適応型歌声自動伴奏システム

井上 渉[†,☆] 橋本周司[†] 大照 完[†]

本論文では，歌唱に適応した自動伴奏を行うシステムについて述べる．第一の適応として，人が歌唱の途中でテンポを速くしたり遅くした場合に自動伴奏のテンポを実時間で変化させることにより，その歌唱テンポに合わせた自動伴奏を行う．第二の適応は，歌唱の音程が楽譜の音程とずれている場合に実時間で歌声の音程補正を行い，スピーカからは正しい音程の歌声を出力する．まず歌唱テンポに適応した自動伴奏を行うために，実時間で歌声の音声認識（日本語の母音）を行う．この音声認識の結果とあらかじめ知識として持っている歌詞情報とのマッチングをとることにより，歌唱の進行状況を監視し，歌唱の演奏情報（歌唱位置，歌唱テンポなど）を得る．この演奏情報から線形予測により歌唱テンポの予測をして，さらに人間と機械の相互作用モデルを用いて自動伴奏のテンポを決定し，伴奏の出力時間のスケジューリングを行う．これに従って伴奏をMIDI楽器により出力し，歌唱テンポに適応した自動伴奏を実現する．歌唱の音程補正を行うために，歌声の音程をPitch-to-MIDI Converterにより実時間で検出し，この歌声の音程と楽譜（メロディ）の音程とのズレを求める．Digital-Sound-Processorのピッチチェンジ機能を用いて，このズレの分だけ歌声の音程を補正し，スピーカからは正しい音程の歌声を出力する．

Adaptive Automated Accompaniment System for Human Singing

WATARU INOUE,[†,☆] SHUJI HASHIMOTO[†] and SADAMU OHTERU[†]

This paper is concerned with an adaptive automated accompaniment and modulation system for singing based on speech recognition. One purpose of our system is to produce an adaptive accompaniment to follow the singing in real time, so that singers can change the tempo according to their own emotional feelings. Another purpose is to control the singing pitch in real time. When one sings out of tune, the system adjusts the singing pitch to the suitable one. In order to follow the singing, the system analyzes the singing voice, and performs the vowel recognition by the help of DSP. The system monitors the singing vowel by comparing with the lyrics, and gets the information about the singing measure of the score. According to this information, the system predicts the singing tempo. The accompaniments tempo are adjusted to the singing tempo by the linear prediction method using the Man-Machine-Interaction model. In order to compensate the singing voice, the singing pitch is detected by the help of the Pitch-to-MIDI-Converter. The system detects the difference between the singing pitch and the melody score. The singing voice is modulated to the pitch of the melody score by the Digital-Sound-Processor according to the detected difference.

1. まえがき

人間の創造的活動である芸術をコンピュータで扱おうとする研究が試みられており，音楽の諸分野においてもコンピュータは盛んに利用されてきた．音楽は時間軸上の芸術であるので，コンピュータを使用する演奏制御システムの構築の際には実時間処理を考慮しなくてはならない．特に実時間処理が重要となる研究として代表的なものに，自動伴奏システムの開発があり，多くの報告がなされている[1)~5)]．しかし，これらのシステムでは人間の演奏に鍵盤楽器を用いることで，メロディ・ライン抽出の問題を簡単にしており，それよりも自動伴奏自体（演奏位置の取得や伴奏出力のスケジューリング等）について主に述べられている．これに対して，メロディ・ラインの抽出が困難であるフルートや歌唱などのアコースティック・サウンドの入力に対する自動伴奏システムも少ないが報告されている[6)~8)]．筆者らもすでに，歌唱に対する自動伴奏システムと歌声の音程の自動補正システムについて報告し

† 早稲田大学理工学部応用物理学科
Department of Applied Physics, School of Science and Engineering, Waseda University
☆ 現在，NTTヒューマンインタフェース研究所
Presently with NTT Human Interface Laboratories

ている[9]．この自動伴奏システムは，歌声の音程と主旋律の楽譜とのマッチングをとることによって歌唱位置，テンポを検出し，歌に合わせた適応的な伴奏を出力する．また音程補正システムは，歌声の音程がはずれている場合，歌声と楽譜の音程の音階差（ズレ）を検出して，歌声の音程を補正することにより正しい音程の歌声を実時間で出力する．

