2007－MUS－71 (3)
2007／8／1

くし形フィルタに基づく自動採譜システムの実現

坂内 秀幸, 夏井 雅典, 田所 嘉昭
(豊橋技術科学大学)

減算処理のみで実現できるくし形フィルタ ($H(z) = 1 - z^{-N_{i,p}}$) に基づく，ビートトラッキングおよび音価割当を行うシステムについて提案する．システムに入力する音楽信号は，MIDI を使用して作成した，2 和音以下，オクターブ 4 のみで構成される楽曲を，6 種類の楽器で演奏したものである．さらに，本研究グループで提案されているくし形フィルタに基づく音高推定アルゴリズムと本システムを組み合わせることで構築した自動採譜システムについて述べ，6 種類の楽曲を対象とした推定実験の結果から，本システムの性能を評価する．結果として，6 種類の楽曲の音楽信号において，本システムは約 99% 正しい楽譜の作成が可能であることを確認した．

Realization of Automatic Musical Transcription System Based on Comb Filters

Hideyuki Sakauchi, Masanori Natsui, Yoshiaki Tadokoro
Toyohashi University of Technology

We propose a novel beat tracking method and a phonetic value assignment method based on comb-filter ($H(z) = 1 - z^{-N_{i,p}}$). The input musical signal are two or less chords in octave 4 and these musical data are created using MIDI source of six musical instruments. Furthermore, we describe an automatic musical transcription system combining the proposed method and the pitch estimation algorithm that has been proposed by our study group. As a result, we could show that the accuracy of the system is about 99%.

1. まえがき

音楽において，楽曲を楽譜として記述することを採譜という．採譜を行うには，音楽的な知識や経験が必要であるが，計算機で自動的に採譜を行うことができれば，音楽的な知識や経験のない人にでも楽譜を作成することができる．特に，Jazz などの即興演奏や，民族音楽など，そもそも楽譜の存在しない音楽に対する自動採譜の需要が高まっている．[1)~3)]

本研究グループは今まで，自動採譜に必要不可欠な技術である音高推定に関する研究を主に進めてきており，くし形フィルタに基づいた推定手法を提案し，良好な結果を得ている．[3)] しかし，音高推定だけでは，各時刻における音高の検出によって，奏譜形式の結果が得られるだけであり，音符としての長さはわからない．そこで本稿では，自動採譜におけるテンポ，音符の長さの決定などにおいて必要な技術である，ビートトラッキングおよび音価割当の手法を提案する．さらに，既存の音高推定アルゴリズムを組み合わせることで，自動採譜システムの実現を図る．

2. 音高推定

2.1 くし形フィルタによる音高推定手法

本研究グループで提案されてきた音高推定アルゴリズム[3)] は，楽音の周波数スペクトルと，くし形フィルタの性質を利用した手法により実現されており，現在 3 和音までの音高推定において，90% 以上の精度での推定が可能となっている．楽音の定常部分における周波数スペクトルは，図 1(a) に示すように，基本周波数 $f_{i,p}$ と，その倍音周波数に等間隔にピークを有する調波構造の性質を持つ特徴がある．また，本研究で使用するくし形フィルタとは，式 (1) で表される伝達関数を持ち，ノッチ特性を示す．このくし形フィルタに，音楽信号 $x(n)$ を入力することで得られる出力 $y_{i,p}(n)$ は，式 (2) で与えられる．

