

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-106989

(P2005-106989A)

(43) 公開日 平成17年4月21日(2005.4.21)

(51) Int. Cl.⁷

G10G 3/04

G10H 1/00

F I

G10G 3/04

G10H 1/00

テーマコード (参考)

5D082

5D378

審査請求 未請求 請求項の数 3 O L (全 12 頁)

(21) 出願番号 特願2003-337914 (P2003-337914)

(22) 出願日 平成15年9月29日 (2003.9.29)

(71) 出願人 000004075

ヤマハ株式会社

静岡県浜松市中沢町10番1号

(74) 代理人 100107995

弁理士 岡部 恵行

(72) 発明者 山内 明

静岡県浜松市中沢町10番1号 ヤマハ株式会社内

Fターム(参考) 5D082 BB01 BB21

5D378 FF19 QQ25

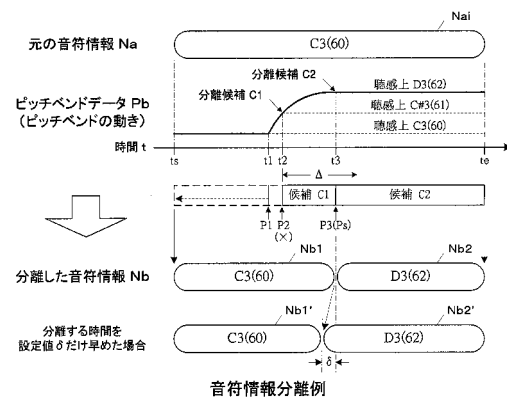
(54) 【発明の名称】 楽譜対応データ生成装置及びプログラム

(57) 【要約】

【課題】演奏データを分析してピッチベンド等により表情付けがなされた音符情報を検出し、聴感上の音高変化に応じて複数の音符情報に分離された楽譜対応データを生成する。

【解決手段】この楽譜対応データ生成システムでは、演奏データ中の音符情報Na及び表情付け情報Pbを分析する。表情付け情報は、各音符情報Na iの聴感上の音高を変化させるピッチベンドデータPb等の情報であり、各音符情報Na i毎に、表情付け情報Pbに基づき所定の音高変化(半音)を与える音高変化点t2、t3を検出し、表情付け情報Pbに対応する奏法を認識する。そして、認識された奏法から元の音符情報Naを分離した方が楽譜に対応すると判断したときは、分離ポイントPsに従って聴感上の音高変化に合う複数の音符情報Nb1、Nb2; Nb1', Nb2'を生成する。分離ポイントPsは、音高変化点t2、t3の内、分離後の音符長が短くなり過ぎない点に決定される。

【選択図】 図2



【特許請求の範囲】

【請求項 1】

演奏データを取得する取得手段と、
取得された演奏データ中の各音符情報、及び、該音符情報の聴感上の音高を変化させる表情付け情報を抽出する抽出手段と、
抽出された表情付け情報に対応する奏法を認識する認識手段と、
認識された奏法に基づいて、抽出された音符情報を異なる音高の複数の音符情報に分離した方が楽譜に対応するか否かを判定する判定手段と、
分離した方が楽譜に対応すると判断されたとき、聴感上の音高が変化するポイントを元にして、当該音符情報を分離し、聴感上の音高変化に合う複数の音符情報を生成する生成手段と
を具備することを特徴とする楽譜対応データ生成装置。 10

【請求項 2】

前記生成手段は、生成される音符情報の音符長が短くなり過ぎないように前記ポイントを決定する手段を有することを特徴とする請求項 1 に記載の楽譜対応データ生成装置。

【請求項 3】

演奏データを取得するステップと、
取得された演奏データ中の各音符情報、及び、該音符情報の聴感上の音高を変化させる表情付け情報を抽出するステップと、
抽出された表情付け情報に対応する奏法を認識するステップと、
認識された奏法に基づいて、抽出された音符情報を異なる音高の複数の音符情報に分離した方が楽譜に対応するか否かを判定するステップと、
分離した方が楽譜に対応すると判断されたとき、聴感上の音高が変化するポイントを元にして、当該音符情報を分離し、聴感上の音高変化に合う複数の音符情報を生成するステップと
から成る手順を音楽情報処理装置に実行させるための楽譜対応データ生成プログラム。 20

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、聴感上の音高を変化させる表情付け情報を有する演奏データを分析して楽譜に対応したデータを得る楽譜対応データ生成システムに関する。 30

【背景技術】

【0002】

従来より、演奏データより音符情報を分析して楽譜表示するものが、種々、提案されている。例えば、特許文献 1 に示されるように、演奏データから装飾音符やグリッサンド、アルペジオなどの奏法を検出し、楽譜上で音楽記号などを用いて音符を密集させずに表示することができる。

