

オンライン手書き楽譜入力システムの試作

永井聡子 天野健児 野口将人 松島俊明

東邦大学理学部 情報科学科

〒274-8510 千葉県船橋市三山 2-2-1

{satoko_n, ken, masato}@ma.is.sci.toho-u.ac.jp, matusima@is.sci.toho-u.ac.jp

あらまし: 既存の音楽ソフトの多くが採用している入力方式は、鍵盤楽器による演奏入力を利用する方式か、メニューから選んだ記号を五線上に貼り付けて行うものである。前者は鍵盤楽器の演奏に慣れていないユーザにとっては使い難く、後者は操作が煩雑で入力に手間がかかるという問題がある。筆者らは、簡単な操作で効率的に楽譜作成が出来る方式として、電子ペンによる手書き入力を利用した楽譜作成システムの研究を行っている。このシステムでは、音符、休符、付点等入力頻度の高い記号については、手書き入力したストロークの形状・位置・既入力記号との関係を調べて自動認識を行い、音部記号や調号等、入力頻度の低い記号は、記号リストから選択して五線上に貼付けて入力する。2つの入力方法の併用で、効率的に楽譜の作成を行うことができる。今回、Windows上で動作するシステムと共に、PDAで動作するシステムを試作したので報告する。

キーワード: 楽譜エディタ, DTMシステム, 手書き入力, PDA

An Experimental Study on Online Handwriting Music Notation Editing System

Satoko NAGAI, Kenji Amano, Masato NOGUCHI and Toshiaki MATSUSHIMA

TOHO University

2-2-1 Miyama, Funabashi

Chiba, 274-8510 JAPAN

Abstract: There exists a lot of music notation software. Most of them adopts to play musical keyboard or to select item from menu list as the input method of the notation symbols. Former method is difficult for non-experienced keyboard players, and latter is niggling task to take much time to input. The authors are studying an online handwritten score making system which can be input notation symbols easier and faster utilized pen and tablet interface. In this system, major symbols such as notes, rests and dots can be input by writing their shapes on a staff. Written strokes are automatically recognized as notation symbols according to their shapes, positions and relation to existing symbols. The other symbols such as clefs and key signatures, which are not so frequently input, are input by menu selection. With these two input methods, it is possible to make a musical score effectively. In this paper, we introduce the system for Windows and PDA.

Keywords: Music Notation Editor, Desktop Music System, Handwriting Editor, PDA

1. はじめに

コンピュータを利用した音楽ソフトにとって、楽譜データを如何に効率的に入力することができるかということは、重要な問題である。現在市販されている楽譜作成ソフトでは、鍵盤楽器による演奏入力を利用する方式のものか、マウス等でメニューから選んだ記号を五線譜上に貼り付けて行うものが主流である。これらの入力方法は、作曲家や編曲家が行っている五線譜上に楽譜記号を筆記してゆくという伝統的な楽譜作成方法と比べて、必ずしも使い勝手が良いとは言えない。そのため、これらに代わる楽譜データ入力の方式として、タブレットによる手書き楽譜入力が従来から提案されている。例えば、図1に示したように、五線上に簡単なストロークを筆記するだけで、音高と音価を同時に入力することが可能である。古い例では、全音符から32分音符に対応する6種類のストロークを認識して入力する方法が紹介されている[1]。他の入力方法に比べて遥かに流暢に入力ができること述べられているが、その一方で、記号に対応したストローク形状を正しく覚えておかなければならない点が欠点として指摘されている。また **Pen-Based Music Editor** [2] ではペンの入力位置を指定し、引き続き **Pop-up** メニューで音符の種類を指定する方法が報告されている。一方、**Presto** [3][4]や**Music Notepad** [5]では、新しい4分音符や4分休符を作る単純なストロークと、音価を2倍および半分にするストロークを組み合わせるさまざまな音符や休符が入力できるように工夫されているが、比較的出現頻度の低い記号やさまざまな音価の音符や休符も直接入力できるよ

