used in MusicXML documents.

For parts within measures, the root is a score-timewise element. A measure element is primary and part elements are nested within each measure element. The two styles of structuring elements are converted back and forth using XSLT (eXtensible Stylesheet Language Transformations) style sheets which provide definitions of XML tree transformations.

A chord is represented using a note element as the root of the chord and note elements of the other notes. Each of the other note elements is followed by a chord element in the lower level. For a broken chord such as an arpeggio, one or more nonsimultaneous voices are written consecutively. However, in order to represent these voices as parallel voices, a backup element makes the time counter move backward.

A time-modification element is used to handle a tuplet. If a note is contained within the tuplet, the note element has the time-modification element in the lower level, and then the number of note elements of the tuplet is specified in the lower level of the time-modification element.

4. A TRANSCRIPTION SYSTEM TO BRAILLE MUSIC NOTATION

4.1. Overview

The proposed transcription system is targeted for the tree structure of measures within parts. The structure of parts within measures is applied to the transcription system after converting to the structure of measures within parts using the XSLT style sheet.

Figure 5 shows the configuration of the transcription system. MusicXML data entered to XML processors is analyzed syntactically. The XML processors are software modules that read the XML document to find out the structure and the content so that the processors may provide application programs with the information they require. As noted earlier, XSLT is an XML processor to convert between styles. DOM (document object model) is also a processor to turn the XML document into a tree structure of elements with the XSLT style sheet.

The structure of the entered MusicXML data can be transformed into a structure of elements in Braille music notation using XSLT and DOM together. Moreover, each element of the transformed structure is transcribed into a Braille music score with the Base format, which is one of the standard Braille codes, using a dictionary for correspondence between elements of the MusicXML format and Braille music notation.

As described in Section 2.3, various Braille music scores may be produced from one music score. The purpose of the proposed transcription system is to produce simple Braille music scores for the beginners.

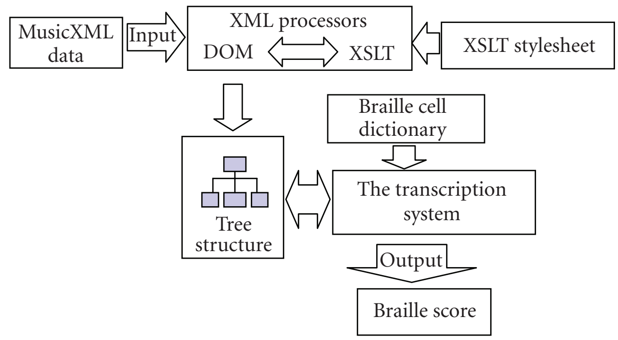


Figure 5: The configuration of the transcription system.

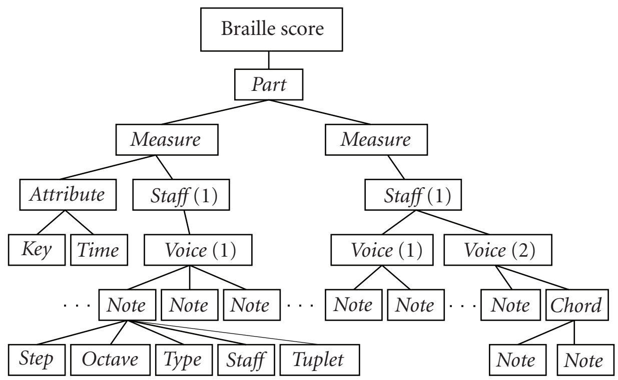


Figure 6: A sample tree structure of elements in Braille music notation.

4.2. Structure modeling of Braille music notation

Braille music notation is syntactically analyzed according to the tree structure of the MusicXML format so that a tree structure of Braille music notation can be obtained as shown in Figure 6. For keyed instruments such as piano and organ, a *staff* element is introduced and distinguishes between right- and left-hand parts using its attribute. In order to handle a broken chord such as an arpeggio without moving the time counter with the backup element, a *voice* element is introduced in the lower level of the *staff* element. The *voice* element also presents a voice such as an entire measure or a part-measure section. The *voice* element has an attribute of starting and ending time counter of its own *voice* element. The attribute enables the *voice* element to represent parallel voices indicated with the full-measure in-accord sign or with the measure-division sign and the part-measure in-accord sign. In the lower level of the *voice* element, elements such as *note*, *chord*, and *rest* elements are placed. The *chord* element has its including *note* elements in the lower level. The *note* elements are placed in descending order from the highest sound for the treble clef and in ascending order from the lowest sound for the bass clef. A tuplet is represented using the first *note* element of the tuplet. The *note* element has a *tuplet* element in the lower level, whose value indicates number of *note* elements of the tuplet.

