情報処理学会論文誌　Vol. 52　No. 12　3853–3867 (Dec. 2011)

VocaListener：ユーザ歌唱の音高および音量を真似る歌声合成システム

中野倫靖[†1]　後藤真孝[†1]

本論文では，ユーザの歌声から，その音高と音量を真似るようにインタラクティブに歌声合成できるシステム VocaListener を提案する．従来，ユーザの歌声から，音高や音量等を推定して歌声合成パラメータとする研究はあったが，歌声合成の条件（歌声合成システムやその音源データ）の違いに対して頑健でなく，歌唱力や歌唱スタイルの修正を行うこともできなかった．そこで VocaListener では，合成された歌唱が入力歌唱と近くなるように，合成パラメータを反復推定することで，上記の条件の変化へ対処する．さらに，入力歌唱に対して，音高のずれやビブラート等の歌唱要素を修正できる支援機能も提供する．2 種類の歌声合成条件を対象とした評価実験では，反復推定に基づく提案システムは従来手法よりも小さな誤差で歌声を合成できた．

VocaListener: A Singing Synthesis System by Mimicking Pitch and Dynamics of User's Singing

Tomoyasu Nakano[†1] and Masataka Goto[†1]

This paper presents a singing synthesis system, *VocaListener*, that interactively synthesizes a singing voice by mimicking pitch and dynamics of a user's singing voice. Although there is a method to estimate singing synthesis parameters of pitch (F_0) and dynamics (power) from a singing voice, it does not adapt to different singing synthesis conditions (e.g., different singing synthesis systems and their singer databases) or singing skill/style modifications. To deal with different conditions, VocaListener repeatedly updates singing synthesis parameters so that the synthesized singing can mimic the user's singing more closely. Moreover, VocaListener has functions to help modify the user's singing by correcting off-pitch phrases or changing vibrato. In an experimental evaluation under two different singing synthesis conditions, mean error values after the iteration were much smaller than the previous approach.

1. はじめに

本研究では，歌声合成システムを利用する多様なユーザが，魅力的な歌声を自由自在に合成して楽曲等を制作し，歌唱という音楽表現の可能性を広げることを支援できる技術の開発を目指す．人間のような歌声を人工的に生成できる歌声合成システムは，多様な歌声での合成が容易に行え，歌唱の表現を再現性高くコントロールできることから，歌唱付き楽曲の制作における可能性を広げる重要なツールである．2007 年以降，市販の歌声合成ソフトウェアを使った楽曲制作を楽しむユーザが急増し，その利用拡大に対する社会的関心の高さから様々なメディアに取り上げられてきた．内閣府による海外向け広報誌においても紹介されている[1] ように，歌声合成ソフトウェアを用いた楽曲が動画共有サービス等に多数投稿され，制作しているユーザが増えただけでなく，そうした楽曲を楽しむリスナも増えた．また一方で，そうして創られた作品は，鑑賞されるだけでなく，そのコンテンツの一部，もしくは全部が新しいコンテンツの中で再利用されるといった，Web を介した音楽の共同制作や新しいコミュニケーションを生み出している現状がある[2),3)]．さらに，高品質な歌声合成技術の実現を目指すことは，人間の歌声知覚・生成機構の解明にもつながる取り組みである．

従来，人間らしい歌声を作るために，歌声合成に関する様々な研究がなされてきた[4)–7)]．近年では，話声の音声合成分野で確立された波形接続方式[8)–10)] や HMM（隠れマルコフモデル）合成方式[11)] といったコーパスベースの合成手法により，実用性の高い歌声合成システムが提案されてきている．コーパスベースの歌声合成システムでは，多様な音源（歌手の歌声）でコーパスを収録すれば，それを切り替えるだけで，多様な声質での合成が容易に行える．これ以降，本論文では音源という単語を「歌声合成用コーパスを構成する同一歌手の歌声の集合」という意味で用いる．現在は，「歌詞」と「楽譜情報（音符系列）」を入力として歌声を合成する合成方式が主流であり，入力された歌詞に基づいた歌声を合成することから，話声合成の text-to-speech（TTS）と対応付けて text-to-singing もしくは lyrics-to-singing と呼ばれることがある[12)]．多くの場合，歌詞に加えて楽譜情報が必要となるが，歌詞のみを入力として自動作曲と歌声合成を同時に行うシステムも存在する[13)]．一方，大量のコーパスを事前に用意することなく，合成対象の歌詞を朗読した話声からその声質を保ったまま歌声に変換する方式も検討されており[12),14)]，speech-to-singing と呼ばれ

†1 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)

る[12]．このような方式では，歌詞と楽譜情報に加えて，歌詞を朗読した話声が必要となる．その他，声質を保ったままの高品質な音声変換や歌声生成に関する詳細な検討[15]もなされてきた．しかし，歌声を入力として与えて，それを真似るように歌声合成できるものはなかった．

本論文では，既存の歌声合成ソフトウェアを用いてユーザの歌声から，その音高と音量を真似るようにインタラクティブに歌声合成できるシステム VocaListener を提案する．本システムにより，ユーザは歌声を入力とした歌声合成方式を新たに選択可能となり，これを本論文では singing-to-singing として新たに提案する．これに類似したアプローチとして，Janer らは入力された歌声を分析して音高・音量・ビブラート情報（深さ・速さ）を推定し，それを正規化してそのまま歌声合成パラメータへ変換していた[16]．しかし，歌声の音高や音量をそのまま合成パラメータとするだけでは，歌声合成の条件（歌声合成システムやその音源）の違いに対してロバストでなく，歌声分析結果の誤りを修正することも考えられていなかった．それに加えて，入力歌唱を真似るだけでは，ユーザの歌唱力を超えることができないという問題もあった．

そこで VocaListener では，合成された歌唱が入力歌唱と近くなるように，合成パラメータを反復更新することで，歌声合成の条件の変化へ対処する．また，歌声分析結果の誤りをユーザがインタラクティブに訂正できる機能，音高のずれやビブラート等の歌唱要素を修正できる支援機能も提供する．本論文では，歌声合成ソフトウェアとして YAMAHA の Vocaloid[10] を対象とし，音高と音量に関する合成パラメータを推定した結果を述べる．

以降，まず 2 章で従来研究の問題点を指摘した後，3 章で VocaListener の実現方法について述べる．続いて，4 章で評価実験，5 章でインタフェースについて述べて，最後に 6 章でまとめと今後の展望を述べる．

2. 関連研究の問題点と解決法

YAMAHA の Vocaloid[10] は，lyrics-to-singing 方式の歌声合成システムであり，歌詞と楽譜情報を入力として与えて合成する．さらに，音高や音量等の歌声合成（表情付け）パラメータを細かく編集・操作することで，より人間らしい歌声や個性的な歌声を合成できる．しかし，このパラメータ調整は，人間らしい自然な歌声を合成しようとすると難易度が高く，適切な知識や時間をかけた調整が必要であるため，誰でも容易に使いこなせるものではなかった．たとえば，細かいニュアンスを表現するために，楽譜と歌詞を入力した後に，ユーザが人手で歌声合成パラメータを長時間調整したりする必要があった．

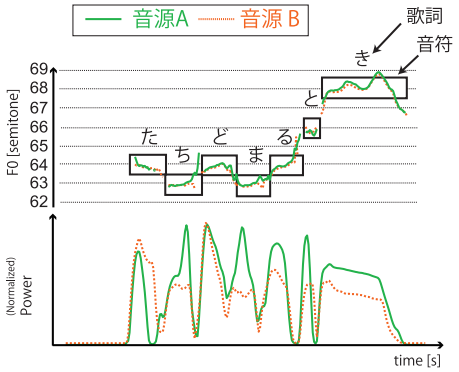

図 1　同じパラメータで合成した場合であっても，歌声合成条件を変えると合成結果が異なる
Fig. 1　Even if the same parameters are specified, the synthesized results always differ when we change the synthesis conditions.