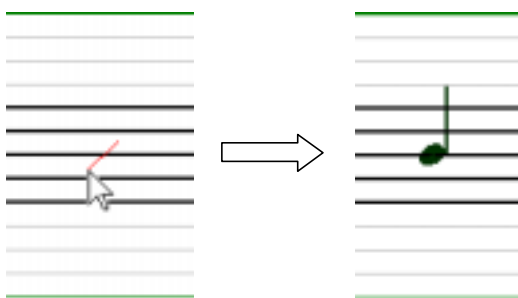


図1 手書きによる楽譜入力例

うにしているため、記号を表すストロークだけでも **Presto** では17種類、**Music Notepad** では23種類のストロークを用意している。そのため、複雑な形状のストロークの認識が必要であったり、また実際の記号形状とは異なる形状のストロークを割当てなければならないため、記号に対応したストロークが直感的に分かり難くなっている。

筆者らは従来より手書き入力とメニュー選択方式を組み合わせることで、効率的に尺八譜の作成が行えるシステムの研究を行ってきた[6][7]。本研究では、ストローク認識とメニュー選択を組み合わせることで、より使いやすくスムーズに楽譜作成ができるシステムを、**Windows**¹と携帯可能なPDAである**Zaurus**²上で実現することで、いつでもどこでも楽譜作成ができるシステムの構築を目標としている。

2. 本システムの記号入力方式

五線譜に書かれる記号には、その形状が付点や小節線のように非常に単純なものから、ト音記号や連符で繋がった音符の塊のように複雑な形状のものまで多種多様である。これら全ての記号を手書きにより入力するためには、各記号に対応した様々な形状のストロークを用意するか、記号形状とは無関係なストロークを割当てる必要があるが、前者は誤認識が生じ易くなり、後者は記号とストロークの対応が分かり難いため操作性が悪くなる。

しかし、一般的に楽譜記号の出現頻度には大きな差がある。例えば、文献[3]に7曲7頁分の楽譜を調べた結果が報告されているが、それによると、音符、休符、付点が全体の70%近くを占め、次いで臨時記号が12%となっている。従って、この記号種の偏在性を利用し、以下の方針による記号入力方法を採用することで、効率的に楽譜記号が入力できるよう工夫した。

- ① 認識精度を高めるため、認識対象のストロー

¹ Windows は Microsoft 社の登録商標である。

² Zaurus は(株)シャープの登録商標である。

クの形状は単純なものに制限し、また、その種類も可能な限り限定する。

- ② 入力頻度の高い音符・休符・付点などの記号については、直接ストロークを筆記することにより入力できるようにする。
- ③ 記号毎に複雑な形状のストロークを割当てる代わりに、ストロークの組み合わせで記号を表すようにする。
- ④ 記号を構成するストローク・セットが全て入力された後に記号の認識を行うのではなく、最も基本的なストロークに対応した記号を基本記号、その他の記号を派生記号とし、基本記号

に追加されたストロークの形状・位置・既入力記号との関係により順次派生記号として認識を行う。

- ⑤ 入力頻度が低い音部記号や調号、シャープ、フラットなどの記号は、記号のメニューから選択して入力を行う。

3. 入力ストロークの処理と認識方法

本システムでは手書きで入力する記号の種類と形状を考慮し、図2に示した10種類のストロークを採用した。①～⑨は音符および休符用のストローク、⑩は付点用で左クリックで入力する。①～⑩の単純なストロークの組み合わせで記号の入力が行えるように、既存記号の周辺に新たなストロークを追加していくことで、別の記号へと変化させていく方法を採用した。音符および休符を入力するストロークと記号の変化の様子を図3および図4に示す。音符については全音符と4分音符、休符については8分音符以外が基本記号となる。例えば、周囲に記号が存在しない五線上にストローク①を筆記すると4分音符が入力され、その記号の上方または下方にストローク①を筆記すると和音に、そ