しかし，この2つの機能は同時に働かせることはできず，正しい音程で歌ったときは自動伴奏が可能で音程補正は行わない，間違った音程で歌ったときは自動伴奏を行うことは困難だが音程補正は行う，となっている．つまり，自動伴奏システムでは自由なテンポで歌うことができるが，歌声の音程の変化パターンを用いて歌唱位置の特定を行っているため，音程が大きくはずれた場合には歌唱テンポに追従した伴奏をすることが難しくなる．また音程補正システムでは音程をはずして歌うことができるが，歌声の音程の変化パターンを用いて歌唱位置を特定することが難しくなるので，歌唱テンポに適応した伴奏出力を考慮に入れていない．

そこで，音程をはずして歌っているときにも適応的な自動伴奏ができるシステムの開発を試みた．ここでは，簡単な音声認識（母音の認識）をシステムに導入し，この認識結果とあらかじめ知識として持っている歌詞情報とのマッチングをとることによって演奏情報（歌唱位置，テンポ）を抽出し，これに適応した伴奏を出力する．また，同時に歌声の音程を監視して，音程がはずれた場合には音程の自動補正を行う．本研究は，音楽を題材として，人間と機械の協調システムの実現を試みたものである．

2. システム概要

本システムの基本構成を図1に示す．主なハードウェア構成は，システム制御用のパーソナルコンピュータ，音声認識用の Digital Signal Processor（以下，DSP と記す）ボード，音程検出用の Pitch-to-MIDI Converter（音程—MIDI 信号変換器），音程補正用の Digital Sound Processor（エフェクタ），および伴奏出力用の MIDI 音源からなっており，曲に関する知識としては主旋律（歌）と伴奏の楽譜，そして歌詞を持っている．

自動伴奏を行うために，マイクからの入力信号を DSP で処理し，音声認識を行う．認識結果とあらかじめ持っている歌詞情報とのマッチングをとって演奏情報（歌唱位置，テンポ）を求め，人間と機械の相互作用を考慮したモデルにより伴奏側のスケジューリングを行う．コンピュータはこのスケジュールに従った

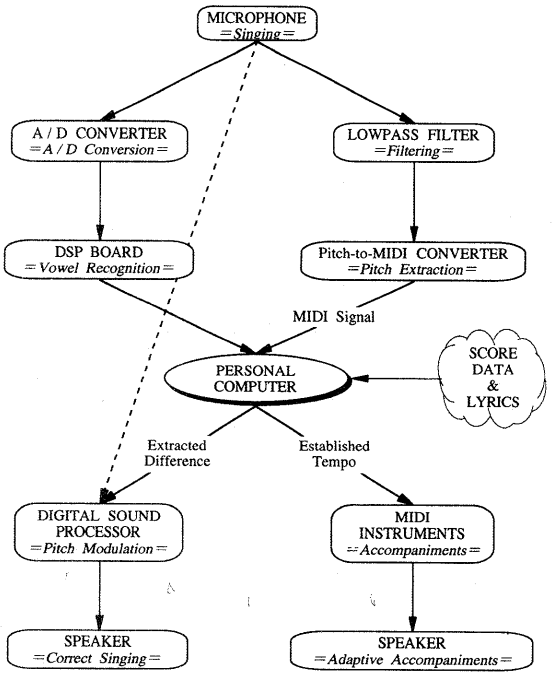


図1 システム構成
Fig. 1 Flow diagram of the system.

演奏，すなわち歌唱テンポに適応した伴奏を出力する．

また，音程の自動補正を行うために，Pitch-to-MIDI Converter を用いて歌声の音程を実時間で MIDI 信号に変換する．検出された音程と主旋律の楽譜から正しい音程とのズレを計算し，この情報を MIDI 信号でエフェクタに送信する．エフェクタは受信した情報により歌声のピッチチェンジを行い，音程を補正した歌声をスピーカから出力する．

3. 歌声の解析

3.1 音声認識

近年，音声認識の技術的進歩には目覚ましいものがあるが，自動伴奏システムでは実時間性が要求されるため，比較的，認識の容易な母音のみの解析を行うことにした．

本研究における音声認識の条件は，連続音声，特定話者，音素単位とした．ただし，音素としては日本語の5種類の母音（/a/,/i/,/u/,/e/,/o/）を対象として認識を行う．

話者条件は特定話者なので，各歌唱者に対して母音の標準パターンの作成を行う．各母音（孤立母音）を歌唱者が普通に出すことができる音程で歌い，各母音の標準パターンを作成する．特徴パターンとしては，スペクトル包絡線を用いている．包絡線は図2に示す