しかも，同じ曲でも歌声合成の条件（歌声合成システムやその音源）が異なると，パラメータを調整しなおす必要があった．これは VOCALOID が素片接続合成方式であるため[10]，収録されている音素ごとに音高と音量にばらつきがあることが，原因の 1 つと考えられる．また，素片接続時に音を滑らかに接続するため[17]，合成時の歌詞や音高，音長に依存して，合成結果をソフトウェア側で調整することも原因と考えられる．実際に 2 種類の異なる音源に対して，同じパラメータで合成すると，合成結果の音高と音量は異なった（**図 1**）．したがって，同じ歌い方で合成したいだけであっても，音源が異なると同一のパラメータを与えても得られる合成結果が異なり，合成システムが異なるとパラメータ自体が異なるため，試行錯誤をともなうパラメータの再調整・再入力が必要であった．そのような原因から，歌声から推定した音高・音量を，そのまま合成パラメータとする従来のアプローチ[16]では，歌声合成の条件の違いに対処できない．**図 2** に，推定した音高と音量をそのまま合成パラメータとした場合の合成結果を示す．

上記 2 つの問題のうち「パラメータ調整の困難さ」に対して VocaListener では，歌声と歌詞を入力とするアプローチで対処する．手作業の代わりに，人間の歌声とその歌詞の分析に基づき，それを真似るように楽譜入力や歌声合成パラメータ調整を行うことで，「歌唱できるが楽譜入力は困難」といったユーザでも歌声合成を楽しみやすい．さらに，歌声入力の結果を手作業で調整したり，手作業による調整で制作した合成歌唱の一部を歌声入力で再調整したりする，といった入力方式の拡張につながって歌声制作の幅を広げることができる．

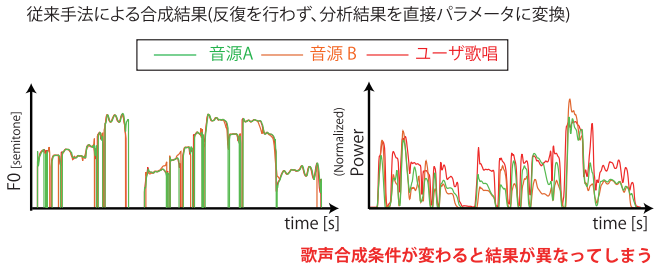

図 2　従来手法[16] の問題点
Fig. 2　Problems of a previous approach [16].

次に「歌声合成の条件の違い」に対して VocaListener では，人があたかも何度も発声練習するかのように，合成音を再度取り込んで分析し，意図したとおりでない部分のパラメータを補正して再度合成する処理を何度も反復することで，歌い方を高精度に真似る歌声合成を実現する．これにより，歌声合成の条件を変えても，その新たな声に合わせて自動的に再度パラメータが調整されるため，上述の問題（図 2）に対処できる．これによって，多様な声質での合成を行いやすく，歌い方やメッセージの表現方法により注力できると考えられる．さらに，ユーザ歌唱の分析結果を編集することで，ユーザ自身が歌唱できない表現（合成したい音高がユーザの声域より高い場合等）に対して歌声合成を行える機能も提案する．

また VocaListener では，歌詞から音符系列を得るために，ユーザ歌唱とその歌詞が入力として与えられると，楽譜情報を用いずに，それらを自動的に対応付ける機能（以降，歌詞アラインメントと呼ぶ）も持つ．Janer らは，音声認識で用いられる Viterbi アラインメントによって，歌詞アラインメントを自動的に行っていた[16]．しかし，ユーザ歌唱を真似て歌声合成するためには，100%に近い精度の歌詞アラインメントが必要だが，Viterbi アラインメントではそのような精度を得ることが難しい．しかも，歌詞アラインメントの結果と，出力される合成音は完全には一致しない[⋆1]が，そのような問題への対処は考えられていなかった．そこで，我々は Viterbi アラインメントによる自動推定を行うだけでなく，有声区間のずれを自動補正し，推定結果が誤っている場合には，その箇所をユーザが指摘するだけで誤りを容易に訂正できる機能も提案する．

⋆1 たとえば Vocaloid[10] では，子音と母音のペアの音節について，母音の開始が発音開始時刻となるよう，時間的に前へずらして合成される．また音節の始端と終端は，前後の音節やそのあるなしによって，出力結果が変化する．

3. VocaListener の実現方法

本論文では，合成歌唱を目標歌唱（入力）へ近づけるコア技術を VocaListener-core，目標歌唱自体を編集する技術を VocaListener-plus と呼ぶ．また，それぞれで必要な要素技術を VocaListener-front-end と呼ぶ．これ以降，ユーザによって与えられた歌唱を目標歌唱，歌声合成システムによって合成された歌唱を合成歌唱と呼ぶ．

図 3 に VocaListener のシステム構成図を示し，これに沿ってシステムの処理概要を説明する．ユーザは，まず**マイク**もしくは**歌声ファイル**によって**歌声入力**を行い（Ⓐ），その歌詞（日本語または英語）を**キーボード**もしくは**テキストファイル**によって**歌詞入力**を行う（Ⓑ）．それらの入力は，**VocaListener-front-end** により分析され，歌詞の**形態素解析**と**Viterbi アラインメント**によって歌声と歌詞の音素を時間的に対応付ける（Ⓒ）．これと同時に，歌声の**音高推定**と**有声・無声判定**を行った後，**ビブラート検出**を行う（Ⓓ）．また**音量推定**も並行して行う（Ⓔ）．これらの処理の中で，音高と音量の推定では **VocaListener-plus** によって声域を変更したり，ビブラートの深さ等を調節したりできる（Ⓕ）．

このようにして得られた歌声分析結果は **VocaListener-core** に渡され（Ⓖ），まず Viterbi アラインメントによって得られた時間情報付きの音素系列（/t a ch i d o m a r u/等）から，歌声合成に必要な音節系列（「たちどまる」等）に変換し，簡易的な音高パラメータ推定を行いながら，入力歌唱と合成歌唱の有声区間が合うように伸縮させる**歌詞アラインメント**（Ⓗ）を行う．続いて，歌声合成ソフトウェアの音高パラメータと音量パラメータを初期値として，**歌声合成システム**とその**音源**（Ⓘ）から，**合成歌唱（一時ファイル）**を得る（Ⓙ）．この一時ファイルの音高と有声区間，そして音量を VocaListener-front-end によって再度分析して（Ⓚ），入力歌唱との違いが小さくなるように音高・音量パラメータを反復推定してゆく（Ⓛ）．ここで，音高と音量に依存関係がある可能性を考慮し，それぞれ別々に推定する．まずは**音高パラメータ推定**（Ⓜ）を行い，続いて**音量パラメータ推定**（Ⓝ）を行った後，反復推定を終了して，歌声合成システムによって**合成歌唱（最終出力）**を得る（Ⓞ）．

また，以上の処理中は，推定結果に誤りがある際に，それをユーザが訂正するため，**モニタ**や**スピーカ**（Ⓟ）によって分析結果や合成結果を随時確認する．

以下，処理の流れに沿って，VocaListener-front-end，VocaListener-plus，VocaListener-core の順にその実現方法を述べる．本文の説明で用いる主要な数式記号を**表 1** に示す．

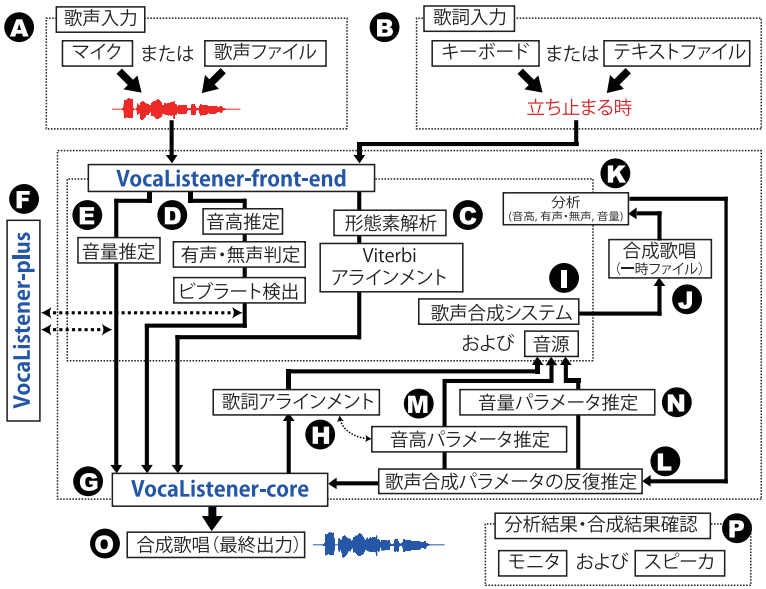

図 3　VocaListener のシステム構成図
Fig. 3　System architecture of VocaListener.