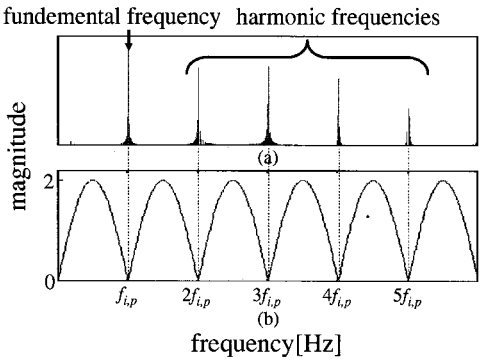

図 1 (a) 楽音の周波数スペクトル（piano, C_4），(b) くし形フィルタの振幅特性

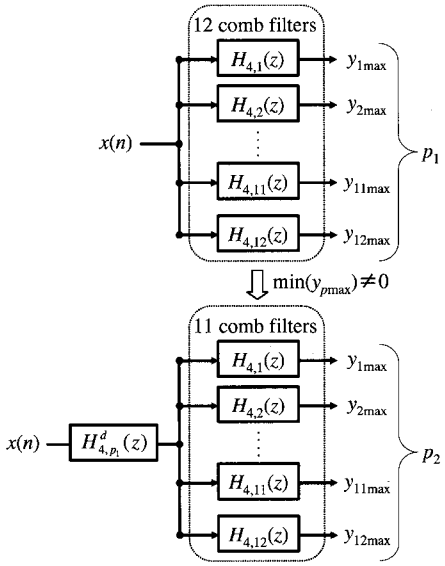

図 2 音高推定アルゴリズム

$$H(z) = 1 - z^{-N_{i,p}} \tag{1}$$

$$y_{i,p}(n) = x(n) - x(n - N_{i,p}) \tag{2}$$

$$N_{i,p} = \left[\frac{f_s}{f_{i,p}}\right] \quad [\,] : \text{integer by rounding} \tag{3}$$

ここで，i はオクターブの高さを表しており，p は音名 C,C$^{\#}$,D,D$^{\#}$,E,F,F$^{\#}$,G,G$^{\#}$,A,A$^{\#}$,B にそれぞれ 1,2,...,12 を割り当てた数である．また，式 (3) の $N_{i,p}$ は，音高 (i,p) におけるくし形フィルタの遅延数を表し，サンプリング周波数 f_s と，楽音の基本周波数 $f_{i,p}$ によって与えられる．くし形フィルタ $H(z)$ の周波数応答（振幅特性）を導出すると，式 (4) のように $N_{i,p}$ の値によって定まる振幅特性となる．具体的には，図 1(b) に示すようなくし形の形状となり，0Hz からサンプリング周波数 f_sHz までの間に $N_{i,p}+1$ 個の等間隔な零点を持つ特性となる．

$$|H(e^{j\omega})| = \left|2\sin(\frac{\omega N_{i,p}}{2})\right| \tag{4}$$

従って，楽音スペクトルのピーク部分と，くし形フィルタの零点が同じ位置にくるように $N_{i,p}$ の値を決めれば，そのくし形フィルタに通した楽音の基本周波数と倍音周波数を全て消すことができる．この性質を利用し，特定の音高が消えたかどうかを検出することで音高推定を行うことができる．通常，短時間の矩形窓で波形を切り取り，窓を少しずつシフトさせながら音高推定を行うことで，音高の時間的な変化を見る．

2.2 音高推定アルゴリズム

次に，音高推定アルゴリズムについて説明する．本アルゴリズムは原理的に任意の和音に対して適用可能である．また，オクターブ推定を行うことも可能であるが，ここでは入力される音楽信号が，オクターブ $i=4$ (C_4:261.6Hz ～B_4:493.9Hz）の高々2 和音のみで構成されることを想定した場合について述べる．