4.3. Categorization of structure transformation

Structure transformation from the MusicXML tree to a Braille music tree is categorized by comparing and contrasting. The resultant categories are shown in Figure 7.

The five categories are preserving, upper inserting, grouping, simplification, and "other." Upper inserting and grouping categories were created because of the syntactical difference between the MusicXML format and Braille music notation.

The simplification category was obtained to remove elements for musicological information because the MusicXML format is designed to describe music with the aspect of musicological information.

The "other" category contains all other possible transformations. Structure transformation is generally performed using XSLT except DOM for grouping and the "other" category.

4.4. Technique for parallel voice division

Parallel voices are represented as a sequence of elements in both MusicXML data and Braille music scores. However, MusicXML format and Braille music notation have different ways to represent parallel voices.

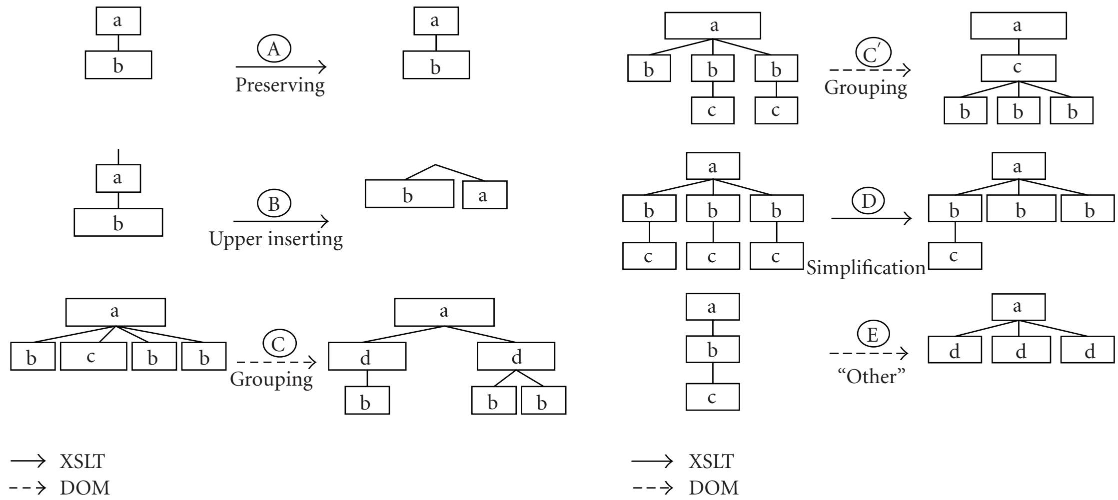


Figure 7: Categories of structure transformation from MusicXML trees to Braille music trees. The left side indicated by an arrow is a part of MusicXML trees and the right side a part of Braille music trees. Preserving is used for cases such as the lower-level element of note element, upper inserting for cases such as octave, accidental, and dot element, grouping for cases such as backup and chord element, simplification for cases such as time modification and note element. "Other" category contains all other possible transformations based on individual elements such as key and direction elements. Here a transformation for the key element is shown as a simple example.

MusicXML format has a concept of time counter. The backup element makes the time counter move backward in order to represent parallel voices sequentially. Figure 8 shows an example of parallel voices in the staff notation.

For the course of music, let $N1$, $N2$, $N3$, and $N4$ be parts of the parallel voices. A part Ni contains notes and can be written as follows with regard only to duration:

$$N1 = \left\{n\frac{1}{8}, n\frac{1}{8}, n\frac{1}{8}, n\frac{1}{8}\right\}, \qquad N2 = \left\{n\frac{1}{2}\right\},$$
$$N3 = \left\{n\frac{1}{8}, n\frac{1}{8}, n\frac{1}{8}, n\frac{1}{8}\right\}, \qquad N4 = \left\{n\frac{1}{2}\right\}, \tag{1}$$

where nt is a note with the note value as the duration time t.

MusicXML format enables us to write the parallel voices in many ways. Let $\langle bt \rangle$ be the backup element to make the time counter move backward for duration time t. Some of the representations are as follows:

$$N1\left\langle b\frac{1}{2}\right\rangle N2N3\left\langle b\frac{1}{2}\right\rangle N4, \tag{2}$$

$$N1N3\left\langle b\frac{1}{1}\right\rangle N2N4, \ldots . \tag{3}$$

The case of (2) is shown in Figure 9(a).