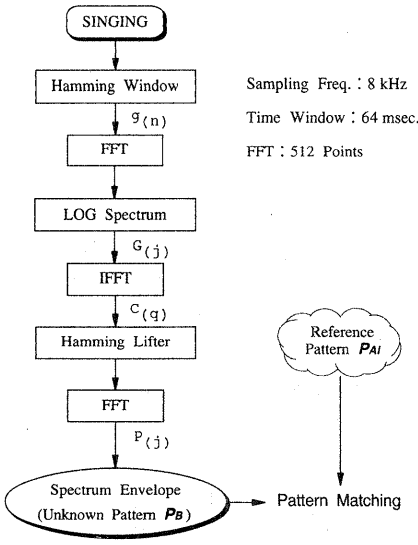


図2 ケプストラム法
Fig. 2 Cepstrum method.

ケプストラム法を利用して求めるが，高速なデジタル信号処理が可能なDSPを用いて実時間処理を行っている[10]．DSPボードには，マイテック社のMSP77230（DSPチップは日本電気の μPD77230）を使用して，標本化周波数8kHz，精度12ビットでA/D変換を行っている．認識率を高くするために周波数分解能をあげると時間分解能が悪くなり，歌唱テンポの検出精度も落ちて伴奏の適応性が悪くなる．そこで認識率および伴奏の適応性を維持できるように，ハミング窓のフレーム時間を64 msec（512 $(=2N)$ 点）とした．ハミング窓で切り出したものを解析する時間を入れて，音声認識の結果は約0.1秒ごとに繰り返し得ることができる．

特徴パターンの正規化の方法を以下に示す．標本化して得られた標本値系列の一部分をハミング窓で切り出したものを $g_{(n)}(1 \leq n \leq 2N)$ とする．認識部では，まず入力信号の音声パワー（短区間平均音声エネルギー）E を式(1)により求める．

$$E = \sum_{n=1}^{2N} \{g_{(n)}\}^2. \qquad (1)$$

この E に対して閾値を設定し，母音，母音以外，音声入力なしの判断をする．そして母音区間と判断された場合に，その区間に対して高速フーリエ変換（FFT）を行い，N 点の対数パワースペクトル $G_{(j)}(1 \leq j \leq N)$ を求め，さらに $G_{(j)}$ を逆フーリエ変換してケプストラム $C_{(q)}$ を求める．そして，$C_{(q)}$ の低ケフレンシー部をリフタ（ハミング窓）によって切り出し，これをフーリエ変換してスペクトル包絡 $P_{(j)}$ を得る．この $P_{(j)}$（すなわち特徴パターン）の正規化を式(2)により行い，$P'(j)$ を求める．ただし，特徴パターンを $\boldsymbol{P} = (P_{(1)}, P_{(2)}, \cdots, P_{(j)}, \cdots, P_{(N)})$ のようにベクトルで表し，正規化したパターンには{'}を付けて表す．

$$P'_{(j)} = \frac{P_{(j)}}{\sqrt{\sum_{k=1}^{N} P_{(k)^2}}} \quad (j = 1, 2, \cdots, N). \qquad (2)$$

以上のような特徴パターンの正規化方法により，各母音に対して，標準となる正規化した特徴パターン $\boldsymbol{P}'_{A_i} = (P'_{A_i}(1), \cdots, P'_{A_i}(N))$ をあらかじめ作成する（添字の $\{A\}$ は標準となるパターンを意味し，{'}は正規化したものを表す．また，$i = 0$: /a/, $\cdots$, $i = 4$: /o/ を意味する．すなわち，$\boldsymbol{P}'_{A_0}$ は/a/の標準パターン，$\boldsymbol{P}'_{A_1}$ は/i/の標準パターン，他も同様）．認識を行う際には，未知入力（歌声）の正規化した特徴パターン $P'_B = (P'_B(1), \cdots, P'_B(N))$ を求めた後，P'_{A_i} と P'_B との間の距離 D_i を式(3)により求める（添字の $\{B\}$ は未知入力のパターンを意味する．また D_0 は P'_{A_0} と P'_B の距離，つまり/a/の正規化した標準パターンと未知入力の正規化した特徴パターンとの距離，他も同様）．

$$D_i = \sqrt{\sum_{j=1}^{N} \left(\boldsymbol{P}'_{A_i}(j) - \boldsymbol{P}'_B(j)\right)^2}. \qquad (3)$$