【特許文献 1】特開平 11-327427 号公報

【0003】

このような従来の方法では、複数の音符情報のタイミング及び長さ等から検出可能な奏法については対応することができるが、1つの音符情報に対応した複数のピッチベンド情報が表現する奏法については対応しきれず、元の 1つの音符のまま表示されてしまう。 40

【発明の開示】

【発明が解決しようとする課題】

【0004】

この発明は、このような事情に鑑み、演奏データを分析してピッチベンド等により表情付けがなされた音符情報を検出し、聴感上の音高変化に応じて複数の音符情報に分離された楽譜対応データを生成することができる楽譜対応データ生成システムを提供することを目的とする。

【課題を解決するための手段】

【0005】

この発明の主たる特徴に従うと、演奏データを取得する取得手段（S3）と、取得された演奏データ中の各音符情報（N a i）、及び、該音符情報（N a i）の聴感上の音高を変化させる表情付け情報（P b）を抽出する抽出手段（S6）と、抽出された表情付け情報（P b）に対応する奏法を認識する認識手段（S7）と、認識された奏法に基づいて、抽出された音符情報（N a i）を異なる音高の複数の音符情報に分離した方が楽譜に対応するか否かを判定する判定手段（S10）と、分離した方が楽譜に対応すると判断されたとき、聴感上の音高が変化するポイント（例えば、t3）を元にして、当該音符情報（N a i）を分離し、聴感上の音高変化に合う複数の音符情報（N b 1, N b 2 ; N b 1', N b 2'）を生成する生成手段（S11；図4）とを具備する楽譜対応データ生成装置〔請求項1〕、並びに、演奏データを取得するステップ（S3）と、取得された演奏データ中の各音符情報（N a i）、及び、該音符情報の聴感上の音高を変化させる表情付け情報（P b）を抽出するステップ（S6）と、抽出された表情付け情報（P b）に対応する奏法を認識するステップ（S7）と、認識された奏法に基づいて、抽出された音符情報（N a i）を異なる音高の複数の音符情報に分離した方が楽譜に対応するか否かを判定するステップ（S10）と、分離した方が楽譜に対応すると判断されたとき、聴感上の音高が変化するポイント（t3）を元にして、当該音符情報（N a i）を分離し、聴感上の音高変化に合う複数の音符情報（N b 1, N b 2 ; N b 1', N b 2'）を生成するステップ（S11；図4）とから成る手順を音楽情報処理装置に実行させるための楽譜対応データ生成プログラム〔請求項3〕が提供される。なお、括弧書きは、理解の便のために、対応する実施例の参照記号等を付記したものであり、以下においても同様である。

【0006】

この発明の楽譜対応データ生成装置における生成手段（S11；図4）は、生成される音符情報（N b 1, N b 2 ; N b 1', N b 2'）の音符長が短くなり過ぎないようにポイントを決める手段（S P 3）を有する〔請求項2〕ように構成することができる。

【発明の効果】

【0007】

この発明による楽譜対応データ生成システムでは、演奏データが供給されると（S3）、この演奏データ中にある各音符情報（N a i）とその音符情報（N a i）の聴感上の音高を変化させる表情付け情報（ピッチベンドデータP bなど）とを抽出し（S6）、抽出した表情付け情報（P b）に対応する奏法を認識する（S7）。そして、認識した奏法より、元の音符を分離した方が楽譜に対応すると判断された場合に（S10→YES）、聴感上の音高が変化するポイント（例えば、t3）を元にして、音符情報（N a i）を分離し、聴感上の音高変化に合う複数の音符情報（N b 1, N b 2 ; N b 1', N b 2'）を生成する（S11；図4）。なお、音符の分離に採用されるポイントは「分離ポイント」と呼ばれ、また、「分離」は、用語上、「分割」と言ってもよい。

【0008】

このように、この発明では、演奏データを分析して、楽譜に対応したデータ（「楽譜対応データ」という）を得るのに、まず、演奏データ中の音符情報（N a i）に対応するピッチベンドデータ（P b）などの表情付け情報を分析することにより、当該音符情報（N a i）に付加されている表情付け（奏法）の種類を推測する（S7）。そして、推測した表情付け（奏法）に従い、聴感上の音高差が元の音符情報（N a i）と等しい複数の音符情報（N b 1, N b 2 ; N b 1', N b 2'）を生成し（S11；図4）、これを楽譜対応データとして出力する（S15）。例えば、ピッチベンドデータ（P b）による表情付けがなされた音符情報（N a i）を検出すると、聴感上の音高変化が半音以上ある場合に、これを複数の音符情報（N b 1, N b 2 ; N b 1', N b 2'）に分離する。

【0009】

従って、音符情報に付加された表情付けを反映した楽譜対応データを生成することができる。つまり、生成された楽譜対応データは、記譜上も元の演奏データに対応し、聴感上の音高差も元データと等しいので、演奏データを楽譜で表示するのに利用したり、曲デー