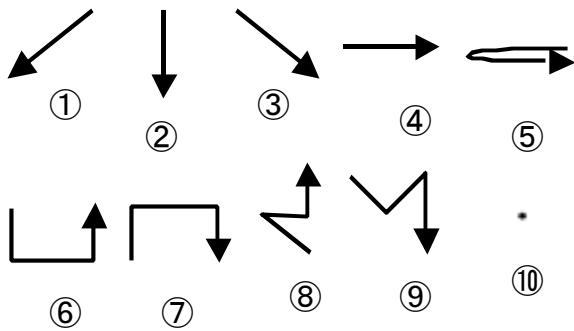


図2 基本ストローク

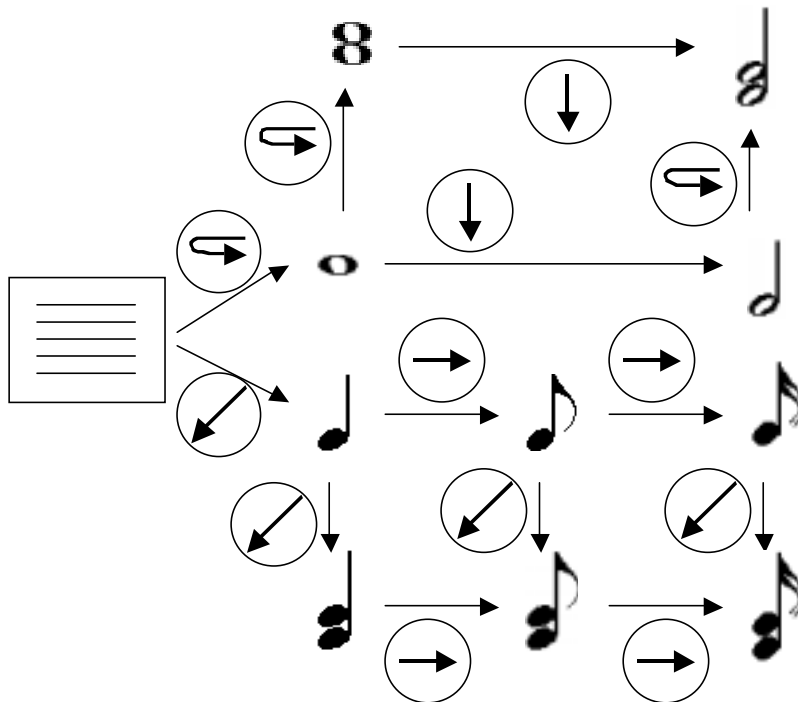


図3 音符を入力するストロークと記号の遷移

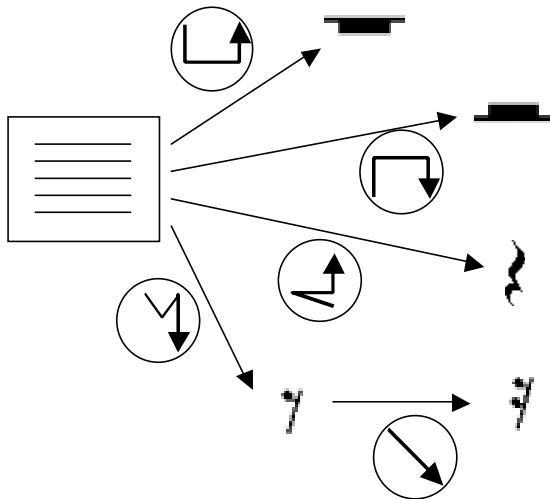


図4 休符を入力するストロークと記号の遷移

の記号の周辺にストローク④を筆記すると 8 分音符, さらに 16 分音符へと変化していく。

入力されたストロークは以下の手順で認識が行われる。オンライン入力では応答性が重要であるため, 出来るだけ単純な処理で認識を行うよう考慮した。入力ストロークは, 最終的に 8 方向のチェインコードに表して処理するが, ユーザが筆記したストロークデータには冗長なサンプル点が多数含まれているため, まず, ストロークの圧縮処理を行う。点間の方向変化が小さいサンプル点と点間の距離が小さいサンプル点を除去した後, 直線部分を統合して冗長なチェインコードが少なくなるよう, 再帰的にサンプル点の除去を行ってストロークを圧縮する。

圧縮されたストロークは, 標準パターンのチェインコードと比較を行う。図 2 に示したストロークは, 1 から 3 方向のチェインコードで表すことができる。一般的に, 長さの異なるパターン同士の比較は DP マッチングにより行うことが多いが, 本システムでは, パターンの系列長が短いことを利用し, より簡便な次の方法により求めることとした。