そして，距離 D_i が最小の母音（$\min\{D_i\}$ となる i）を認識結果 $R\ (=i)$ とする．つまり，最終的な認識結果は，

- 各母音 : $R = 0, 1, \cdots, 4$,
- 母音以外 : $R = 5$,
- 音声入力なし : $R = 6$,

の7種類がある．この認識結果をシステム制御部へ送り，歌唱の進行を監視する．

3.2 ピッチ検出

マイクから入力された歌声の音程はPitch-to-MIDI Converter（Roland：CP-40）により，実時間でMIDI信号（MIDIのノートナンバ，ベロシティ，ノートオン/オフ等）に変換され，システム制御部へ送られる．現在，音程の検出は半音単位で行っているが，今後，より正確な音程補正を行う場合には，人の比弁別閾が440 Hz（感覚レベル60 dB）で0.5%程度（約2.2 Hz）であり，半音の約10分の1の変化を感知することができる[11]ことから，この程度の精度のピッチ検出が必要になると考えている．

音程検出の精度を向上させるため，マイクからの

入力信号を低域通過フィルタ（8次バタワース）に通して高調波を除去し，Pitch-to-MIDI Converter へ入力する．歌声の基本周波数が検出できればよいので，フィルタのカットオフ周波数は男声の場合，一般的なテノール歌手の最高音である約 500 [Hz] とした．

4. システム制御部

4.1 楽譜情報

システムで扱う最も短い音符（以下，最小音長音符と呼ぶ）の持続時間を最小音長時間と定義する．本システムでは最小音長音符を 32 分音符としている．楽譜情報は，

- MIDI ノートナンバー
- 音長（最小音長音符の整数倍）
- MIDI ベロシティ
- 単音，和音を表すフラグ
- MIDI チャンネル

の5バイトを1つのイベント・データとして持っている．また歌詞情報として，歌詞と，その歌詞（1文字）に対応する音符の音長を，楽譜情報と同様に最小音長音符の整数倍で表したものとして持っている．

楽譜情報の作成は，コンピュータ上で数値データをキーボード入力により打ち込む方法や，市販のシーケンサ上で作成した曲データを MIDI インターフェイスを通じてコンピュータ入力し，上記の楽譜情報に変換する方法などを用いて行う．

4.2 歌詞情報とのマッチング

最新の認識結果が得られるたびに，あらかじめ持っている歌詞情報とのマッチングをとることによって歌唱の進行を監視し，演奏情報（歌唱位置，テンポ）を得る．

以下のようにして，誤認識の影響を取り除く．最新の認識結果に過去 k 個の結果を合わせた $(k+1)$ 個における各結果（各母音，母音以外，音声入力なし）の出現頻度 F_i を式 (4) により求める．

$$F_i = \sum_{j=-k}^{0} W_j \cdot E_{(i,R_j)} \quad (i = 0, 1, \cdots, 6) \quad (4)$$

ただし，$E_{(i,R_j)} = \begin{cases} 0 & (i \neq R_j) \\ 1 & (i = R_j) \end{cases}$，$R_j$：過去 k $(= -j)$ 番目の認識結果（$0 \leq R_j \leq 6$，R_0 は最新の認識結果を表す）．

頻度の最も高い結果（$\max\{F_i\}$ となる i）をその時点での結果とする（ただし，より新しい認識結果の影響は大きく，より古い認識結果の影響は小さくなるように重み W_j をつけて出現頻度を求める）．この結果と歌詞情報（次に歌われる母音）が一致したとき，その歌詞が歌われたと判断し，歌唱テンポを求める．

こうして求められたテンポがシステムで定めた最も速いテンポよりも速いときには，誤って歌唱進行を判断したとし，次に新しい認識結果が得られた際に再び同じ歌詞情報とその新しい認識結果とのマッチングをとる．また，システムで定めた時間（後述の実験では，初期テンポの2倍の遅さとした）が経過しても歌詞情報と一致しないときは，システムで定めた最も遅いテンポで伴奏を出力し，曲が途中で止まらないようにしている．

音声認識の結果が約0.1秒ごとに得られることから，テンポの検出も 0.1 秒単位で可能となる．たとえば，四分音符が並んでいる曲を初期テンポ ♩= 60 で演奏する場合，システム側で最も速いテンポを ♩= 120（初期テンポの2倍の速さ），最も遅いテンポを ♩= 30（初期テンポの 2 倍の遅さ）と制限したとすると，♩= $30, 31, \cdots, 54, 60, 66, \cdots, 120$ というテンポを検出することができるので，♩= 60 の演奏から上記の各テンポ（♩= $30, \cdots, 120$）へのテンポ変化に追従することが可能である．