タなどの演奏データについて聴感上も含めた音高の動きを調べるツールなどに適したものとすることができる。

【0010】

また、この発明の楽譜対応データ生成装置では、元の音符情報（N a i）を分離する場合、分離後の音符情報（N b 1, N b 2 ; N b 1', N b 2'）の音符長が短くなり過ぎないように分離ポイントを決定する（S P 3）ようにしている。例えば、表情付けにより所定の最小時間幅（ Δ ）以上の間に聴感上の所定の音高変化があったポイント（t 3）を分離ポイントとする。これにより、楽譜対応データを聴感上の音符に一致させると共に、必要以上に音符を分割しないように音符分離処理を改善することができる。

【0011】

なお、この発明においては、さらに、表情付けによる聴感上の所定の音高変化が終了するポイント（t 3）よりも所定の時間幅（ δ ）だけ早めに分離ポイントを変更する（S P 7）ように構成することにより、分離後の音符について、元データに対する聴感上の音高一致度を更に高めることができる。

【発明を実施するための最良の形態】

【0012】

〔システム構成〕

図1は、この発明の一実施例による楽譜対応データ生成システムのハードウェア構成例を示す。この例では、楽譜対応データ生成システムが構築される音楽情報処理装置には、パーソナルコンピュータ等の汎用の情報処理装置に演奏入力及び楽音生成機能を設けたものが用いられるが、電子楽器などの音楽専用の処理装置を用いてもよい。この音楽情報処理装置は、中央処理装置（C P U）1、ランダムアクセスメモリ（R A M）2、読出専用メモリ（R O M）3、外部記憶装置4、入力操作部5、表示部6、音源部7、通信インターフェース（通信 I / F）8等の要素を備え、これら要素1～8はバス9に接続される。

【0013】

装置全体を制御するC P U 1は、所定の制御プログラムに従って各種処理を実行し、R A M 2は、これらの処理で利用される種々の情報を一時的に記憶するための処理バッファとして機能する。例えば、楽譜対応データ生成プログラムによる処理時には、演奏データ中の音符情報（音符データともいう）や、同データ中のピッチベンドデータから得られる当該音符要素に対するポイント要素（音高変化点のピッチベンド値と位置）、生成される楽譜対応データ（分離後音符情報）等を格納するためのバッファ領域（Note, PBPointList, DataList）がR A M 2上に確保される。また、R O M 3は、楽譜対応データ生成プログラムを含む各種制御プログラムや必要な制御データ、各種音楽データを記憶している。

【0014】

外部記憶装置4は、ハードディスク（H D）や、コンパクトディスク・リード・オンリィ・メモリ（C D - R O M）、フレキシブルディスク（F D）、光磁気（M O）ディスク、デジタル多目的ディスク（D V D）、メモリカード等の記憶媒体を用いた記憶手段である。楽譜対応データ生成プログラムや制御データ、演奏データ等の各種データは、R O M 3だけでなく外部記憶装置4に記憶させることができる。

【0015】

例えば、R O M 3に制御プログラムが記憶されていない場合、H DやC D - R O Mなどの外部記憶装置4に制御プログラムを記憶させておきそれをR A M 2に読み込むことで、R O M 3に制御プログラムを記憶している場合と同様の動作をC P U 1にさせることができ、制御プログラムの追加やバージョンアップ等を容易に行うことができる。例えば、楽譜対応データ生成処理に用いられるプログラムや必要な制御データ等をインストールすることによって、所望の楽譜対応データ生成装置を実現することができる。

【0016】

入力操作部5は、パネル操作子や演奏操作子及び検出回路から成り、これらの操作子による設定乃至編集操作、演奏入力操作などの内容を検出回路で検出し、対応する入力情報をシステム本体内に導入する。例えば、マウスやキー等のパネル操作子にて、最小音長及

10

20

30

40

50

び移動時間幅 (Δ , δ) 等の楽譜対応データ生成処理に必要な各種情報を入力することができる。演奏操作子は、所定のキーやマウス等に音高入力要素や表情付け入力要素（ピッチベンダ）の機能が割り当てられるか、或いは、これらの機能を備えた演奏入力操作装置を別設することにより構成され、その操作に従って演奏データを入力することができる。

【0017】

表示部6は、これに接続されるディスプレイ（CRT、LCD等の表示器）10の表示内容や各種インジケータの点灯状態をCPU1からの指令に従って制御し、入力操作部5の操作に対する表示援助を行い、或いは、このシステムで生成された楽譜対応データに従って楽譜を表示するのに利用される。

【0018】

音源部7は、音源（ソフトウェアを含む）や効果付与DSPを含み、CPU1で処理された演奏データに対応する楽音信号を生成し、音源部7に接続されるサウンドシステム11は、D/A変換部やアンプ、スピーカを備え、音源部7からの楽音信号に基づく楽音を発生する。