3.1　VocaListener-front-end：歌声分析および歌声合成の要素技術

VocaListener-front-end として，「歌声分析」および「歌声合成」に関する，要素技術を概説する．これ以降，歌声は特に明記しない限り，サンプリング周波数 44.1 kHz のモノラル信号を扱い，処理の時間単位は 10 msec とする．

3.1.1　歌声分析の要素技術

歌声分析においては，歌唱の音響信号から，合成に必要な歌唱の要素を抽出する必要がある．以下，本研究では「音高」，「音量」，「発音開始時刻」，「音長」の抽出のための要素技術を説明する．これらの要素技術は，状況に応じて別の手法で代用してもかまわない．

音高　歌声の基本周波数（F_0 [Hz]）を音高として抽出し，有声/無声の判定も同時に行う．本論文では，Gross Error が低いと報告されている SWIPE [18] を用いた．これ以降 F_0 は，特に明記しない限り，次式で MIDI ノートナンバに対応する単位の実数値 f へ変換して扱う（半音が 1，中央ハ音が 60 に相当）．

表 1　主要な数式記号
Table 1　List of symbols.

数式記号	記号の説明
F_0	音高（基本周波数 [Hz]）
f	音高の MIDI ノートナンバに対応する単位の実数値
f_d	歌唱スタイル変更用の変数（オフセット）
f_t	歌唱スタイル変更用の変数（音高トランスポーズ）
$f'(t)$	ローパスフィルタをかけた音高
$\tilde{f}(t)$	歌唱スタイル変更後の音高
f_n	ノートナンバ
$f(t)$	目標歌唱の音高
$\bar{f}^{(i)}(t)$	i 回目の反復における合成歌唱の音高
$\Delta f_p^{(i)}(t)$	i 回目の反復における PIT
$\Delta f_s^{(i)}(t)$	i 回目の反復における PBS
$\Delta f^{(i)}(t)$	i 回目の反復におけるノートナンバからのピッチベンド
$p(t)$	音量
$p'(t)$	ローパスフィルタをかけた音量
$\tilde{p}(t)$	歌唱スタイル変更後の音量
$p(t)$	目標歌唱の音量
$\bar{p}^{(i)}(t)$	i 回目の反復における合成歌唱の音量
$\bar{p}_m(t)$	DYN を 64 としたときの合成歌唱の音量観測値
$\hat{p}^{(i)}(t)$	i 回目の反復における DYN から音量観測値へ変換した値
ϵ	誤差
$\epsilon_f^{(i)}$	i 回目の反復における音高の相対誤差
$\epsilon_p^{(i)}$	i 回目の反復における音量の相対誤差

$$f = 12 \times \log_2 \frac{F_0}{440} + 69 \tag{1}$$

音量　音量 $p(t)$ は，N を窓幅，$x(t)$ を歌声波形，$h(t)$ を窓関数として，以下のように計算した．

$$p(t) = \sum_{\tau=-N/2}^{N/2-1} \left(\sqrt{(x(t+\tau) \times h(\tau))^2} \right) \tag{2}$$

現在の実装では，N は 2,048 点（約 46 ms），$h(t)$ はハニング窓とした．

発音開始時刻，および音長　Viterbi アラインメントによって自動的に推定して利用する．ここで，漢字かな混じり文の歌詞は，形態素解析器（MeCab [19] 等）によってかな文字列に変換した後，音素列に変換する．変換結果に誤りがあった場合は，ユーザが手作業で訂正する．Viterbi アラインメントでは，音節境界に短い無音（short pause）が入るこ

とを許容した文法を用いた．音響モデルには，連続音声認識コンソーシアムで頒布されている，2002 年度版の不特定話者 monophone HMM [20] を歌声に適応させて使用した．音響モデル適応の際のパラメータ推定手法としては，MLLR（Maximum Likelihood Linear Regression）と MAP 推定（Maximum *A Posteriori* Probability）を組み合わせた MLLR-MAP [21] を用いた．特徴抽出，Viterbi アラインメント，MLLR-MAP による適応には，16 kHz にリサンプリングした歌声を用い，HTK Speech Recognition Toolkit [22] で行った．

3.1.2　歌声合成の要素技術

本研究では，歌声合成システムとしてヤマハ株式会社の開発した Vocaloid2 [10] の応用商品である，クリプトン・フューチャー・メディア株式会社の「初音ミク（以下，CV01）」および「鏡音リン（以下，CV02）」[⋆1]を用いた．

これは，歌声分析と同様，状況に応じて別のシステムで代用してもかまわない．ただし本研究では，歌詞と楽譜情報を入力でき，また表情（音高および音量）に関するパラメータを各時刻ごとに指定できる必要がある．上記システムを採用した理由としては，そうした条件を満たし，市販されていて入手しやすいこと，異なる音源データを利用できること，VSTi プラグイン（Vocaloid Playback VST Instrument）によって後述する反復推定の実装が容易であることがある[⋆2]．

3.2　VocaListener-plus：目標歌唱の編集

VocaListener-plus は，歌唱入力の表現を広げるために目標歌唱自体を編集する機能であり，以下の 2 種類を実現した．これらは状況に応じて利用し，使わなくてもよい．

音高の変更機能

- 調子はずれの補正
  音高がずれた音を修正する．
- 音高トランスポーズ
  自分では歌えない声域の歌唱を合成する．

歌唱スタイルの変更機能

- ビブラート深さの調整
  ビブラート区間を強く・弱くという直感的操作で変更できる．
- 変動成分の調整
  ビブラート区間以外の音高および音量の変動を，強調もしくは抑制できる．

3.2.1　音高の変更機能

「調子はずれの補正」として，歌唱力の評価において相対音高が重要であるため[23]，それを補正する．ここで，相対音高とはある音高から別の音高へ遷移する歌唱の際に，前後の相対的な音高の差を意味する．具体的には，そういった音高の遷移が半音単位となるように音高をずらす．このような補正方法をとることで，ユーザ歌唱の歌唱スタイルを保持したまま調子はずれを補正できると考えられる．本論文では，有声音と判断された区間ごとに，次式で定義する半音間隔に大きな重みを与える関数（半音グリッド）をずらしながら，その区間の F_0 軌跡（$f(t)$）が最も適合するオフセット f_d を決定する．

$$f_d = \underset{g}{\operatorname{argmax}} \sum_t \sum_{i=0}^{127} \exp\left\{-\frac{(f(t)-g-i)^2}{2\sigma_i^2}\right\} \tag{3}$$

現在の実装では $\sigma = 0.17$ とし，$f(t)$ にはカットオフ周波数 5 Hz のローパスフィルタをかけ平滑化[⋆3]を行った．平滑化を行うのは，歌唱における F_0 の 4 種類の動的変動成分（オーバーシュート，ビブラート，プレパレーション，微細変動）[24),25)] を除去するためである．ローパスフィルタのカットオフ周波数は，ビブラート[⋆4]の周波数が，およそ 5 Hz〜8 Hz であること[26),27)] から，それを抑制できるように決定した．オフセット f_d は $0 \le F_d < 1$ の範囲で計算し，音高を次式で変更する．

$$\tilde{f}(t) = \begin{cases} f(t) - f_d & (0 \le f_d < 0.5) \\ f(t) + (1 - f_d) & (0.5 \le f_d < 1) \end{cases} \tag{4}$$