Furthermore, it is even possible to divide $N1$ and/or $N3$ into pieces. This method allows various representations for parallel voices. MusicXML data for the parallel voices may be generated differently depending on the software used for converting scores into MusicXML data. Note that each representation of the parallel voices could not always be based on the melody line.

In Braille music notation, parallel voices should be divided into a pair of voices as shown in Figure 9(b). In addition, it is desirable for the beginners to divide the parallel voices based on the melody line in order to improve the readability. Our parallel voice division provides the divided voices in the right way, so that our system could minimize its count of the backup element in each measure of the MusicXML data. Meanwhile, the parallel voice division may not provide the voice based on the melody line because it is difficult to recognize the melody line with only syntax analysis. However, our system divides parallel voices into a pair of voices to correspond to Braille representations as shown in Figure 3.

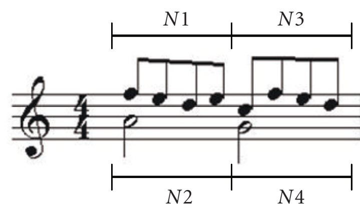
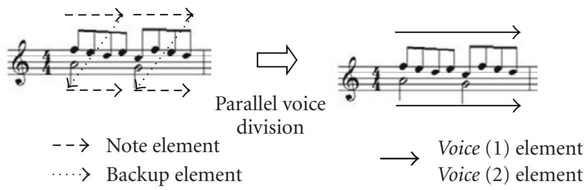


Figure 8: The course of music in parallel voices.



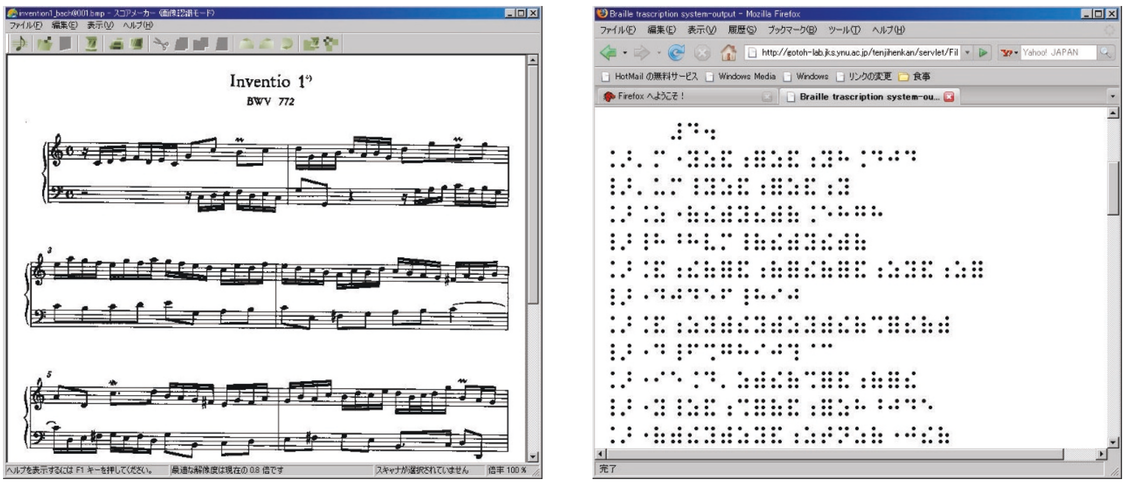
(a) Connectivity of elements in a given MusicXML data

(b) Divided voices in the Braille representation

Figure 9: A change of connectivity of elements by parallel voice division.

```
For each measure in input MusicXML data
    Seek sets of elements segmented by backup element.
      For each set of the sought sets of elements
        Divide the set into pieces based on a meter.
          For each piece in the set
            Add attribute of the starting and ending time counter.
          End
      End
      For each piece in all divided pieces
        Seek candidates of the next which the piece may be connected to.
        For each of the candidates
        Calculate the closeness to the piece.
        End
        Select one with the highest closeness as the next.
        Connect the piece to the next.
      End
End
```

Algorithm 1: Parallel voice division.



(a) A music score used in the experiments. This score was converted to MusicXML data using a commercially available program for scanning and editing

(b) The resultant Braille music score displayed on our web site. The obtained MusicXML data was transcribed by the transcription system via the Internet

Figure 10: A sample of the resultant Braille music score.