【0019】

また、図示の通信I/F8は、ローカルエリアネットワーク（LAN）や、インターネット、電話回線などの一般用通信ネットワーク、或いは、MIDI用ネットワークに接続される各種インターフェースを一括して表わしており、サーバ等の他のコンピュータやMIDI機器などの種々の外部機器PDと各種情報を授受することができる。例えば、この装置に制御プログラムやデータが記憶されていない場合、通信I/F8を介して他のコンピュータPDから制御プログラム等をダウンロードすることができる。また、外部機器PDに、他の鍵盤楽器や、弦楽器、管楽器などの演奏入力操作装置を含む各種MIDI機器を用いて、実演奏データや自動演奏データを通信I/F8から取り込むこともできる。

【0020】

〔音符情報分離〕

この発明の一実施例による楽譜対応データ生成システムでは、楽譜対応データ生成プログラムに従い、演奏データ中の音符情報及びピッチベンドデータなどの表情付け情報を利用して楽譜対応データが生成される。図2は、この発明の一実施例による音符情報の分離ルールを説明するための図である。

【0021】

ここで、図2を用いてこの楽譜対応データ生成システムにおける楽譜対応データ生成の概要を極く簡単に説明しておく。このシステムでは、外部記憶装置4や外部MIDI機器PD又は入力操作部5などから演奏データが供給されると、この演奏データから音符情報Na及び表情付け情報Pbを検出し分析する。表情付け情報は、各音符情報Na_iの聴感上の音高を変化させるピッチベンドデータPb等の情報であり、各音符情報Na_i毎に、表情付け情報Pbに基づき所定の音高変化（半音）を与える音高変化点t₂, t₃を検出し、表情付け情報Pbに対応する奏法を認識する。そして、認識された奏法から元の音符情報Naを分離した方が楽譜に対応すると判断したときは、分離ポイントPsに従って聴感上の音高変化に合う複数の音符情報Nb₁, Nb₂; Nb₁', Nb₂'を生成する。分離ポイントPsは、最小時間幅 Δ に基づき、音高変化点t₂, t₃の内、分離後音符情報Nb₁, Nb₂; Nb₁', Nb₂'の音符長が短くなり過ぎない点に決定される。

【0022】

以下、図2につきより詳しく説明する。この例では、演奏データ中の元音符情報Naに対応するピッチベンドデータPbを分析し、個々の音符情報Na_iを複数の音符情報Nb₁, Nb₂; Nb₁', Nb₂'に分離する。図2の上2段は、演奏データ中のノートイベントデータ及びピッチベンドイベントデータ（列）が表わす元の音符情報Na及びピッチベンドデータPbによる音高（ピッチ）変化を図解的に示しており、また、横軸は時間tの経過を示す。この例では、最上段の或る元音符情報Na_iは、時間tsで開始し時間teで終了するノート“C3(60)”（括弧内はノートナンバを表わす）を示している。

【0023】

10

20

30

40

50

第2段目のピッチベンドデータ Pb は、縦軸に音高（ピッチ）を採ってピッチベンドの動きで表わしており、破線は半音毎のピッチベンド値に対応する。この例では、ピッチベンドデータ Pb は、音符情報 $Na i$ のノートイベントデータの開始乃至終了時間 $t_s \sim t_e$ 内において時間 t_1 で始まり時間 t_3 で終わるピッチベンドイベントデータ列に従って、時間 t_1 からピッチベンド値が変わり始め、時間 t_2 で+1半音に相当する値となり、時間 t_3 で+2半音に相当する値になってピッチベンド値の変化を終了する。

【0024】

つまり、音符情報 $Na i$ 及びピッチベンドデータ Pb により表わされる実質的な音高は、ノート開始時間 $t_s \sim$ ピッチベンド開始時間 t_1 の間は音符情報 $Na i$ の音高 C_3 (60) を示し、時間 $t_1 \sim t_3$ の間は、ピッチベンドデータ Pb により、時間 t_1 から聴感上の音高が徐々に増大し、時間 t_2 の音高 $C\#_3$ (61) を経て、時間 t_3 で音高 D_3 (62) に到達し、ピッチベンド終了時間 $t_3 \sim$ ノート終了時間 t_e の間は音高 D_3 (62) を維持する。

【0025】

この実施例では、このようなピッチベンドデータ Pb による音高変化に対して、次のようなルールを適用する：

- (1) 原則として、半音毎の音高変化点 t_2 , t_3 で音符情報 $Na i$ を分離する。
- (2) ただし、或る音高変化点 t_2 から次の音高変化点 t_3 までの時間が所定の最小音長時間幅（分離最小時間幅） Δ 以上でなければ、当該音高変化点 t_2 では分離しない。
- (3) 音高変化が終了する最後の音高変化点 t_3 については、調整可能な所定の移動時間幅（音符分離位置調整幅） δ だけ早めた時点で分離する場合がある。
- (4) なお、例外的に、一旦半音以上の最高音に上げた後再び元の音高に戻る「往復音高変化」の場合には、音符情報 $Na i$ を最高音に変更する。