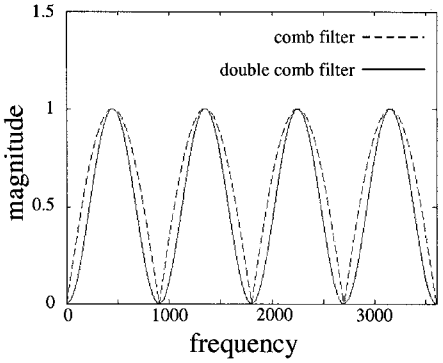

図 3 くし形フィルタと 2 重くし形フィルタ

図 2 のように，C_4～B_4 の音高を除去する 12 個のくし形フィルタを並列接続し，音楽信号を入力することを考える．このとき，出力信号の最大振幅 $y_{pmax} = \max(|y_{4,p}(n)|)$ を最も小さくするフィルタ $H_{4,p_1}(z)$ は，入力信号のスペクトルを最も多く除去していることになる．従って，入力信号に音高 $(4,p_1)$(オクターブ 4 の p_1 音) が含まれていた可能性が高いといえる．音高 $(4,p_1)$ を 1 音目とし，次に，2 音目があるかどうかを 2 重くし形フィルタ $H^d_{4,p}(z) = (1-z^{-N_{4,p}})^2$ によって判定する．この 2 重くし形フィルタは，図 3 のように零点付近の減衰域に幅を持った振幅特性となり，2 重くし形フィルタに通して得られる出力 $y^d_{4,p_1}(n)$ は，より確実に指定した音高を除去した波形となる．2 重くし形フィルタの出力信号 $y^d_{4,p_1}(n)$ において，式 (5) を用いて入出力のパワー比 r_{io} を計算し，r_{io} の値が閾値 T_{zero} 以下になった場合を零出力とする．なお，ここで N_T は総和を計算する区間である．

$$r_{io} = \frac{\sum_{n_T=0}^{N_T-1} y^d_{4,p_1}(n+n_T+N_{4,p})^2}{\sum_{n_T=0}^{N_T-1} x(n+n_T)^2} \tag{5}$$

この時点で零出力となった場合は，入力信号が単音であったことがわかる．零出力とならなかった場合には，図 2 下部に示すように，1 音目の音高 $(4,p_1)$ を除去する 2 重くし形フィルタ $H^d_{4,p_1}(z)$ を，11 個並列接続されたくし形フィルタの前段に接続する．この 11 個の出力信号から，前と同様に最大振幅 y_{pmax} を最小となるくし形フィルタを特定し，音高を推定する．ここで出力信号が零出力となれば，入力信号が 2 和音であったことがわかる．本稿で扱う楽曲は 2 和音以下の楽曲であるため，ここで出力信号が零出力とならなかった場合には，音高は推定不能であるとする．

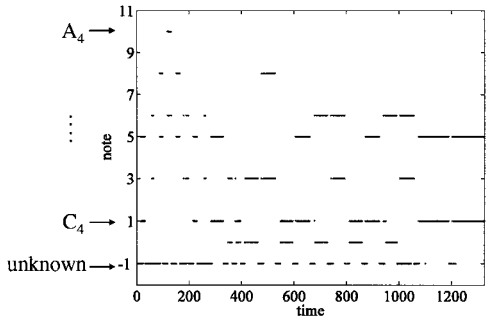

図 4 piano による簡単な演奏の音高推定結果

2.3 音高推定アルゴリズムの適用例

入力する楽曲の音楽信号として，オクターブ 4 の 2 和音以下の簡単な演奏を行うテンポ 100 の MIDI データを用意し（以後，楽曲 A と呼ぶ．楽譜は図 12 を参照），Piano での演奏をサンプリング周波数 $f_s = 44100$Hz で録音した．また，N_T を 441 点（10ms）とし，最大振幅を 1 に正規化した入力信号に対して $T_{zero} = 0.01$ と設定し，零出力を検出した．音高推定の適用結果の例を図 4 に示す．図 4 において，縦軸は音高を表しており，零出力により単音のみと判定された部分は，2 音目を 0 としている．また，零出力検出の結果，推定不能となった場合は，1 音目，2 音目を共に-1 として出力している．横軸は時間であるが，ここではサンプル数 $W_w = 800$ の矩形窓で切り抜き，シフト幅 $W_s = 400$ ずつシフトさせて音高推定を行った際のインデックスである．図から短時間ごとの音高が検出されていることがわかる．