A sequence of note elements segmented by the backup element is divided into pieces based on a meter, as shown in Algorithm 1. Once the initial piece is determined, the next is selected from the connectable pieces to the piece by evaluating the closeness of the piece until all pieces are connected. Then each voice is composed to connect the pieces in selected order. The closeness of the piece is currently defined as a weighted average of two terms, one for pitch and the other for duration. Each term is computed as a difference between means of pitch or duration for the piece and for another piece.

5. FEASIBILITY STUDY

Experiments were performed to check the feasibility of the transcription system to provide universal access to music scores. Figure 10 shows an example of a Braille music score transcribed by the system. The resultant Braille scores were inspected by volunteer transcribers. They verified that the Braille scores were correctly transcribed and suitable for the beginners. It can be said that the algorithm of parallel voice division can divide simply as shown in Figure 9(b). However, it is not always the case that parallel voices are

Table 5: The processing time required transcribing.

Music score	In MusicXML data		Time (s)
	Measure (no.)	Element (no.)	
Suite 1 G-dur, BWV1007 Praeludium (J.S.Bach)	42	7320	6.14
Suite 1 G-dur, BWV1007 Allemande (J.S.Bach)	34	5335	5.25
Suite 1 G-dur, BWV1007 Courante (J.S.Bach)	44	4908	5.00
Suite 1 G-dur, BWV1007 Sarabande (J.S.Bach)	16	1903	4.16
Suite 1 G-dur, BWV1007 Menuetto1 (J.S.Bach)	24	1684	4.09
Suite 1 G-dur, BWV1007 Menuetto2 (J.S.Bach)	24	1568	3.94
Suite 1 G-dur, BWV1007 Gigue (J.S.Bach)	37	2495	4.30
Inventio 1 BWV 772 (J.S.Bach)	22	5296	6.78
Inventio 4 BWV 775 (J.S.Bach)	52	5310	5.61
Fantaisie-impromptu (Frederic Chopin)	143	46679	122.63

Table 6: Available Braille signs for music symbols.

Music symbol		Availability
Note		○
Harmony	Chord	○
	In-accord	△
Rest		○
Ornament		△
Slur		X
Tie		○
Rhythmic groups		X
Dynamic mark		○
Expression mark		X
Portato		○
Repeat mark		△
Key signature		○
Time signature		○
Fingering		X

○: available, X: not available,
△: partly available (available for standard symbols).

divided along the melody line. Automation of this recognition process including the melody line is a very challenging subject of pattern recognition as well as artificial intelligence.

Table 5 shows the processing time required of transcribing in the experiments. The time required for processing was fast enough for practical use. It suggests that the system may be practically used to transcribe from the MusicXML format to Braille music notation. The experiments as shown in the last three lines of Table 5 were performed to use the music score for keyed instruments. The right-hand part and left hand part are represented using the *staff* element with its attributes (1) and (2), respectively. It is confirmed that music symbols were handled as shown in Table 6. They are generally used in the standard music scores for the beginner musicians.

As our system is transcription based on the MusicXML format to produce Braille music scores, the Japanese version of the system is currently available for those who are interested on the World Wide Web. There are about 200 hits a month on the page. The trial English version is also currently available at the following web page: http://gotoh-lab.jks.ynu.ac.jp/GOTOH_HP_e/index.html.

In order to improve the ability to transcribe, a function will be added to modify the divided individual voices to match them with the melody line, to transcribe MusicXML data to Braille music scores for experienced musicians, to provide additional functions of signs for modern notation, and to develop a friendly interface for visually impaired users. There remains considerable work to be done in this transcription as new technologies emerge in the multimedia music field.

6. CONCLUSION

In this paper, we have presented a transcription system based on the MusicXML format to produce Braille music scores with structure transformation using a structure model of Braille music notation. The experimental results have shown that this approach is suited for producing international standard Braille music scores for the beginners. The presented algorithm of parallel voice division is effective for simple division. In addition, the presented structure model and structure transformation lend themselves to efficiently produce Braille music scores so the transcription system should be put to practical use via the Internet.

ACKNOWLEDGMENTS

The authors would like to thank TONICA (The group of volunteer Braille music transcribers) for helpful discussions and close inspection of transcribed Braille music scores. The work was supported in part by Grant no. 15500062 of Grant-in-Aid for Scientific Research from Ministry of Education, Culture, Sports, Science, and Technology of Japan.