4.3 自動伴奏

前節で求めた演奏情報を用いて次のテンポを予測し，伴奏側のスケジューリングを行う．これに従って伴奏を MIDI 音源により出力し，人間の歌唱と協調した自動伴奏を実現する．

伴奏側のテンポは，人間と機械の相互作用モデルを用いて決定している[12]．以下に，本自動伴奏のモデルを示す．ただし，曲は単位音符の並んだものとしてある．

演奏時の歌唱と伴奏について，それぞれ i 番目の音符の出力時間を $X_{(i)}$，$Y_{(i)}$，単位音符の時間長を $x_{(i)}$，$y_{(i)}$，同期位置 i における歌唱と伴奏の出力時間の差を $\Delta_{(i)}$ とする．また，γ_x，γ_y をお互いに位相をどの程度合わせるかを表すパラメータとすると，

$$X_{(i+1)} = X_{(i)} + x_{(i)} - \gamma_x \cdot \Delta_{(i)}, \quad (5)$$

$$Y_{(i+1)} = Y_{(i)} + y_{(i)} + \gamma_y \cdot \Delta_{(i)}, \quad (6)$$

$$\Delta_{(i)} = X_{(i)} - Y_{(i)}. \quad (7)$$

と表せる．ここで γ_x および γ_y は 0 から 1 の値をとる．たとえば，$\gamma_x = 0$ は演奏者が伴奏とのズレ（位相差）の影響を受けずに演奏を行う場合に相当する．$\gamma_x = 1$ は伴奏とのズレがその後の演奏者の演奏に影響を及ぼし，演奏者が伴奏に合わせようとする場合に相当する．γ_y についても同様で，$\gamma_y = 0$ は伴奏側（システム）が演奏者とのズレの影響を受けずに次の演奏を行う場合に相当する．

次に，$F_{x(i)}$ を人間側の歌唱テンポの目標値，$x'_{(i)}$ を伴奏側が予測した人間側のテンポ，$y'_{(i)}$ を人間側が予測した伴奏側のテンポとして，式(5)および式(6)の $x_{(i)}$，$y_{(i)}$ をそれらのかねあいより，次式のように与える．

$$x_{(i)} = (1-\alpha)\cdot F_{x(i)} + \alpha \cdot y'_{(i)}, \quad (8)$$

$$y_{(i)} = (1-\beta)\cdot x_{0(i)} + \beta \cdot x'_{(i)}. \quad (9)$$

ただし，$x_{0(i)}$ は単位音符の時間長の初期値，α，β は直接相手側のテンポから受ける影響の度合いを示すパラメータである．α および β は0から1の値をとる．たとえば $\alpha = 0$ は人間側のテンポが人間側の歌唱テンポの目標値によってのみ決定され，伴奏側のテンポの影響を受けない場合に相当する．また $\alpha = 1$ は，人間側の歌唱テンポの目標値には関係なく，伴奏側のテンポによってのみ人間側のテンポが決定される場合に相当する．$\beta = 0$ は伴奏側のテンポが人間側のテンポの影響を受けず，初期のテンポで伴奏を行う場合に相当し，$\beta = 1$ は初期テンポに関係なく人間側のテンポに従って伴奏を行う場合に相当する．初期テンポに近いテンポで自動伴奏を行いたいときには β を小さくし，また初期テンポの影響をあまり受けずに自動伴奏を行うときは β を大きくする．

伴奏側のテンポ変化は $y_{(i)}$ を変化させることで実現する．$y_{(i)}$ を決定するために，まず式(5)より $x_{(i)}$ を求め，$x'_{(i+1)}$ を式(10)に示す線形予測式により求める．

$$x'_{(i)} = \sum_{j=1}^{m} c_{(j)} \cdot x_{(i-j)} \quad (c_{(j)}: \text{予測係数}). \quad (10)$$

本システムでは過去2~4個のデータ（式(10)の m に対応）を用いて，線形予測を行っている．

次に $x'_{(i+1)}$ を式(9)に代入し，伴奏側の単位音符の時間長 $y_{(i+1)}$ を決定する．式(6)によって $(i+1)$ 番目の伴奏側の音符が出力される時間 $Y_{(i+1)}$ を計算し，これに従って伴奏出力を行う．