【0026】

図2の例では、半音以上の音高変化があるので、まず、ルール(1)により、音符情報 $Na i$ から、第3段目に示すように、半音毎の音高変化点 t_2 , t_3 で分離候補ノート C_1 , C_2 を分離し、聴感上の音符に一致させる。なお、図示しないが、ルール(1)に従うと、ピッチベンドにより半音以上の音高変化を与えるスラー等のレガート表現の場合には、直ちに、その音高変化点で音符情報 $Na i$ が分離され、半音未満の音高変化を繰り返して与えるビブラート等のゆらぎ表現の場合には音符情報 $Na i$ は分離されない。

【0027】

次に、図2の第3段目に示すように、分離候補ノート C_1 の時間幅 $t_2 \sim t_3$ が所定の最小時間幅 Δ に満たない場合、このような時間幅の短い候補ノート C_1 が中間に介在することは、厳密には聴感上の音符と一致しないし、一般的な記譜方法とも一致しない。そこで、ルール(2)により候補ノート C_1 を削除する。また、これは、必要以上に音符を分割せず、以後の分離後音符情報の生成処理の簡単化につながる。なお、この最小時間幅 Δ は、テンポや曲想にも依存するのでユーザ設定可能とするが、例えば、4分音符＝120程度の速めの曲の場合、16分音符長程度に設定される。

【0028】

上述の処理により、図2の第4段目に示すように、音高変化点 t_3 を分離ポイント Ps として、音符情報 $Na i$ の音高に対応する音符情報 Nb_1 ($t_s \sim t_3$) と分離候補ノート C_2 ($t_3 \sim t_e$) に対応する音符情報 Nb_2 が得られる。この場合、更に聴感上の一致度を高めたいときには、ルール(3)を適用して、最終の音高変化点 t_3 よりもユーザ設定可能な所定の時間幅 δ だけ早めの時点 $t_3 - \delta$ に分離ポイント Ps を移動して、第5段目（最下段）に示すような分離後音符情報 Nb_1' , Nb_2' を得ることができる。

【0029】

なお、往復音高変化の場合（図示せず）、聴感上の音符は一般的には1つの音符に聞こえ、その音高は最高の部分の音高の音符として知覚される傾向にあるので、ルール(4)に従って最高の音高を採用すると、聴感上の音符と一致した楽譜対応データが得られる。

【0030】

〔楽譜対応データ生成の動作フロー〕

10

20

30

40

50

図3は、この発明の一実施例による楽譜対応データ生成の動作例を表わすフローチャートであり、図4は、この発明の一実施例による音符分離処理例を表わすフローチャートである。まず、図3の楽譜対応データ生成動作について説明する。

【0031】

パネル操作子5のユーザ操作により楽譜対応データ生成の開始が指示され、この楽譜対応データ生成プログラムに基づく動作フローがスタートすると、CPU1は、ユーザ設定操作に従って初期設定を行う（ステップS1：以下においては、ステップ記号“S…”，“SP…”単独で該当ステップを表わす）。この初期設定では、例えば、この楽譜対応データ生成に使用する演奏データを選択設定したり、分離後音符情報の最短音長を決める分離最小時間幅 Δ 、音符分離位置の移動量を決める音符分離位置調整幅 δ などを設定する。 10

【0032】

CPU1は、次いで、分離後音符情報即ち楽譜対応データ列を格納するバッファDataList、及び、演奏データ中の音符情報を格納するバッファNoteをRAM2上にセットして、これらを初期化する（S2）（以下、記号DataList、Note単独でこれらのバッファを表わす。）。続いて、設定された演奏データをRAM2上にロードし（S3）、ロードされた演奏データから、順次、音符情報Naを1音ずつNoteに読み込む（S4）。 10

【0033】

次に、Noteに音符情報Na iがあるか否かを判断する（S5）。ここで、音符情報Na iがあるときには（S5→YES）、ロードされた演奏データより、音符情報Na iに対応するピッチバンドデータPbを抽出し（S6）、更に、抽出されたピッチバンドデータPbに基づいて、ピッチバンドに関する奏法を検出する（S7）。なお、対応するピッチバンドデータPbがない場合には、これら抽出及び検出処理（S6、S7）をスルーする。 20

【0034】

奏法検出ステップ（S7）で検出される奏法には、スラー、チョーキング、スライド、装飾的チョーキング、ビブラート、往復音高変化、等々の各種奏法がある。例えば、ピッチバンドデータPbによる音高が、音符情報Na iのノートイベント期間の途中で急に1半音以上変化してその後安定していたらスラーと判断し、また、図2の第2段目のように、途中で1半音以上曲線的に変化してその後安定していたらチョーキングと判断する。