続いて「音高トランスポーズ」は，ユーザ歌唱の音高を全体的，もしくは部分的にずらす機能である．本機能によって，ユーザ自身が表現できない声域の歌唱を合成できる．変更したい区間を選択した後，次式によって f_t だけ変更する．

$$\tilde{f}(t) = f(t) + f_t \tag{5}$$

たとえば，f_t を +12 とすれば，1 オクターブ高い音高の合成歌唱が得られる．

⋆1 http://www.vocaloid.com/product.ja.html
⋆2 観測値や歌声合成パラメータの推定における，処理の時間単位は前述のように 10 msec だが，VSTi によって合成するときのみ，合成パラメータを線形補間によって約 1 msec ごとに与えた．
⋆3 FIR フィルタを使用し，不自然な平滑化を避けるために，無音や閾値（1.8 半音）以上の周波数変化がない区間のみで平滑化した．
⋆4 ビブラートとは，主に音を伸ばすときに周期的に音高を変化させる（揺らす）歌唱テクニックである．

3.2.2　歌唱スタイルの変更機能

目標歌唱の歌唱スタイルを変更するために，前述した歌唱における 4 種類の動的変動成分に着目し，これらを修正する機能として「ビブラート深さ[⋆1]の調整」「変動成分の調整」を提案する．これらの機能は後述するように，音高と音量の関数の 2 つにより定義され（式 (6), (7)），ビブラート深さの調節ではビブラート区間の，変動成分の調整ではビブラート以外の区間の音高と音量の変動を抑制したり強調したりできる．

まず，$f(t)$ にカットオフ周波数 3 Hz のローパスフィルタをかけて，4 種類の F_0 動的変動成分を除去した $f'(t)$ を得る．また，音量に関しても同様に $p(t)$ から，ローパスフィルタによって $p'(t)$ を得る．カットオフ周波数は，ビブラートの周波数が 5 Hz〜8 Hz であること[26),27)]から，ビブラートを含めて，すべての動的変動成分を漏れなく抑制できるように実験的に決定した．

目標歌唱の歌唱スタイルを変更するために，ビブラート深さと変動成分の大きさを，それぞれ調節パラメータ r_v と r_s によって，次式でその度合いを調節する．

$$\tilde{f}(t) = r_{\{v|s\}} \times f(t) + (1 - r_{\{v|s\}}) \times f'(t) \quad (6)$$

$$\tilde{p}(t) = r_{\{v|s\}} \times p(t) + (1 - r_{\{v|s\}}) \times p'(t) \quad (7)$$

r_v はビブラート自動検出法[23)]で検出された区間に適用し，r_s はそれ以外の区間に適用する．ここで，$r_v = r_s = 1$ のときに元の歌唱となる．これらは歌唱全体に対して適用しても，ユーザが指定した区間だけに適用してもよい．$r_v > 1$ とすればビブラートをより強調し，$r_s < 1$ とすれば F_0 の動的変動成分を抑制できる．たとえば，オーバーシュートは，歌唱技量の差によらず生起するが，プロによる歌唱の方が，アマチュアによる歌唱よりも変動が小さいという知見[28)]があり，$r_s < 1$ とすることで変動を小さくできる．

3.2.3　運 用 結 果

図 4 に，音高変更機能として「調子はずれ補正」を，歌唱スタイル変更機能として「ビブラート深さの変更」および「変動成分の調整（抑制）」を適用した結果を示す．実際に音高の補正，ビブラートのみの深さの変更，プレパレーションとオーバーシュートの抑制が可能なことを確認した．

ただし，調子はずれの補正に関しては，有声区間ごとに補正を行ったため，短い音符が適切に補正されない場合があった．その場合であっても，補正後は「音高トランスポーズ」によって，半音単位でずらすだけで適切な音高になることが多かった．単純なずらしのみでなく，調を考慮できたり，音をうまく当てられず徐々に音高が上がったり下がったりする場合を補正できれば，より実用的である．こうした補正に関する機能の拡張は，今後の研究の余地がある．

また「音高トランスポーズ」は，ユーザ歌唱の声域の限界を超えるうえでは必須の機能で，実際の運用では，男性歌唱を入力として女性の歌唱で合成する場合や，その逆に関して特に有効であった．

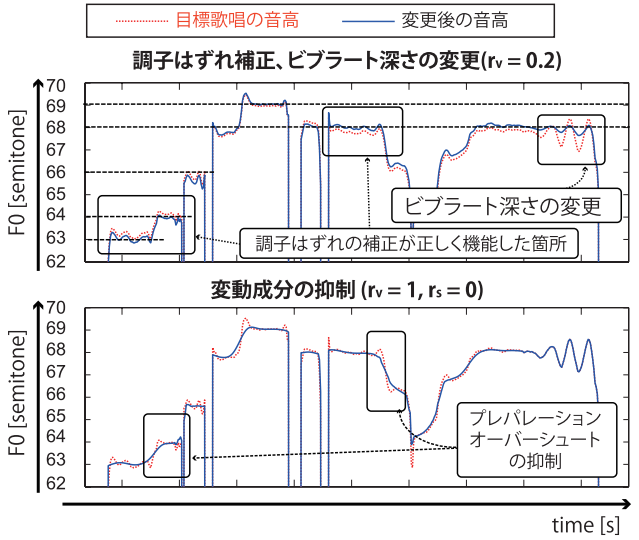

図 4　VocaListener-plus を用いて調整された $F_0(t)$ の例
Fig. 4　Examples of $F_0(t)$ adjusted by VocaListener-plus.

3.3　VocaListener-core：歌声合成パラメータの推定

VocaListener-core では，歌声分析によって得られた目標歌唱と合成歌唱の分析結果に基づいて，歌声合成パラメータを推定する（図 3）．VocaListener-plus で目標歌唱を編集した場合は，その結果を用いる．合成歌唱の分析が必要なのは，合成パラメータが同一であっても，歌声合成の条件の違いによって，合成される歌声が異なるからである．これ以降，合成パラメータとの区別を明確にするため，分析によって得られた値は観測値と呼ぶ．

⋆1 ビブラート深さは，ビブラート区間の平均音高を中心とした変動の幅を意味する．また，実際には音高の変動に同期して音量も同様に変動する．

表 2　推定する歌声合成パラメータと初期値
Table 2　Singing synthesis parameters and those initial values.

歌声合成パラメータ		設定可能な値	初期値
音高	ノートナンバ	0〜127	音節ごとに設定
	PIT	−8,192〜8,191	0（全時刻）
	PBS	0〜24	1（全時刻）
音量	DYN	0〜127	64（全時刻）

3.3.1　初期値の決定

まず，歌詞アラインメント，音高および音量に関する初期値を与える．歌詞アラインメントには，Viterbi アラインメントによって得られた母音の開始時刻と終了時刻を初期値として与えた．音高に関するパラメータは，Vocaloid2 では「音符の音高（ノートナンバ）」「ピッチベンド（PIT）」「ピッチベンドセンシティビティ（PBS）」，音量は「ダイナミクス（DYN）」であり，それぞれ MIDI 規格と同様の意味を持つ（DYN は MIDI 規格の Expression に相当）．

合成パラメータとしての PIT，PBS，DYN 初期値は，全時刻でデフォルトの値とした．各パラメータの設定可能な値と初期値を**表 2** に示す．ここで，PIT は音符の音高に対して，相対的に音高を微調整し，時間軸上で動的に変化させることができるパラメータである．PBS によってその相対変化の幅を設定できて，PBS が 1 なら，ノートナンバから ±1 半音の範囲を 16,384 の分解能で表現できる．ノートナンバは，1 が半音，12 が 1 オクターブに相当し，3.3.3 項の手法を用いて音節ごとに決定する．DYN は 0〜127 の値によって，歌声の音量を制御するパラメータである．基本的には振幅の絶対値に対応していて，その変化の仕方は線形だと期待されるが，正確には歌声合成システムに依存する．

3.3.2　歌詞アラインメントの推定，および誤り訂正

歌詞アラインメントの処理の流れを**図 5** に示す．まず，Viterbi アラインメントの性能や歌声合成システムの特性が原因で，指定した発音開始時刻や音長と時間的にずれて合成されることがある．そこで，有声区間のずれを以下の処理によって補正する．

Step 1)　歌詞の各音節に 1 つの音符を割り当てる．この際，Viterbi アラインメントによって歌声と歌詞の対応付けを行い，それぞれの音節における母音の発音開始時刻と音長を初期値として与える．