以上のように，本システムを用いることで，和音を対象とした音高推定が可能となる．ただし，本システムによって得られるのは，図 4 に示すような，いわゆる奏譜形式と呼ばれるものであり，実際の楽譜形式として出力するためには，ビートトラッキングによるテンポの推定や，音価割当などを行う必要がある．このことを踏まえ，次章ではくし形フィルタを用いたテンポの推定手法について提案する．

3. ビートトラッキング

3.1 テンポとビート長の関係

テンポとは，楽曲の演奏などにおいて，1 分間に何拍が刻まれるかを表す値である．例えばテンポ 120 といえば，1 分間に 120 拍（1 拍 0.5s）が刻まれることになる．また，1 拍の時間長をビート長と呼ぶ．ビート長（*beat*）とテンポ（*tempo*）には式 (6) のような関係がある．従って，ビート長を検出することができれば，そこからテンポを算出することができる．なお，本稿で入力信号として用いる楽曲のテンポは，一般によく用いられる 60〜185 の範囲にあるものとする．すなわち，ビート長の範囲は 0.324〜1s となる．

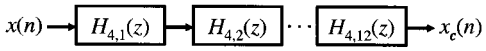

図 5 12 個縦続接続のくし形フィルタ

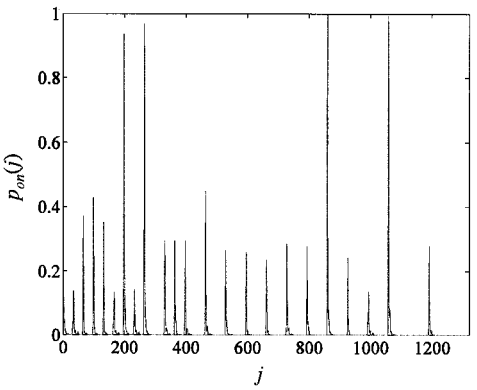

図 6 発音開始パルス

$$tempo = \frac{60}{beat} \tag{6}$$

3.2 発音開始パルス

テンポ検出の手掛りとなるパラメータとして，ここでは発音開始パルス $p(j)$ を抽出することを考える．発音開始パルスとは，楽曲中に含まれる楽音が発音された瞬間に，パルス状のピークを持つ波形を意味する．発音開始パルスに関する音楽的知識[4)5)] によると，発音開始パルスのピークはビート時刻に存在することが多い．あるいは，ビート時刻でなくとも，ビート長の半分，さらにその半分に存在する可能性が高いことが知られている．従って，発音開始パルスのピークの間隔や周期性を検出することにより，テンポを算出することができると考えられる．本稿では，くし形フィルタにより発音開始パルスを検出する手法を提案する．

3.2.1 検出手法：Comb Filter(CF) 法

2.1 節で述べたように，くし形フィルタを用いて所望の音高を除去することで，音高推定を行うことができる．しかし，楽音において除去可能なのは，調波構造を持つ定常部分だけであり，発音開始直後のような非調波構造を持つ部分においては，楽音を完全に除去することができず，波形が残る．従って，くし形フィルタに通した後に残っている箇所は，全て発音開始部分に相当することになる．この性質は，音高が推定できないという点では問題となるが，発音開始パルスを検出する上では，逆に利用することができる．

以前に示した楽曲 A を，入力信号 $x(n)$ とする．C_4〜B_4 を除去するくし形フィルタを，それぞれ $H_{4,1}(z)$〜$H_{4,12}(z)$ とし，この 12 個のくし形フィルタを，図 5 のように縦続接続して，$x(n)$ を入力する．得られる出力 $x_c(n)$ において，短時間ごとのパワー $p_{on}(j)$ を，窓幅 W_w，窓

のシフト幅 W_s として，式 (7) を用いて求めると，楽音の定常部分が全て除去されて，発音開始部分のみがパルス状に残る．

$$p_{on}(j) = \frac{\sum_{n=jW_s}^{jW_s+W_w-1} \{x_c(n)\}^2}{W_w} \quad j = 1, 2, 3, \cdots \qquad (7)$$