【0035】

続いて、抽出及び検出ステップ（S6、S7）での処理結果を判定するステップ（S8～S10）に移り、まず、音符情報Na iに対応するピッチバンドデータPbがあったかを判定し（S8）、ピッチバンドデータPbがあったときは（S8→YES）、当該ピッチバンドデータPbにより所定のピッチ変化があったかを判定する（S9）。ここで、所定のピッチ変化即ち1半音以上の音高変化があったときは（S9→YES）。さらに、検出された奏法が音符情報Na iを分離すべき奏法であったか否かを判定する（S10）。 30

【0036】

奏法判定ステップ（S10）で、1半音以上の音高変化のあるスラー、チョーキングなどの場合のように、分離すべき奏法であったと判定したときは（S10→YES）、図4の「音符分離処理」のステップ（S11）に進んで、元の音符情報Na i及びピッチバンドデータPbの内容に従って当該音符情報Na iを分離してDataListに記憶する。 40

【0037】

一方、奏法判定ステップ（S10）で、往復音高変化奏法などの場合のように、1半音以上の音高変化があっても分離すべき奏法ではないと判定したときは（S10→NO）、Noteの音符情報Na iと、検出された奏法に関する情報とをDataListに記憶する（S12）。この場合、奏法によっては音符情報Na iの音高を書き換える。例えば、上例の往復音高変化奏法の場合、音符情報Na iの音高を、聴感上知覚されるピッチ変化中の最高値に対応する音高に変更する。

【0038】

また、対応するピッチバンドデータが無いとき（S8→NO）、或いは、例えば、音高変化が半音に満たないビブラート奏法のように、所定のピッチ変化がないときは（S9→NO） 50

）、Noteに読み込んだ元の音符情報N a iをそのままDataListに記憶する（S 1 3）。後者の場合、ピッチベンドデータP bから得た奏法をDataListに付記することができる。

【0039】

このようなDataList記憶（更新）処理（S 1 1～S 1 3）の後には、音符情報読み込みステップ（S 4）に戻り、次の音符情報N a iをNoteに読み込み、Noteに音符情報N a iがある間は（S 5→YES）、上述の動作（S 6～S 1 3）を繰り返す。

【0040】

演奏データが終了しNoteのデータが空（音符情報無し）になると（S 5→N0）、ピッチベンド以外の他の観点から、必要に応じて、DataListを更に楽譜対応化させる（S 1 4）。この楽譜対応化には、例えば、装飾音符の付加などがある。そして、DataListを外部記憶装置5に出力し（S 1 5）、楽譜対応データとして保存する。或いは、DataListを表示部6に出力して楽譜などで表示してディスプレイ10でユーザが確認した後、外部記憶装置5に出力して（S 1 5）、この楽譜対応データ生成フローを終了する。

【0041】

図4は、図3のステップS 1 1で実行される音符分離処理の一例である。この音符分離処理は、前半の「分離ポイントの絞込み」（S P 1～S P 3）と後半の「分離した音符情報の生成」（S P 4～S P 8）の処理に二分される。音符分離処理フローが開始すると、まず、分離ポイント絞込みの第1ステップ（S P 1）で、バッファPBPointListを初期化する。このバッファPBPointListは、Noteに読み込まれている元の音符情報N a iを分離するためのポイント要素（ピッチベンド値及び時間的な位置）をリスト形式で記憶するバッファである（以下、記号PBPointList単独でこのバッファを表わす）。

【0042】

次に、音符情報N a iに対応するピッチベンドデータP bから、ピッチベンド値が変わり始める直前〔例えば、図2の時点t1（以下、必要に応じて、図2の例を括弧書きで補足する）〕のピッチベンド値と位置を取得し、これをポイント要素〔P1〕としてPBPointListの先頭に記憶し、その後、ピッチベンドの変化が終了するまで〔時点t3まで〕、音高が半音変化する毎に〔t2, t3〕そのときのピッチベンド値と時間位置を取得し、これをポイント要素〔P2, P3〕として、順次、PBPointListに記憶して行く（S P 2）。

【0043】

そして、PBPointList内の先頭〔P1〕と末尾〔P3〕以外に記憶されたポイント要素〔P2〕について、設定された分離最小時間幅 Δ （例えば、16分音符長）に従って、直後の要素〔P3〕と時間位置が近い（ Δ 以下）もの〔P2〕を削除する（S P 3）。このようにして、音符情報N a iの分離ポイントP sを絞り込むと〔P s=P1, P3〕、分離した音符情報の生成処理（S P 4～S P 8）に進む。この分離音符情報の生成処理は、PBPointList内の各要素について順番に繰り返し行われ、以下のように実行される。

【0044】

分離音符情報の生成処理の第1ステップ（S P 4）では、PBPointList内の要素〔P1, P3〕の内、処理対象とするポイント要素の時間位置を読み出し開始位置Startとし、その次のポイント要素の時間位置を読み出し終了位置Endとする。ここで、処理対象のポイント要素が先頭〔P1〕の場合は、音符情報N a iの開始時間〔ts〕を読み出し開始位置Startとし、対象のポイント要素が末尾〔P3〕の場合には、音符情報N a iの終了時間〔te〕を読み出し終了位置Endとする。図2の例では、先頭要素P1を処理する場合は、ノートN a iの開始時間t sを読み出し開始位置Startとし、次の要素P3の時間位置t 3を読み出し終了位置Endとする。また、末尾要素P3を処理する場合には、当該要素P3の時間位置t 3を読み出し開始位置Startとし、ノートN a iの終了時間t eを読み出し終了位置Endとする。