REFERENCES

[1] K. Sato, M. Komatsu, Y. Oyama, S. Matsuzaki, and K. Kanda, "Construction of the Braille score system," IPSJ SIG Technical Reports 2000-MUS-36-7, pp. 37–42, Information Processing Society of Japan, Tokyo, Japan, 2000, (in Japanese).

[2] D. Langolff, N. Baptiste-Jessel, and D. Levy, "NIFF transcription and generation of Braille musical scores," in *The Virtual Score: Representation, Retrieval, Restoration*, vol. 12 of *Computing in Musicology*, pp. 34–44, MIT Press, Cambridge, Mass, USA, 2001.

[3] Toccata, "Braille Music Translation Software—Toccata," 2006, http://members.optusnet.com.au/~terryk/toccata.htm.

[4] G. Yusa, H. Adachi, T. Gotoh, N. Tamura, and Y. Emori, "A translation system of Braille-score from MusicXML," IPSJ SIG Technical Reports 2004-DD-42, pp. 15–22, Information Processing Society of Japan, Tokyo, Japan, 2004, (in Japanese).

[5] T. Matsushima, M. Tanaka, S. Hashimoto, S. Ohteru, and Y. Enari, "Automatic translation of Braille music using computerized type-setting data," in *Abstracts of the 10th Sensory Substitution Symposium on Association of Sensory Substitution*, pp. 93–97, Tokyo, Japan, 1984, (in Japanese).

[6] Y. Emori, D. Kuroda, and H. Nakatsuka, "The music publishing and playback system by the music information DTD, the music information using XML technology," Information-technology Promotion Agency, Japan, Musical Plan, 2001, (in Japanese).

[7] M. Good, "MusicXML for notation and analysis," in *The Virtual Score: Representation, Retrieval, Restoration*, vol. 12 of *Computing in Musicology*, pp. 113–124, MIT Press, Cambridge, Mass, USA, 2001.

[8] W. R. McCann, "New tools for transcribers and blind musicians from duxbury and dancing dots," in *Proceedings of Technology and Persons with Disabilities Conference*, Los Angeles, Calif, USA, March 2005, http://www.csun.edu/cod/conf/2005/proceedings/2461.htm.

[9] Recordare, "MusicXML software," 2006, http://www.recordare.com/software.html.

[10] T. Kahlisch, M. Leopold, and C. Waldvogel, "DaCapo, a project on transforming ink printed to Braille notes semi-automatically: the project's current state," in *Proceedings of the 9th International Conference on Computers Helping People with Special Needs (ICCHP '04)*, vol. 3118 of *Lecture Notes in Computer Science*, pp. 224–227, Paris, France, July 2004.

[11] B. Krolick, "New international manual of Braille music notation," The Braille Music Subcommittee World Blind Union, 2006, http://www.brl.org/music/manual/index.html.

D. Goto received his M.S. degree from Yokohama National University in 2006. He joined Yahoo Japan Corporation in the same year. From 2004 to 2006, he worked in the development of Braille music translation system in the university.

T. Gotoh received his B.S. and Ph.D. degrees from Tokyo Institute of Technology in 1975 and 1981, respectively. He joined Fujitsu Laboratories LTD, in 1981. He was appointed Associate Professor at Faculty of Engineering of Yokohama National University in 1990, a Professor at the same faculty in 1996, and a Professor at Graduate Scholl of Environment and Information Systems of the same university. He has engaged in research and education of image processing, computer graphics, medical image analysis, and assistive information technology.

R. Minamikawa-Tachino received her M.S. degree from Tsuda college, Japan, in 1979 and a Ph.D. degree in computer science from Ochanomizu University, Japan, in 2000. She joined the Tokyo Metropolitan Institute of Medical Science (TIMS) in 1979. She has engaged in research of signal processing and image processing in biomedicine. She is the Senior Research Fellow at the TIMS since 2001. Her current research interests include image processing, computer graphics, and pattern recognition in biomedicine and social assistance.

N. Tamura was born in Tokyo, Japan, in 1956. He received an Engineering degree of computer science in 1980 from the Tokyo Institute of Technology, where he received the M.S. degree, in 1982, and the Ph.D. degree, in 1985, in computer science. His current research deals with Braille note processing for mutual transcription between MusicXML scores and Braille notes and problems in construction of Braille note database. He is also interested in natural language processing, especially in the temporal structure and text structure analysis for the newspaper articles and editorials. He is presently a Professor of the Department of Information Media and Environment Sciences, of the Graduate School of Environment and Information Sciences, Yokohama National University.