4.4 音程補正

Pitch-to-MIDI Converter で検出した歌声の音程と主旋律の楽譜から正しい音程との音階差を求め，MIDI信号でエフェクタに送信する．音程補正はエフェクタ（ヤマハ：Digital Sound Processor SPX90II）のピッチチェンジプログラムを使用している．

音程補正の量は，エフェクタ側の“BASE KEY”パラメータで指定した音名と，MIDI信号で入力された音名との音階差で決められる．ピッチチェンジは±1オクターブ（±12半音）の範囲で変化させることが可能である．補正の単位は今のところ半音である．さらに細かい補正も可能であるが，歌声の音程の変動も考慮すると，あまり細かい補正は逆に歌唱の個性を失わせる結果になると思われる．

ピッチチェンジにより声質（音色）の変化が生じる．補正後の歌声を周波数解析すると，基本周波数およびその倍音の周波数の近接した周波数にピークが見られ，そのために音色が変化すると考えられる．そこでフィルタを用いて，倍音以外の周波数成分を減少させ，十分ではないが簡単な補正を行っている．

5. 実験結果

5.1 システム動作

本システムを実行すると，まず各母音の標準パターンの作成を行う．作成した標準パターンをファイルとして保存しておけば，次からはこの保存ファイルを読み込むことによって，標準パターン作成の代わりとすることができる．

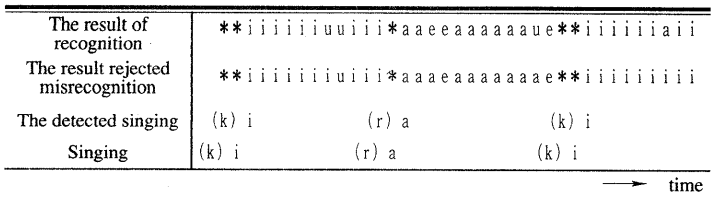
標準パターンの作成（あるいはファイルの読み込み）が終わると伴奏開始の待機状態となり，何か音声が入力されると前奏が開始され，適応自動伴奏および音程の自動補正が行われる．

5.2 結　　果

音声認識の結果の一例を**図3**および**図4**に示す．図において，1段目が4.2節で述べた方法で誤認識を取り除く前の認識結果，2段目が誤認識を取り除いた後の認識結果，3段目が最終的な認識結果と歌詞とのマッチングにより歌われたと判断されたところ，4段目が実際の歌声を表している．4.2節で述べた方法で誤認識を取り除く前の認識率に比べ，誤認識の影響を取り除いた後の認識率の方が10%以上あがっており，誤認識が減少していることが分かる．

図5および**図6**にテンポ追従の一例を示す．図の横軸の時間は曲の進行を表し，縦軸はテンポを表す．ここで，各パラメータは以下のようにして実験を行った．式(5)および式(6)の互いに位相をどの程度合わせるかを表す γ_x，γ_y は，$\gamma_x = 0.5$，伴奏側が遅れているとき $\gamma_y = 0.6$，伴奏側が速いとき $\gamma_y = 0.3$ とし，ある程度，歌唱者と自動伴奏が互いに影響を与えあうものとした．また式(9)において，歌唱テンポから受ける影響の度合いを表す β は $\beta = 1$ とし，式(10)において，$m = 2$（すなわち，過去2個のデータを使用），予測係数 $c_1 = c_2 = 0.5$ として，線形予測を行った．

図5の曲はほぼ単位音符が並んだ曲（「きらきら星」）で，図6は数種類の音符からなる曲（「早稲田の栄光」）である．図において実線（細）が実際の歌唱テンポ，点線がシステムが検出した歌唱テンポ，実線（太）が



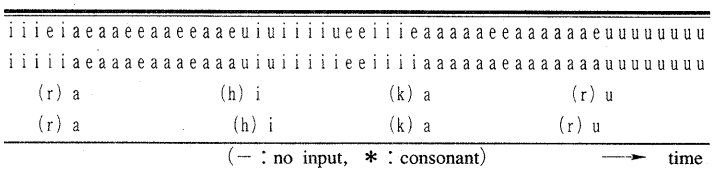
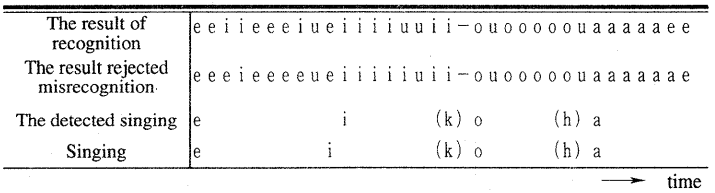


図3 音声認識の結果（1）—「きらきら星」
Fig. 3 The result of speech recognition.