入力ストロークを x , k 番目の標準パターンを $x^{(k)}$ とし, それぞれ n 個, m 個のチェインコードからなるとする。即ち,

$$x = (x_1, x_2, \dots, x_n), x^{(k)} = (x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}) \quad \dots(1)$$

と表せるとする。 n と m の最小公倍数が l であるとす。 x を構成する各要素をそれぞれ l/n 回ずつ繰り返して得られるパターン $y = (y_1, y_2, \dots, y_l)$, $x^{(k)}$ を構成する各要素を l/m 回ずつ繰り返して得られるパターン $y^{(k)} = (y_1^{(k)}, y_2^{(k)}, \dots, y_l^{(k)})$ は, 共に同じ長さ l のパターンになる。これにより, (1)式で与えられた系列長が異なる入力パターン x と標準パターン $x^{(k)}$ の類似度 $d(x, x^{(k)})$ を(2)式で計算することができる。

$$d(x, x^{(k)}) = d(y, y^{(k)}) = \frac{\left\{ \sum_{i=1}^l (y_i - y_i^{(k)})^2 \right\}^{1/2}}{l} \dots(2)$$

全ての k について類似度を計算し, 最も小さい値を与える k を x のパターンに決定する。 n と m が互いに素かつ大きな値の場合は l が非常に大きな値になり非効率的であるが, n, m 共に高々 3 程度であるため l は高々 10~20 程度にしかならず, 処理に要する時間も短くて済む。

4. ユーザインターフェースと機能

本研究では, PC 用は Windows 上で, PDA 用はシャープの Zaurus 上に構築を行った。 PC と PDA では, 処理能力や画面の大きさが異なるため, ユーザインターフェースや入力可能な記号などが異なっている。以下, 各システムの画面設定と操作方法について述べる。

4.1 Windows 版システム

Windows 版は記号の入力ウィンドウ(図 5)と音符等以外の記号入力のための Tool Box(図 6)で構成されている。まず Tool Box の「①パート追加」を選択して「パートの追加」ダイアログ(図 7)を表示し, 音部記号・調号・拍子記号のパート情報を指定して五線を生成する。音符, 休符, 付点は五線上に直接筆記して入力を行い, 臨時記号は Tool

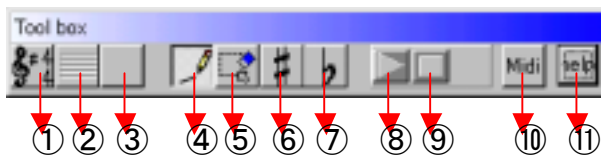
Box の⑥または⑦で種類を選択後、臨時記号を追加したい符頭をクリックして入力する。なお、今回は SMF 形式で表現可能な情報を主としたため、符尾の向きや連符、スタッカート等の発想記号については入力対象外とした。

4.2 Zaurus 版システム

Zaurus では表示領域が 240x320 画素と Windows に比べて非常に小さい。そのため、入



図5 Windows版入力画面



- ①パート追加 ②五線追加 ③音符位置最適化 ④ペン入力モード ⑤音符削除 ⑥シャープ追加 ⑦フラット追加 ⑧演奏 ⑨演奏停止 ⑩SMFで保存 ⑪ヘルプ表示

図6 Windows版でのTool Boxメニュー



図7 パートの追加ダイアログ

力可能な楽譜を 1 パートに限定し、Tool Box もメニュー化して画面の上部および下部に配置した (図 8)。音符、休符、臨時記号については Windows 版と同様に入力が可能である。演奏機能や SMF での保存機能はないが、Windows 版とデータの互換性があるので、Zaurus 上で作成した楽譜を Windows 上で再編集したり、演奏することが可能である。

5. システム評価

図 9 に Windows 版で入力した 2 パート 2 段の楽譜の例を示す。図 10 は Zaurus 版での楽譜入力の様子である。本システムの有効性を確認する