Step 2)　2 つの音符がつながっておらず（前の音符の終端と次の音符の始端が離れていて），かつ，目標歌唱ではその区間が有声と判定されていた場合，前の音節の終端を次の音節の始端まで伸ばす．

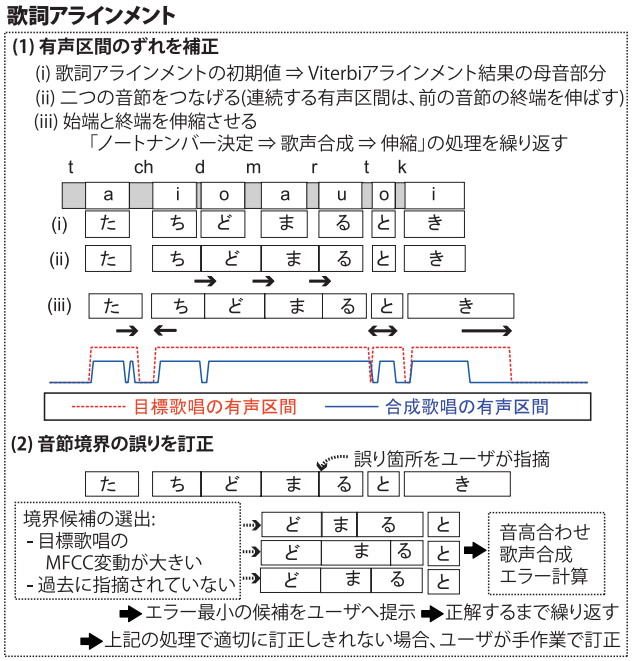

図 5　VocaListener-core における歌詞アラインメントの流れ
Fig. 5　The lyrics alingment procedure of VocaListener-core.

Step 3)　合成歌唱と目標歌唱の有声区間を比較し，それが近くなるように音節の始端と終端を伸縮させる．

Step 4)　ノートナンバを推定して歌声合成し，Step 2) から Step 4) の処理を繰り返す．

続いて，その合成歌唱をユーザが聴いて，ある音節境界が誤っていることに気付いて指摘すると，他の境界の候補が提示される．システムがユーザに提示する新しい境界候補は，目標歌唱の MFCC の時間変化が大きい上位 3 カ所について，それぞれの候補をまず音高を 3.3.3 項および 3.3.4 項で述べる方法で合わせて合成し，目標歌唱との MFCC 距離が最小のものとした．それも誤りだと指摘されたら，次の候補を提示していく．

最後に，上記の処理で適切に訂正しきれない箇所のみ，ユーザが手作業で訂正を行う．

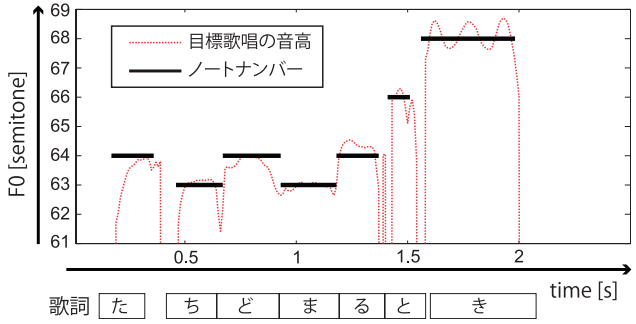

図 6　目標歌唱の音高（F_0）と選択されたノートナンバ
Fig. 6　F_0 of the target singing and estimated note numbers.

3.3.3　音高パラメータの推定 (1)：ノートナンバ

観測された F_0 からノートナンバを決定する．合成歌唱の音高観測値は，PIT と PBS の組合せによっては，ノートナンバ ±2 オクターブまで表現可能であるが，大きな PBS では量子化誤差が大きくなってしまう．そこで，その音符の区間に存在する音高の出現頻度から，PBS の値が小さくなるように，以下の式で F_0 が長い時間とどまっているノートナンバ f_n を選択する（**図 6**）．

$$f_n = \operatorname*{argmax}_n \left(\sum_t \exp\left\{ -\frac{(n - f(t))^2}{2\sigma^2} \right\} \right) \tag{8}$$

ここで，ガウス関数が 1 半音の範囲をカバーするように $\sigma = 0.33$ として計算し，t は音符の始端から終端の時刻で計算する．また，計算には有声区間のみを使用した．

3.3.4　音高パラメータの推定 (2)：ピッチベンド

ノートナンバは固定したまま，合成歌唱の音高 $\bar{f}^{(i)}(t)$ が目標歌唱の音高 $f(t)$ に近づくように，反復によって音高パラメータ（PIT，PBS）を更新して推定する．

時刻 t，i 回目の反復における PIT と PBS を $\Delta f_p^{(i)}(t)$ および $\Delta f_s^{(i)}(t)$ として，以下の処理を繰り返して更新する．

Step 1)　合成歌唱と現在のパラメータを得る．

Step 2)　$\bar{f}^{(i)}(t)$ を推定する．

Step 3)　目標歌唱の音高 $f(t)$ とノートナンバとの音高のずれ $\Delta f^{(i)}(t)$ を以下の式によって更新する．

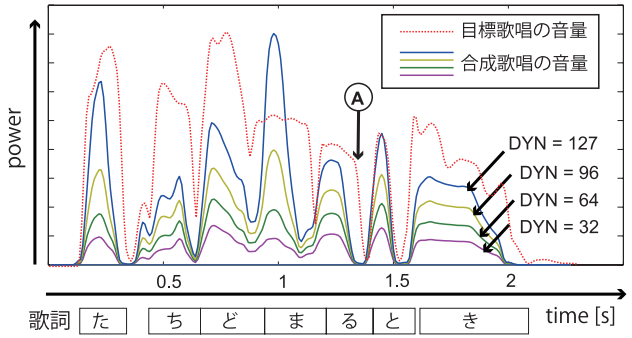

図 7　目標歌唱と 4 種類の DYN パラメータで合成した歌唱の音量観測値
Fig. 7　Power of the target singing and power of the singing synthesized with four different dynamics.

$$\Delta f^{(i+1)}(t) = \Delta f^{(i)}(t) + \left(f(t) - \bar{f}^{(i)}(t) \right) \tag{9}$$

ここで $\Delta f^{(i)}(t)$ は，PIT と PBS から計算される MIDI ノートナンバに対応する単位の実数値（対数周波数）であり，次式で定義される[⋆1]．

$$\Delta f^{(i)}(t) = \frac{\Delta f_p^{(i)}(t)}{8192} \times \Delta f_s^{(i)}(t) \tag{10}$$

Step 4)　$\Delta f^{(i+1)}(t)$ から $\Delta f_s^{(i+1)}(t)$ を最小にするように，$\Delta f_p^{(i+1)}(t)$ と $\Delta f_s^{(i+1)}(t)$ を得る．

3.3.5　音量パラメータの推定 (1)：目標音量の相対値化

目標歌唱の音量観測値は，収録条件の違い等が原因でその絶対的な値が変化するため，相対値化を行う．すなわち，音量の相対的な変化を表現するパラメータを推定するために，目標歌唱の音量を α 倍する．**図 7** に，DYN の値を 0〜127 まで変化させた合成歌唱と，目標歌唱の音量観測値をそれぞれ示す．ここで，合成歌唱の音量と目標歌唱の音量観測値が大きく異なっているのは，歌声合成システムとその音源固有の音量を表示しているからであり，これを近づけるように相対値化を行ってから，DYN の反復推定を行う．

目標歌唱の相対変化を完全に表現するためには，全時刻で目標歌唱の音量を，DYN = 127 で合成した歌唱の音量以下に調整する必要がある．しかし，そのような条件を図 7 のⒶの

⋆1 ノートナンバに $\Delta f^{(i)}(t)$ を加えた値が合成目標の F_0 に相当する．

箇所等でも満たそうとすると，相対値化によって目標音量が小さくなりすぎて，量子化誤差が大きくなってしまう．

そこで，図 7 Ⓐのような一部の再現を断念する代わりに，全体としての再限度が高くなるよう相対値化を行う．目標歌唱の音量観測値を $p(t)$，全時刻の DYN を 64 としたときの合成歌唱の音量観測値を $\bar{p}_m(t)$ として，次式を最小化する相対値化係数 α を決定した．

$$\epsilon^2 = \sum_t \left(\alpha \cdot p(t) - \bar{p}_m(t)\right)^2 \tag{11}$$

具体的には α の推定値は次式で求まる．

$$\alpha = \frac{\sum_t \left(p(t) \times \bar{p}_m(t)\right)}{\sum_t p(t)^2} \tag{12}$$