【0045】

次に、PBPointList内において処理対象とするポイント要素〔P1又はP3〕のピッチベンド値を読み出して、音符情報N a iに対応する音高に調整してこの音高をキーナンバで表わした音高値データKeyNumber〔60又は62〕を取得する（S P 5）。なお、先頭の要素〔P1〕のピッチベンド値は、音符情報N a iの音高〔C3〕から変化していないものとする。続

いて、当該処理対象要素〔P1又はP3〕の読出し開始位置Start、読出し終了位置End、音高値KeyNumber及び音符情報Na iのベロシティ値Velocityにより、分離された新たな音符情報〔Nb1又はNb2〕を生成する（SP6）。

【0046】

さらに、生成した新たな音符情報〔Nb1又はNb2〕の時間〔ts又はt3のタイミング〕及び長さ〔ts～t3又はt3～te〕を、音符分離位置調整幅 δ の設定状況に基づき調整する（SP7）。例えば、この調整幅 δ が有効な場合、調整幅 δ だけ早い時間で音符〔Nb1又はNb2〕を分離する。そのため、先頭になる音符は長さが短くなる〔Nb1→Nb1'〕。

【0047】

このようにして生成した音符情報〔Nb1'又はNb2'〕と奏法に関する情報をDataListに記憶して（SP8）、当該ポイント要素〔P1又はP3〕に対する分離音符情報生成処理を終了し、次の要素があれば、順次、次の要素について上述の分離音符情報生成処理（SP4～SP8）を行い、次の要素に対応して分離された新たな音符情報を生成する。そして、全要素の分離音符情報生成処理を実行し終わると、この音符分離処理フローを終了する。

【0048】

〔種々の実施態様〕

以上、この発明の好適な一実施例について説明したが、これは単なる一例であって、この発明は、その精神を逸脱しない範囲で種々の変更が可能であり、種々の態様で実施することができる。例えば、生成した楽譜対応データは楽譜の表示に使用してもよいし、演奏ガイドに使用してもよい。また、曲データ中の聴感上も含めた音高の動きを分析するためなどに使用することもできる。

【0049】

ピッチベンドを使った奏法で音符を分離するタイプの例には、チョーキングやスライド、スラー等がある。また、ピッチベンドによる表情付けが和音状態の複数音符情報に付加されている場合、各音符情報それぞれについて分離処理を行ってもよい。音符分離の基準時間幅 Δ については、実施例の値や態様に限らない。デフォルト固定でもよいしユーザ設定可能であってもよい。或いは、曲に設定された再生テンポによって変わってもよい。

【0050】

実施例では、ピッチベンドデータ等による音高変化が半音レベルと交差するときのレベルをノート分離の候補とするようにしているが、厳密に半音でなくてもよい。つまり、実際の演奏データでは、音高変化が聴感上は半音に交差しているように聞こえるが、データとしては交差する直前であるという場合がありますので、このような場合に対応して、音高変化が半音レベルと交差することを評価する際に、マージン（例：最大0.2半音程度）を設定し、完全に半音を交差する少し前に半音交差ありと評価する処理が好ましい。

【0051】

一往復音高変化奏法のように1つの音符情報の中でピッチベンドによる音高変化が往復する奏法については、この音高変化が所定時間（例えば、16分音符長など）以内に往復するような場合は、音符の分離は行わずに、音高を聴感上の最高音高に変更するが、音高変化が或る程度長い音符長になる場合には、分離ルールに従って分離した方がよい。さらに、音高変化を伴う表情付け情報には、ピッチベンドデータ以外にも、例えば、ブレスコントロール情報なども制御対象になる場合がある。また、表情付けに応じた奏法は、ピッチベンドなどのイベント列より検出してもよいし、ユーザが別途入力設定したもの、或いは、別システムにて予め判別されて記憶されていたものでもよい。

【0052】

元になる演奏データは、記録された曲データでもよいし、ユーザが入力したものでもよい。実施例ではピッチベンドで表情付けされた音符情報に対する処理を中心としているが、実際には、アルペジオなどとして複数の音符をまとめて扱えるかどうかの判断や、音量情報の変化を見て強弱記号の付いた音符情報に対する処理などを行う必要がある。演奏データは、実施例では1トラックとしているが、複数トラックあってもよい。複数トラックある場合は、それぞれのトラックに対し処理を行う。

生成される楽譜対応データには、分離した音符情報の他に、元々の表情データであったピッチベンド情報をまとめた形で音楽（奏法）記号データを生成し含ませてもよい。もしくは、分離した音符情報に、例えば属性情報の形で、音楽（奏法）記号の始点／終点などを表わす情報を付加しておいてもよい。