これを発音開始パルスとし，ピークは発音開始部分を表していることから，テンポを検出する手がかりとなる．図 6 に，楽曲 A の発音開始パルスを示す．

3.3 発音開始パルスに基づくテンポ検出

CF 法によって得られた発音開始パルスからビート長を求めるために，発音開始パルスのピークがビート時刻となる可能性が高いこと（発音開始パルスに関する知識）を利用する．矩形窓のシフト回数を W_f として，式 (8) から自己相関関数 $R(k)$ を求めると，いくつかのピークを持った波形となる．これらは発音開始パルスの周期性によるピークであり，発音開始パルスに関する音楽的知識から，ビート時刻上の発音開始パルスによって生じたピークの極大値は，他のピークよりも大きくなる可能性が高い．従って，ビート時刻上のパルスによって生じた $R(k)$ のピークの最大値におけるインデックス k_{peak} から，式 (9) を用いてビート長を求めることができる．このとき，テンポの範囲およびビート長の範囲が決められているため，ビート長を求めるためのピークの範囲も，式 (9) から逆算することで求まる．本稿においては，ビート長は前述した通り 0.324〜1s であるので，k_{peak} の範囲は 36〜110 に限られる．従って，k =36〜110 の範囲内で $R(k)$ が最大となる点 k を k_{peak} とし，k_{peak} から求めたビート長が，正しいビート長である可能性が高いことになる．$beat$ を求めたら，式 (6) からテンポを算出することができる．ビート長は，後述する音価割当で使用する．楽曲 A においては $k_{peak} = 66$ となり，$W_s = 400, f_s = 44100$Hz より，$beat \simeq 0.599$s, $tempo \simeq 100$ となった．

$$R(k) = \frac{1}{W_f - k} \sum_{j=0}^{W_f-1-k} p_{on}(j) p_{on}(j+k) \qquad (8)$$

$$beat = \frac{k_{peak} \times W_s}{f_s} \qquad (9)$$

3.4 音価割当・音高割当

3.3 節より，楽曲 A における 1 拍の長さが $beat$=0.6s と求まったので，この値から音符の長さを割り当てる．この操作を音価割当という．

m 番目の発音開始パルスのピークの位置を $loca(m)$ とする．$loca(m)$ は音符の開始点を表す時刻になり，$loca(m+1)$ は $loca(m)$ に発音を開始した音符の終了時刻とみなす．本稿では 2 和音以下の楽曲を用いるため，$loca(m)$ から $loca(m+1)$ の間には，音符が 1 つあるいは 2 つ存在することになる．この区間の長さを式 (10) のように 1 拍の長さ k_{peak} で割り，値を量子化することで，その音符が何拍であるかを算出できる．本稿では，音符の長さを 0.5 拍刻みの拍数に量子化するために，$leng(m)$ を 0.5,1.0,1.5,2.0,$\cdots$ のうち，最も近い値に近似させることで拍数を割り当てる．

$$leng(m) = \frac{loca(m+1) - loca(m)}{k_{peak}} \qquad (10)$$

次に，音価割当を行った音符において，音高推定結果から音高割当を行う．前述したように，実際は m 番目の発音開始パルスから $m+1$ 番目の発音開始パルスまでの区間には 1 つあるいは 2 つの音符が含まれるが，音高推定結果に誤推定が含まれれば 3 つ以上の音が含まれてしまうことがある．そこで，この区間において，最も多く推定された音高を割り当てる．すなわち，図 8 に示すように，0 と 6 が最も多ければ，その区間は F_4 が単音で演奏されているとし，3 と 6 が最も多ければ，その区間には D_4 と F_4 の 2 和音が存在するとする．この手法を用