3.3.6　音量パラメータの推定 (2)：ダイナミクス

こうして得られた相対値化係数 α は固定したまま，音量パラメータ（DYN）を反復推定する．そのために，まずはすべての DYN における合成歌唱の音量観測値を取得する必要がある．そこで，DYN $= (0, 32, 64, 96, 127)$ のそれぞれで実際に各フレーズを合成して，音量観測値を取得しておき，その間は線形補間で求めた．

時刻 t，i 回目の反復において，DYN から上述のように求めた音量観測値へ変換したものを $\hat{p}^{(i)}(t)$ とし，その DYN で合成された歌唱の音量観測値を $\bar{p}^{(i)}(t)$ として，以下の処理を繰り返して更新する．

Step 1)　合成歌唱と現在のパラメータを得る．

Step 2)　$\bar{p}^{(i)}(t)$ を推定する．

Step 3)　$\hat{p}^{(i)}(t)$ を以下の式によって更新する．

$$\hat{p}^{(i+1)}(t) = \hat{p}^{(i)}(t) + \left(\alpha \cdot p(t) - \bar{p}^{(i)}(t)\right) \tag{13}$$

Step 4)　$\hat{p}^{(i+1)}(t)$ から，上述の，DYN とその音量観測値の関係を利用して，音量パラメータ DYN に変換する．

4. 評 価 実 験

本章では，VocaListener-core に関して「歌詞アラインメントの誤り訂正機能の有効性」，「パラメータの反復推定の有効性」および「音源データの違いに対する頑健性」の観点から評価する．

表 3　実験 A，B で用いた目標歌唱（すべて女性歌手）および歌声合成条件
Table 3　Dataset for experiment A and B and synthesis conditions. All of the song samples were sung by female singers.

実験番号	曲番号	使用箇所	曲の長さ [sec]	目標歌唱（歌手名）	合成用音源データ
A	No.07	1 番	103	緒方 智美	CV01
A	No.16	1 番	100	吉井 弘美	CV02
B	No.07	冒頭	6.0	緒方 智美	CV01,02
B	No.16	冒頭	7.0	吉井 弘美	CV01,02
B	No.54	冒頭	8.9	凛	CV01,02
B	No.55	冒頭	6.5	鏑木 朗子	CV01,02

※曲番号は RWC-MDB-P-2001

4.1　VocaListener-core の評価：実験条件

以下の A〜B の 2 種類の実験を行った．RWC 研究用音楽データベース（ポピュラー音楽）RWC-MDB-P-2001 [29] の伴奏なし歌唱データをユーザ歌唱と見なし，実験用の目標歌唱とした．実験で利用した楽曲を**表 3** に示す．すなわち，図 3 における歌声ファイルが，表 3 の「歌手名」による歌声であり，著者がシステムを操作して歌声合成を行った．歌声合成システム（Vocaloid2）では，「ビブラートをつけない」，「ベンドの深さを 0 %」と設定した以外はすべてデフォルト値を用いた．音源データとしては CV01 および CV02 を用いた．

実験 A　長い歌唱（曲中の 1 番，100 秒以上）を利用し，歌詞アラインメントの誤り訂正機能の有効性を評価する．

実験 B　短い歌唱（曲中の 1 フレーズ）を利用し，以下で定義する反復 i 回目の平均誤差（音高誤差 $\epsilon_f^{(i)}$ および音量誤差 $\epsilon_p^{(i)}$）を用いて，パラメータの反復推定の有効性と頑健性を評価する．

$$\epsilon_f^{(i)} = \frac{1}{T_f} \sum_t \left| f(t) - \bar{f}^{(i)}(t) \right| \tag{14}$$

$$\epsilon_p^{(i)} = \frac{1}{T_p} \sum_t \left| 20 \log \left(\alpha \cdot p(t)\right) - 20 \log \left(\bar{p}^{(i)}(t)\right) \right| \tag{15}$$

ここで，音高誤差は目標歌唱と合成歌唱がともに有声となる区間のみ，音量誤差はそれぞれの音量が 0 とならない区間のみを計算する．T_f は有声区間のフレーム数，T_p は音量が 0 でない区間のフレーム数である．

ただし，実験 B では，パラメータ更新の評価が目的であるため，歌詞アラインメント（発音開始時刻と音長）については，人手で正解を与えた．

表 4　音節境界の誤りを指摘した数，および回数（実験 A）
Table 4　Number of boundary errors and number of repairs for correcting (pointing out) errors in experiment A.

曲番号	合成用音源データ	音節総数	誤り指摘 n 回目の誤り数			
			$n=0$	$n=1$	$n=2$	$n=3$
No.07	CV01	166	8	5	2	0
No.16	CV02	128	3	2	0	—

4.2　VocaListener-core の評価：実験結果

以下，前節の 2 つの実験（A，B）の結果を述べる．

4.2.1　実験 A：歌詞アラインメントの誤り訂正

VocaListener-front-end での Viterbi アラインメント結果は，No.07 ではフレーズをまたぐ等の大きな誤りは起きず，No.16 では大きな誤りが 2 カ所起きた．それらをインタラクションによって訂正した後，実験 A を行った結果を**表 4** に示す．No.07 では，計 166 個の音節について，8 カ所の境界誤りがあり，それらは 3 回以内の指摘で訂正できたことを表す．自動推定に誤りが発生する箇所としては，音節境界の直後の音節が/w/や/r/（半母音・流音），/m/や/n/（鼻音）で始まる箇所が多かった．

4.2.2　実験 B：ユーザ歌唱からの合成パラメータ推定

表 5 に，No.07 のフレーズにおける VocaListener を用いた場合の平均誤差を示す．ここで “$i=$” の列は，合成前の反復回数を表し，“$i=0$” は初期値での合成を意味する．この表からは，初期値で合成した場合の大きなエラー（“$i=0$”）が，反復とともに減少して，4 回目の反復で目標に最も近づいた（“$i=4$”）ことが分かる．これ以外のフレーズに関しても，反復によってエラーは同様に減少した．**表 6** に，すべての楽曲における初期値および 4 回の反復での，平均誤差の最大値と最小値を示す．また，表 5 および表 6 における「従来手法」の列は，分析によって得られた音高と音量の観測値を正規化するだけでパラメータへ変換する方法を意味し，Janer らによる従来手法[16] に相当する．反復を 4 回行った後の平均誤差は，従来手法よりも低かった．

また，No.07 の冒頭 2.2 秒に関して，それぞれのパラメータの推定結果と合成結果の音高と音量を**図 8** に示す．

4.3　考　　察

表 4 の結果では，音節境界の誤り自体が少なく（No.07 では 166 音符中 8 カ所，No.16 では 128 音符中 3 カ所），2，3 回の指摘で改善できており，歌詞アラインメント手法，および音節境界の誤り訂正手法が有効に機能していることが分かった．本論文では，話声 HMM を歌声へ適応させて実験を行ったが，歌声のみで学習した音響モデルを用いることで，ここで述べた以上の性能が得られる可能性がある[⋆1]．

表 5　n 回目の反復における平均誤差 [%]（実験 B：No.07）
Table 5　Mean error values after each iteration for song No.07 in experiment B.

推定したパラメータ	合成用音源データ	平均誤差（$\epsilon_f^{(i)}$ [semitone] および $\epsilon_p^{(i)}$ [dB]）					
		従来手法	VocaListener（反復 i 回目）				
			$i=0$	$i=1$	$i=2$	$i=3$	$i=4$
音高	CV01	0.217	0.386	0.091	0.058	0.042	0.034
音高	CV02	0.198	0.352	0.074	0.041	0.029	0.024
音量	CV01	13.65	11.22	4.128	3.617	3.472	3.414
音量	CV02	14.17	15.26	6.944	6.382	6.245	6.171

表 6　実験 B の全フレーズにおける平均誤差の最小値および最大値
Table 6　Minimum and maximum error values for all four songs in experiment B.