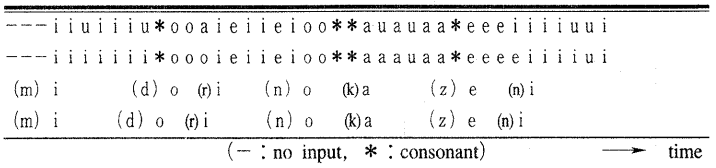


図4 音声認識の結果（2）—「早稲田の栄光」
Fig. 4 The result of speech recognition.

自動伴奏のテンポ，そして図の下方の実線が歌唱と伴奏のズレを表している．ここで，実際の歌唱テンポは歌唱者のタッピングから取ったものである．図を見てわかるように，歌唱と伴奏のテンポに大きなズレはなく歌唱テンポに伴奏が適応している．音声認識の結果を得るのが約 0.1 秒ごとであることから，4.2 節で述べたように検出できる歌唱テンポは制限されているが，テンポ予測により歌唱テンポに適応した伴奏となっている．また音程補正では，前述のように歌声の音色に変化が感じられるが，音程は正しい音程に補正することができた．

6. むすび

音声認識を導入することにより，音程をはずして歌っている場合にも歌詞から歌唱位置を知ることができるようになり，歌唱テンポに適応した自動伴奏が可能となった．また同時に，歌唱の音程補正も行うことによって，より適応的な自動伴奏システムとした．

以前のシステムではピッチ検出の結果を用いたのに対して，本システムでは音声認識の結果を用いて演奏情報の抽出を試みたが，音声認識の時間遅れのため，若干，応答が悪くなった．これは FFT のポイント数を減らすことによってある程度改善できるが，今後は音声認識とピッチ検出の両方の結果を効果的に用いることにより，ロバストで応答特性のよいシステムを実現することを考えている．また，歌詞の間違いやよけいな言葉（場を盛り上げるために使われる言葉等）が歌唱中に出てきても，適応的な伴奏を続けることができるようにしてゆきたい．そのためには，音声の意味解析などを検討する必要がある．

一般に，芸術においては感性が重要であり，コンピュータを用いて音楽情報処理を行う場合にも感性的処理が求められるようになってきている．本論文で検討してきた音楽演奏におけるテンポの変動や揺らぎも，音楽において感性情報を扱ううえで重要であると考えている．将来は，単に歌唱テンポに伴奏を適応させる

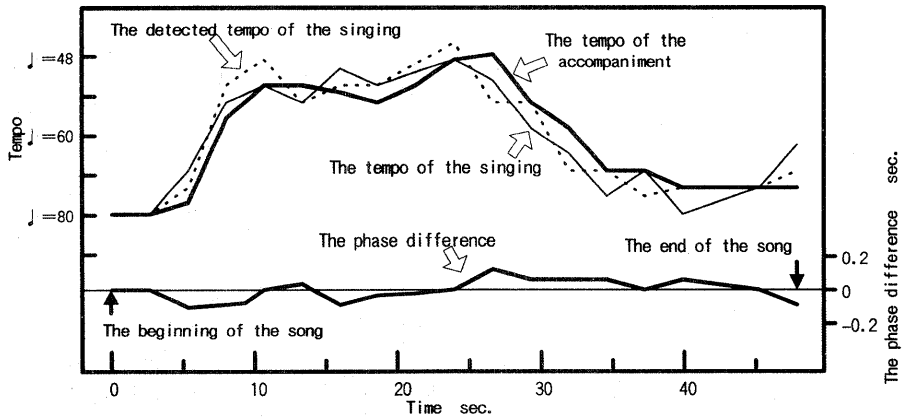


図5 自動伴奏の結果（1）—「きらきら星」
Fig. 5 The result of the automated accompaniment.