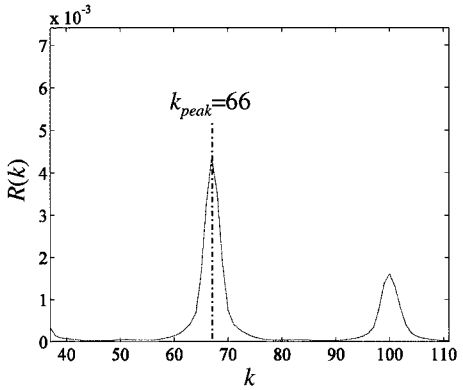

図 7 自己相関関数

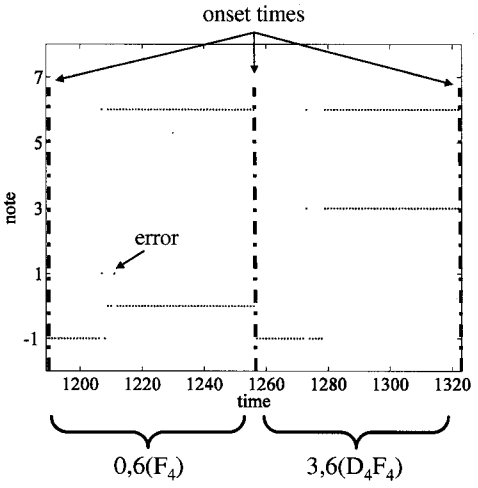

図 8 音高割当

いることで，音高推定に多少の誤推定が生じても，音高を正しく割り当てることができる．

4. 音符の連結処理 (connect)

ここまでの段階で，各音符の音高と長さがわかったことになる．しかし，発音開始パルスのピーク部分は，必ず音符の開始であると認識するため，図 9(a) のような演奏と，図 9(b) のような演奏は，どちらも図 9(b) のように認識されてしまう．そこで，図 9(a) のような演奏において，実際には音符が繋がっているということを検出し，音符を繋げる操作が必要である．この操作を以後 connect と呼ぶことにし，図 9(a),(b) のように同じ音高が続く場合の接続点を connect point と呼ぶことにする．図 9(b) のような演奏である場合は，音符が繋がっていないということを検出し，connect を行わないようにする必要がある．

4.1 STFT に基づく connect の検出手法

まず，入力波形において，connect point の周辺部分のみを抜き出す．例えば図 9 の演奏の場合，C_4 の音に切れ目があるかどうかで connect を行うかどうかを判定する必要があるが，遅れて発音される E_4 の音が検出の妨げとなる。そこで，抜き出した波形において，遅れて発音される E_4 の音をくし形フィルタで除去しておく．発音開始部分は完全に除去することができないが，定常部分においてはほぼ E_4 の音を除去することができる．

次に，くし形フィルタで E_4 を除去した波形において，STFT(Short-Time Fourier Transform: 短時間フーリエ変換) を適用する．C_4 の基本周波数は既知であるので，STFT の結果から，基本周波数および倍音周波数におけるスペクトルの時間的な変化を見る．ここでは，時間 l における基本周波数および基本周波数の 2 倍にあたる倍音周波数において，スペクトルの大きさをそれぞれ $s_1(l)$, $s_2(l)$ とし，この 2 つを用いることにする．connect point で再び C_4 が発音されている場合には，connect point の前後でスペクトルの急激な変動が生じ，発音していない場合には，スペクトルの変動が比較的小さくなる．この