図8 Zaurus版入力画面

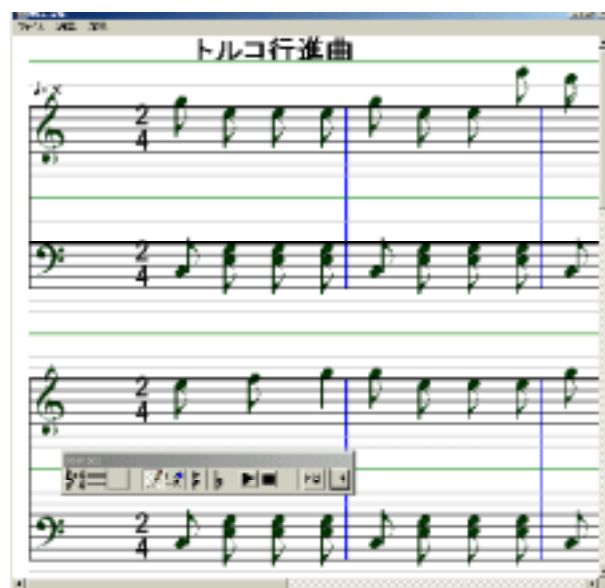


図9 Windows版での楽譜作成例



図10 Zaurusによる楽譜入力の様子

ため、12小節1パート分の楽譜を入力するのに要した時間を、本システム(Windows版)とメニューからマウスで記号を選択して五線上に貼り付ける方式の市販ソフトで測定した。全ての被験者が両方のソフトを使用するのが始めてだったため、入力方法の練習を暫く行った後に実験を行った。表1に実験結果を示す。本システムによる入力の方が、メニューから選択して五線上に貼り付ける方法よりも1.5倍から2倍くらい速いことが分る。このような結果が得られた最も大きな理由は、手書きによる入力では音符選択に要する時間ロスがないためと思われる。

表1 市販のDTMソフトと本システムによる楽譜データ入力時間の比較

	市販のDTMソフト	本システム
被験者1	3分41秒	2分1秒
被験者2	2分40秒	1分53秒
被験者3	3分51秒	2分11秒
平均	3分24秒	2分2秒

6. まとめ

WindowsおよびPDA上で動作する手書き楽譜認識システムの試作を行った。特にPDAはペンによる操作が一般的であるため、手書きによる楽譜入力システムはPDAに適したアプリケーションであると言える。携帯性に優れたPDA上で楽譜作成が可能となることで、いつでもどこでも手軽に作曲・編曲が可能な環境が実現できる。

今後の課題としては、楽譜の表示品質の向上と共に、現在対象外となっている連桁や各種の発想記号を入力できるようにすること、また、NIFFやMusicXML等汎用の楽譜記述形式への対応を行っていく予定である。卒業研究で本研究に従事した斎藤智治氏(現富士ソフトABC)に感謝する。

参考文献

- [1] William Buxton, Richard Sniderman, William Reeves, Sanand Patel and Ronald Baecher: The Evolution of the SSSP Score Editing Tools, Computer Music Journal, Vol.3, No.4, pp.14-25 (1979).
- [2] Katherine Silberger: Putting Composers in Control, IBM Research, Vol.23, No.4, pp.14-15 (1996).
- [3] Jamie Anstice, Tim Bell, Andy Cockburn and Martin Setchell: The Design of a Pen-Based Musical Input System, OzCHI'96: The Australian Conference on Computer-Human Interaction, pp.260-267 (1996).
- [4] Elizabeth Ng, Tim Bell and Andy Cockburn: Improvements to a Pen-Based Musical Input System, OzCHI'98: The Australian Conference on Computer-Human Interaction, pp.239-252 (1998).
- [5] Andrew Forsberg, Mark Dieterich, and Robert Zeleznik: The Music Notepad, ACM Symposium on User Interface Software & Technology, pp.203-210 (1998).
- [6] Toshiaki Matsushima: Recognition and Editing System for Shakuhachi Tablature, Proc. Intl. Computer Music Conf. '93, pp.405-407 (1993).
- [7] 野口将人, 田島ゆう子, 松島俊明, 坪井邦明, 志村哲: 尺八譜の情報処理システム「尺八くん2002」—システム評価と新機能—, 情報処理学会研究報告, MUS-47-10 (2002).