【図 1】 この発明の一実施例による楽譜対応データ生成システムのハードウェア構成例を示すブロック図である。

10

【図 4】 この発明の一実施例による音符分離処理例を表わすフローチャートである。

$N a ; N a i$ 演奏データ中の元の音符情報（ノートイベント、音符データ）、
 $t s, t e$ 個々の音符情報 $N a i$ の開始（ノートオン）及び終了（ノートオフ）時間、
 $P b$ ピッチベンドデータ（ピッチベンドイベント、ピッチベンド情報）、
 $t 1, t 2, t 3, \dots$ ピッチベンドの開始時間及び半音変化毎の時間、
 $C 1, C 2, \dots$ 分離候補音符、
 $P 1, P 2, P 3, \dots$ ポイント要素（ポイント、要素）、
 Δ 分離最小時間幅（最小音長時間幅）、
 $P s$ 分離ポイント、
 $N b ; N b 1, N b 2, \dots ; N b 1', N b 2', \dots$ 分離後の音符情報、
 δ 音符分離位置調整幅（移動時間幅）。

20

システムのハードウェア構成例

元の音符情報 Na

ピッチベンドデータ Pb
(ピッチベンドの動き)

時間 t

分離候補 C1

分離候補 C2

聴感上 D3(62)

聴感上 C#3(61)

聴感上 C3(60)

ts

t1

t2

t3

te

Δ

候補 C1

候補 C2

P1

P2 (x)

P3 (Ps)

Nb1

Nb2

C3(60)

D3(62)

Nb1'

Nb2'

C3(60)

D3(62)

δ

分離した音符情報 Nb

分離する時間を
設定値よりだけ早めた場合

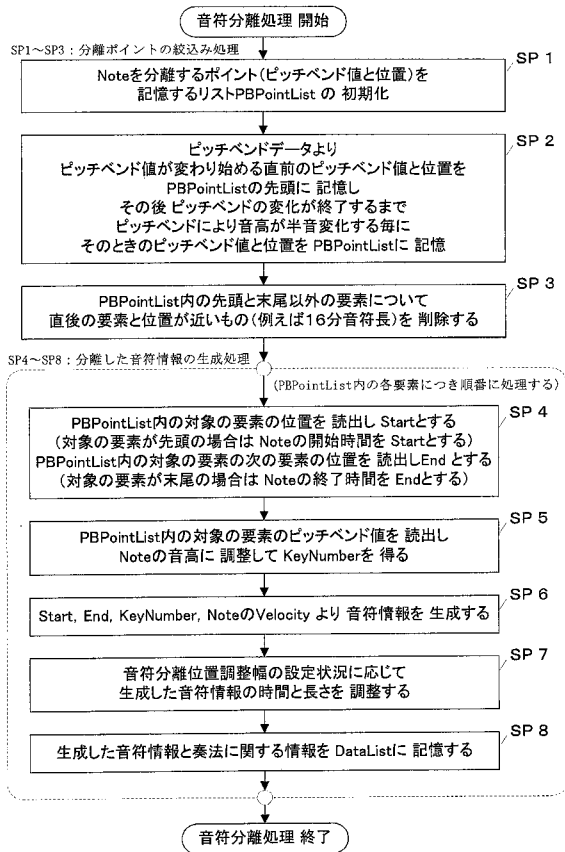
音符情報分離例

```

graph TD
    Start([楽譜対応データ生成 開始]) --> S1[S1 ユーザ設定]
    S1 --> S2[S2 楽譜対応データ列DataList  
演奏データ中の音符情報Note  
の初期化]
    S2 --> S3[S3 演奏データのロード]
    S3 --> S4[S4 音符情報(Note)の読み込み]
    S4 --> S5{S5 Noteあり?}
    S5 -- NO --> S14[S14 他の観点から必要に応じて  
DataListを更に楽譜対応化する  
(装飾音符など)]
    S5 -- YES --> S6[S6 演奏データより  
Noteに対応するピッチベンド  
データを抽出する]
    S6 --> S7[S7 ピッチベンドデータより  
奏法を抽出する]
    S7 --> S8{S8 ピッチベンドデータあり?}
    S8 -- NO --> S13[S13 NoteをDataListに記憶]
    S8 -- YES --> S9{S9 ピッチの変化あり?}
    S9 -- NO --> S13
    S9 -- YES --> S10{S10 Noteを分離すべき奏法?}
    S10 -- NO --> S13
    S10 -- YES --> S11[S11 音符分離処理]
    S11 --> S12[S12 Noteと奏法に関する情報を  
DataListに記憶]
    S12 --> S13
    S14 --> S15[S15 DataListを出力]
    S15 --> End([楽譜対応データ生成 終了])

```

【図 4】



フロントページの続き

【要約の続き】