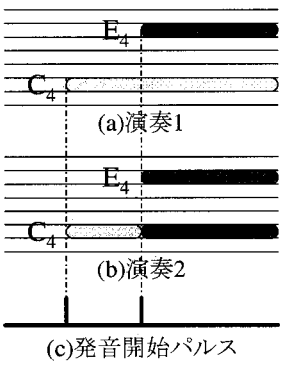

図 9 和音の演奏における問題点

両者を，分散の値によって比較することを考える．両者のスペクトルの時間的な変動において，徐々に減衰するような緩やかな変動は検出せず，急激な変動だけを検出するために，式 (11),(12) を用いる．

$$s_{h1}(l) = \max\{(s_1(l), s_1(l+1)\} - \min\{s_1(l), s_1(l-1)\} \quad (11)$$

$$s_{h2}(l) = \max\{(s_2(l), s_2(l+1)\} - \min\{s_2(l), s_2(l-1)\} \quad (12)$$

式 (11),(12) は，振幅の増加のみを検出し，減少する箇所は 0 として出力されるため，発音による音圧の増加のみを抽出することができる．connect point において発音がされていない場合は，図 10 のように，$s_{h1}(l)$, $s_{h2}(l)$ の両方がほとんど変動しないのに対し，発音されている場合は図 11 のように，どちらか一方，あるいは両方が大きく変動する．従って，この 2 つの波形 $s_{h1}(l)$, $s_{h2}(l)$ において，式 (13),(14) から分散 σ^2_{h1},σ^2_{h2} を計算し，両方が閾値 T_{con} より小さい場合は発音されていないとする．また，どちらか一方，あるいは両方が T_{con} 以上となる場合は発音されているとする．なお，ここで $\bar{s_{h1}}$, $\bar{s_{h2}}$ はそれぞれ $s_{h1}(l)$, $s_{h2}(l)$ の平均値を表す．

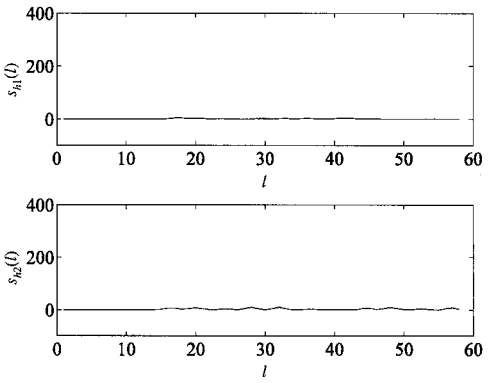

図 10 connect point において発音のない場合の $s_{h1}(l)$,$s_{h2}(l)$

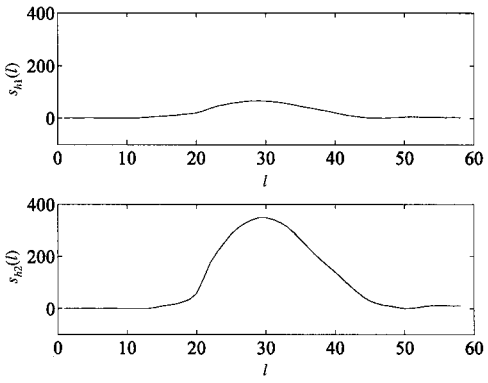

図 11 connect point において発音のある場合の $s_{h1}(l)$,$s_{h2}(l)$

表 1　各楽器における正答率

楽器	Piano	Guitar	Violin
音高正答率 (%)	100	94.4	100
拍数正答率 (%)	100	91.7	100
テンポ正誤	○	○	○
connect 正答率	7/7	6/7	7/7
楽器	Flute	Hamonica	Trumpet
音高正答率 (%)	100	100	100
拍数正答率 (%)	100	100	100
テンポ正誤	○	○	○
connect 正答率	7/7	7/7	7/7

$$\sigma_{h1}^2 = \frac{1}{l}\left[\sum_{l}\{s_{h1}(l) - \bar{s_{h1}}\}^2\right] \tag{13}$$

$$\sigma_{h2}^2 = \frac{1}{l}\left[\sum_{l}\{s_{h2}(l) - \bar{s_{h2}}\}^2\right] \tag{14}$$