推定したパラメータ	平均誤差（最小値 − 最大値）		
	従来手法	VocaListener（反復 i 回目）	
		$i=0$	$i=4$
音高	0.168−0.369	0.352−1.029	0.019−0.107
音量	9.545−15.45	10.46−19.04	1.676−6.560

また表 5 および表 6 の結果では，反復によって誤差が減少して目標歌唱へ近づいており，音源を変えることで初期値が異なっても，最終的に目標歌唱の音高・音量を得るためのパラメータを推定できていた．また，図 8 にも示すように，ユーザ歌唱を真似るためのパラメータがそれぞれ推定され，音源が異なっても，合成された歌唱の音高と音量は目標歌唱によく一致した．このように反復は有効に機能し，従来手法よりも音源の違いに対して頑健であるといえる．ただし，反復回数をさらに増やしても平均誤差は減少しにくくなり，パラメータの量子化誤差や，歌詞アラインメント結果や音高推定結果の軽微な誤りによって，いずれは微少に増減するようになる．実用上は，4 回程度で十分に高い精度で合成パラメータが得られており，定性的な聴取印象ではあるが，合成歌唱には入力歌唱の歌い方が反映されていた．具体的な合成結果は，本システムに関するホームページ[⋆2]を閲覧するか，「VocaListener

⋆1 そのような歌声音響モデルも構築済みであり，予備実験として行った音素書き起こし（音素タイプライタ）では，本音響モデルよりも良い性能を得ていた．
⋆2 http://staff.aist.go.jp/t.nakano/VocaListener/index-j.html

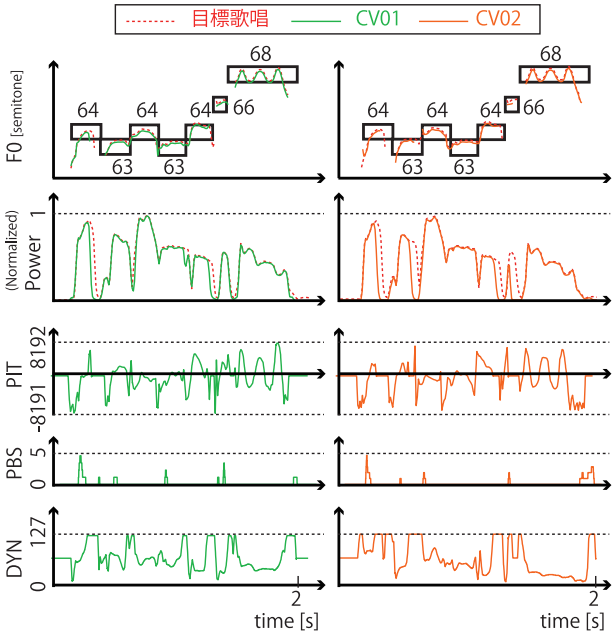

図 8　パラメータの推定結果と合成結果
Fig. 8　The estimated parameters and synthesized results.

（ぼかりす）」を含む Web ページを検索することでアクセスできる．

なお，本システムで合成した歌唱を目標歌唱として与え，パラメータの再推定を試みた結果，元のパラメータとほぼ同じとなることも確認した．本論文ではユーザの歌唱を目標歌唱として入力することを前提に説明したが，このように歌声合成システムの出力を入力してもよい．たとえば，過去に CV01 用に手作業でパラメータ調整した合成歌唱を目標歌唱として，本システムで CV02 用にパラメータ推定するといった使い方もできると考えられる．

5. インタフェース構築

VocaListener による歌声合成の動作検証のために，3 つの基本機能

- ユーザ歌唱の歌い方を真似る歌声合成パラメータ推定機能
- 違和感を感じた箇所を指摘するだけで訂正できる『ダメ出し』インタラクション機能
- 合成結果を自分好みの表現へ変更できる歌唱力補正機能

を実現するユーザインタフェースを実装した．処理は C++言語によって記述し，GUI の実装には Visual Studio 2005 を用いた．製品レベルのクオリティや使いやすさは備えておらず，エンドユーザの利用を想定していない動作確認用の GUI である．

インタフェースの画面例を**図 9** に示す．図の上部は，次の 3 つのウィンドウで構成される．

- 楽譜ウィンドウ（図 9：Ⓐ）
  歌詞とともに，自動推定された音符列（緑の長方形）と F_0 軌跡（赤：目標歌唱，青：合成歌唱）が表示される．縦軸は音の高さで，図中右端のスライダでその表示範囲を変更できる．
- アラインメントウィンドウ（図 9：Ⓑ）
  自動推定された歌詞アラインメント（目標歌唱と歌詞の時間的対応付け）の結果が表示される．誤り訂正もこのウィンドウ上で行う．
- パワーウィンドウ（図 9：Ⓒ）
  目標音量（赤）および合成音量（青）が表示される（図中の表示は反復前の音量）．

歌声合成は，図の下部の各種コントロール（ボタン等）により，インタラクティブに行う．まずユーザは事前に，録音した歌声（wav 形式）を用意する．それを目標歌唱としてインタフェースに入力し，次節で述べる手順で歌声合成を行う．その際，目標歌唱を聴いたり，その音高・音量や歌詞のアラインメントといった分析結果を見たり，VocaListener による合成結果を聞いたりしながら，インタラクティブに誤りを訂正して歌声合成する（図 3）．

5.1　本インタフェースによる歌声合成

本インタフェースを用いた歌声合成について，処理を手順ごとに説明する．

5.1.1　ステップ 1：歌声合成条件の選択

まず，合成に使用する歌声合成システムとその音源を選択する[⋆1]（図 9：Ⓓ）．ヤマハ株式会社の歌声合成技術 Vocaloid/Vocaloid 2 に基づく，2010 年 12 月時点での全ソフトウェア（日本語と英語歌詞のみが流通）に対応させた．

5.1.2　ステップ 2：目標歌唱と歌詞の分析・誤り訂正

次に，目標歌唱とその歌詞テキスト（漢字仮名交じり文）を指定する．歌声合成のための音高（F_0）・音量推定，歌詞のテキスト処理がなされ，音高と音量がⒶⒸに，歌詞を発音系列（平仮名）へ変換した結果がⒺ下部に，それぞれ表示される．F_0 推定結果が誤っていた

⋆1 ただし，合成の過程で好きなときに変更することが可能である．

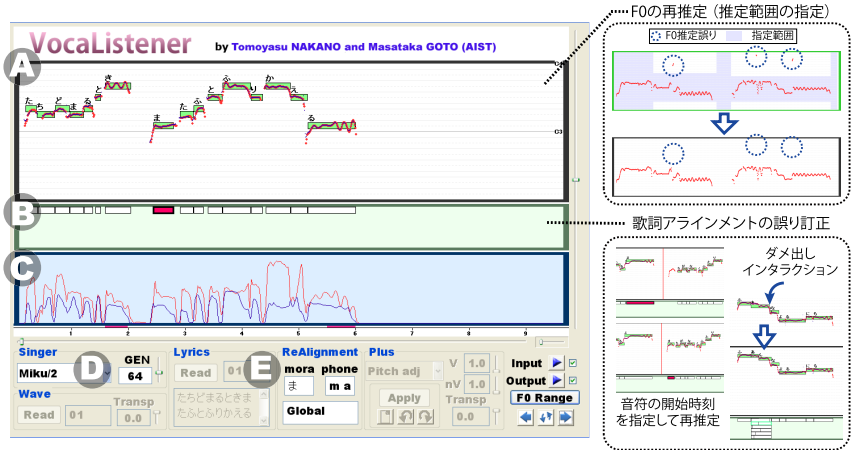

図 9　VocaListener の表示画面例
Fig. 9　An example VocaListener screen.