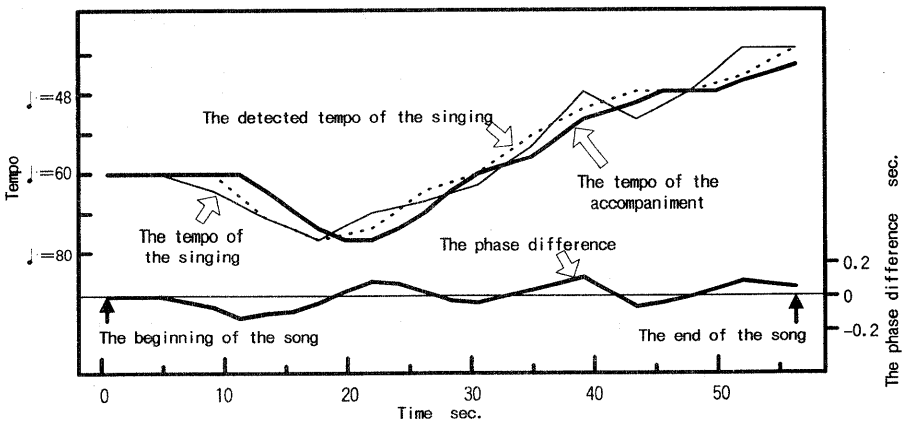


図6 自動伴奏の結果（2）—「早稲田の栄光」
Fig. 6 The result of the automated accompaniment.

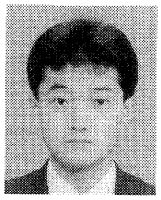
のではなく，歌唱から人間の感性に関わる情報を取り出し，それを考慮に入れた適応伴奏の研究を進めていきたい．

参考文献

1) Dannenberg, R.B.: An On-Line Algorithm for Real-Time Accompaniment, *Proceedings of International Computer Music Conference (ICMC)*, pp.193–198 (1984).
2) Dannenberg, R.B. and Mont-Reynaud, B.: Following an Improvisation in Real-Time, *Proc. of ICMC*, pp.241–248 (1987).
3) 直井，大照，橋本：実時間拍検出機能を用いた自動伴奏システム，日本音響学会講演論文集，pp.465–466 (March, 1989).
4) 和気，加藤，才脇，井口：演奏者の感情を考慮した協調型演奏システム—JASPER—，音楽情報科学研究会・夏のシンポジウム'92 Paper Session Proceedings, pp.43–46 (1992).
5) 堀内，藤井，田中：伴奏者の自主性を考慮した自動伴奏システム，音楽情報科学研究会・夏のシンポジウム '92 Paper Sessions Proceedings, pp.73–78 (1992).
6) Vercoe, B.: The Synthetic Performer in the Context of Live Performance, *Proc. of ICMC*, pp.199–200 (1984).
7) Vercoe, B. and Puckette, M.: Synthetic Rehearsal: Training the Synthetic Performer, *Proc. of ICMC*, pp.275–278 (1985).
8) Katayose, H., Kanamori, T., Kame, K., Nagashima, Y., Sato, K., Inokuchi, S. and Simura, S.: Virtual Performer, *Proc. of ICMC*, pp.138–145 (1993).
9) Inoue, W., Hashimoto, S. and Ohteru, S.: A Computer Music System for Human Singing, *Proc. of ICMC*, pp.150–153 (1993).

10) 新居，大崎：音声処理とDSP，啓学出版，(1989)．
11) 電子通信学会編：新版聴覚と音声，コロナ社，(1980)．
12) 井川，直井，大照，橋本：相互作用モデルによる実時間適応自動伴奏とその動作解析，電通春期全大，pp.7-216–7-216 (1990)．

(平成6年 5 月31日受付)
(平成7年10月 5 日採録)

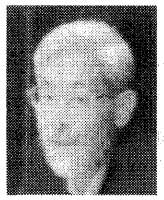


井上　涉（正会員）

1969年生．1992年早稲田大学理工学部応用物理学科卒業．1994年同大学院理工学研究課物理学及応用物理学専攻修士課程修了．音楽情報処理の研究に従事．同年日本電信電話株式会社入社．現在，NTTヒューマンインタフェース研究所音声情報研究部にて，通信サービスの研究に従事．

橋本　周司（正会員）

1948年生．1970年早稲田大学理工学部応用物理学科卒業．東邦大学講師，助教授，早稲田大学助教授を経て，現在，早稲田大学理工学部教授．神経回路網，画像処理，ロボティックス，音楽情報処理の研究に従事．工学博士．



大照　完（正会員）

1921年生．1946年早稲田大学理工学部電気工学科卒業．1964年早稲田大学理工学部教授．1992年早稲田大学名誉教授．この間，電磁気計測，画像処理，学習機械，ロボティックス，音楽情報処理の研究に従事．工学博士．1994年1月に逝去．