発音されていない場合は，音が繋がっていることになるため，音符を connect する．なお，閾値 T_{con} の値として，今回は経験的に式 (15) のような値を設けた場合，最も正しく connect を行うことができた．なお，$stft$ は STFT によって得た短時間ごとのスペクトルの時間的な変化を 2 次元の配列に格納したものであり，$\max(stft)$ は $stft$ の最大値，$\min(stft)$ は $stft$ の最小値を表す．

$$T_{con} = \frac{\max(stft) - \min(stft)}{3} \tag{15}$$

5. 実　　験

5.1 音高，拍数，テンポ，connect の推定正答率

楽曲 A の MIDI データにおいて，Piano, Guitar, Violin, Flute, Hamonica, Trumpet による 6 種類の演奏を用意し，それぞれサンプリング周波数 $f_s = 44100$Hz で録音して入力信号とした．各楽器ごとに得られた出力結果において，音高，拍数，テンポ，connect の正答率を表 1 にまとめた．結果より，Guitar 以外の楽器は全ての項目において正しい検出が行えている．Guitar においては，音高推定は正しく行えていたが，発音開始パルスの検出ミスにより，音符の存在を検出できず，それに伴って音符の長さを誤っていた．発音開始パルスの検出手法を改善する必要があるといえる．

5.2 自動採譜

楽曲における音符の音高・拍数の出力結果から，MusiXTeX[6] 形式で楽譜として出力するスクリプトを記述し，これを用いて楽曲 A を自動採譜した結果を図 12 に示す．以上の手順により，楽曲の音楽信号から音高，拍数，テンポを検出し，connect を行い，自動的に楽譜を作成することができるといえる．

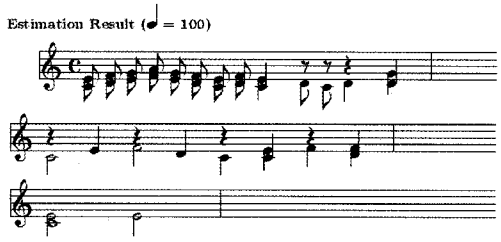

図 12　自動採譜システムで作成した楽譜

6. ま と め

オクターブ 4 の 2 和音以下の楽曲 A について，音高推定アルゴリズムによって音高推定を行い，CF 法を用いて発音開始パルスを検出し，発音開始パルスの自己相関関数からビート長，テンポを計算した．また，検出した発音開始パルス，ビート長を音高推定結果と組み合わせることによって，音価割当，音高割当を行い，connect point において音の繋がりを検出することによって，connect を行う手法を提案した．その結果，楽曲 A を 6 種類の楽器によってそれぞれ演奏した際の音高，拍数，テンポ，connect を約 99% の正答率で検出できた．

今後の課題としては，本システムを複数のオクターブで構成された楽曲や，休符，3 和音以上の和音が含まれた楽曲における採譜へと，拡張することを考える必要がある．

参 考 文 献

1) 三輪多恵子，田所嘉昭，斎藤　努，“零出力に注目したくし形フィルタによる音階検出，”電学論 (C), vol.J118-C,no.1,pp.57- 64,1998.
2) 山口　満，三輪多恵子，田所嘉昭，“並列構成くし形フィルタと特異値分解による多重唱の音高推定，”信学論 (D-II)，vol.J87-D-II,no.4,pp.1020-1029,April. 2004.
3) 森田健夫，山口　満，田所嘉昭，“並列構成くし形フィルタの出力値に注目した採譜のための音高推定法，”信学論 (D-I I)，vol.J87-D-II,no.12,pp.2271-2279,Dec. 2004.
4) 後藤真孝，“コンピュータと音楽の世界-基礎からフロンティアまで-:拍節認識（ビートトラッキング），”共立出版，pp.100-116，1998.
5) Anssi Klapuri Manuel Davy, “Signal Processing Methods for Music Transcription, ” Springer, pp.101-129,2006.
6) MusiXTeX：http://www.ctan.org/tex-archive /macros/musixtex/taupin/