場合は，Ⓐ上のドラッグで推定する時間・音高（周波数）範囲を指定すると自動的に正しく再推定される．また，発音系列の誤りは仮名のテキスト編集で訂正できる．

5.1.3　ステップ 3：歌詞アラインメント結果の誤り訂正（『ダメ出し』インタラクション）

歌詞アラインメント結果に含まれる誤りを訂正する．まず，まれに比較的長い無音をまたいだ大きな対応付け誤りが発生するので，そのおおよそ正しい開始時刻を指定すると，自動的に正しく再推定される．次に，より細かい音節境界誤りを訂正する．誤っている箇所を指摘するだけで，可能性の高い新たな境界候補が自動生成されてⒷ上に表示されるので，それを選択するだけで簡単に訂正できる．最後に，上記で半自動的に訂正できなかった誤りを，音符位置や長さを微調整して訂正する．

5.1.4　ステップ 4：歌唱力補正

補正したい時間範囲をドラッグで指定し，3 種類の補正ができる．まず「調子はずれ補正」では，音程（音高の遷移）がずれている箇所を自動補正して合成できる．次に「歌唱スタイルの変更」では，ビブラート区間とそれ以外の区間の音高と音量を，別々に強調・抑制して補正でき，自分好みの表現で合成できる．最後に「音高トランスポーズ」では，合成の声域を変え，ユーザが歌えない声域での歌唱も合成できる．

5.2　提案インタフェースの有用性に関する考察

本章で提案したインタフェースは，歌詞と楽譜情報を入力する従来の歌声合成インタフェース（Vocaloid2 における Score Editor [10] 等）と比較して次の 2 点が新しく，有用であった．

i) 歌声による楽譜入力および歌声合成パラメータ調整
　従来は，マウスやキーボードでしか入力できなかったが，歌声での入力が可能となり，歌声合成のパラメータ調整をする際の選択肢が増えた．歌声による入力では，歌詞アラインメントや F_0 推定の誤りが存在するため，従来は不要であった「誤り訂正」が必要となるが，ダメ出しインタラクションによる容易な誤り訂正を実現した．

ii) 歌声合成の条件の違いに頑健な歌声合成パラメータ調整
　音源等を変更した際に，ユーザがパラメータ調整に費やす時間が削減され，「どのような歌声を作りたいか」により注力できた．真似るための入力としての音量や音高は，必ずしも歌声から推定したものをそのまま使う必要はなく，編集による歌唱力補正を可能にした．今後は，そうした補正に加え，他の入力デバイスによる音高や音量の直接修正や，音高や音量だけを他の楽器にものに置換する拡張等も考えられる．

6.　おわりに

本論文では，人間の歌唱を入力としてそれを近似する歌声合成パラメータを推定するシステム VocaListener およびそのインタフェースについて述べた．また本技術に対する様々な観点からの考察を議論した．我々は，VocaListener の「模倣」を出発とするアプローチは自然性達成の第 1 歩として適切だと考えており，まずは模倣でもよいから「自然さ」を的確に達成したうえで，次のモデル化等の段階につなげていきたいと考えている．今回は，音高と音量に着目し，それを高精度に模倣できる新しい技術の提案を行った．ただし，正しい音高と音量が再現されても，歌声として自然である保証はできない．なぜなら，合成される歌声の自然さは，歌声合成ソフトウェアの性能にも依存するからである．音源として収録されている音素の品質によっては，局所的に不自然な合成結果になる現象を確認しており，より自然な歌声合成へ向けて，今後も研究を進めていきたい．

このような，ユーザ歌唱を真似る歌声合成は，本人の歌唱をそのまま使うことにない利点が得られる．つまり，本人以外の多様な声質で手軽に歌唱曲を制作できることが重要で，たとえば女性が男性ボーカルの楽曲を制作したければ，いくら本人が歌えても不十分である．さらに VocaListener では，手作業で労力をかけて調整し直さずに，多様な歌声合成ソフトウェアの声質を手軽に切り替えて曲のイメージに合うか試すことができる利点がある．

このように，歌声合成手段の多様化だけでなく，声質の多様化を可能にする点も，本研究の1つの意義であり，それが，ピッチ変換や市販のボイスチェンジャ等の方法と異なる点である．声質変換に関しては，さかんに研究開発がなされているが[30),31)]，入力の録音状況の影響を受けやすく，歌声合成ソフトウェアに匹敵する合成品質はまだ手軽には得られにくい．VocaListener では，入力に多少のノイズが含まれていても音高と音量さえ推定できればよく，合成音はつねに歌声合成ソフトウェアによるものであるため，クリーンで多様な声が手軽に得られる利点がある．

今後は，以下のように研究を発展させていきたい．

より人間らしい合成歌唱の実現

ブレス音（吸気音，息継ぎ音）は，合成歌唱をより人間らしく聴こえさせるために重要であると考えられるため，ブレスの自動検出手法[32)]を利用して付与する．また，本論文では，声質に関するパラメータはあえて推定していなかったが，声質（スペクトル包絡）の動的な変動を組み込むことができれば，より人間らしい歌唱の実現につながっていく．

VocaListener-plus の機能の充実

VocaListener-plus の機能を拡張し，調子はずれの補正法の改良や，他人の歌唱スタイルを利用する機能を追加することで，ユーザがより歌声合成システムを使いやすくできる．後者は，たとえば，様々なユーザのビブラートを収録したテンプレートを用意し，好みのビブラートを状況に応じて付与する機能が考えられる．

歌詞アラインメントの高精度化と評価

本論文では，歌詞アラインメントにおいて話声 HMM を歌声に適応させて利用したが，別途，歌声用音響モデルの構築を試みている．歌詞アラインメントは高品質な singing-to-singing 合成システムの構築には必須であるため，今後構築と評価を行っていきたい．

『メタ歌声合成システム』の実現

従来，様々な歌声合成システムやその音源データが存在し，ユーザは手作業でその合成パラメータを決定していた．しかし，歌声合成の条件が変わると，パラメータの再調整が必要となるという問題があった．本システムはそのような問題を解決し，ユーザはパラメータを1度だけ調整すれば，歌声合成システムや音源データに依存せず，同一の表現を様々な条件で合成できる．そこでこれを，「メタ歌声合成システム」と名付ける．今後，そうした観点からも，機能の追加や拡張を行う予定である．

人間を知る追求

本研究の根底には，「人間らしい歌唱」とは何かを解明し，より人間を知ることがあり，本システムは，そうした歌声研究の基本ツールとしても貢献できる．たとえば，音高や音量を独立に真似た合成歌唱を用いて心理実験を行うことで，歌唱の個人性知覚に関する新しい知見が得られる可能性がある．

謝辞　本研究の一部は，科学技術振興機構 CrestMuse プロジェクトによる支援を受けました．本研究では，ヤマハ株式会社および，クリプトン・フューチャー・メディア株式会社の歌声合成ソフトウェア「初音ミク（CV01）」「鏡音リン（CV02）」を使用しました．ニコニコ動画等の様々な場で，本研究の合成結果や研究活動全般についてコメントし，議論いただいた方々に感謝いたします．また，本研究に対し有益な議論をしていただき，実装に関するご助言をいただいた藤原弘将氏，音響モデルの適応等でご助言をいただいた緒方淳氏，歌声合成に関して有益なご意見をいただいた齋藤毅氏に感謝いたします．本研究では，RWC 研究用音楽データベース（ポピュラー音楽 RWC-MDB-P-2001）を使用しました．

参　考　文　献

1) Cabinet Office, Government of Japan: Virtual Idol, *Highlighting JAPAN through images*, Vol.2, No.11, pp.24–25 (2009), available from ⟨http://www.gov-online.go.jp/pdf/hlj_img/vol_0020et/24-25.pdf⟩.
2) 濱崎雅弘，武田英明，西村拓一：動画共有サイトにおける大規模な協調的創造活動の創発のネットワーク分析—ニコニコ動画における初音ミク動画コミュニティを対象として，人工知能学会論文誌，Vol.25, No.1, pp.157–167 (2010).
3) 濱野智史：インターネット関連産業，デジタルコンテンツ白書 2009，pp.118–124 (2009).
4) Depalle, P., Garcia, G. and Rodet, X.: A virtual castrato, *Proc. International Computer Music Conference* (*ICMC'94*), pp.357–360 (1994).
5) Cook, P.R.: Identification of Control Parameters in An Articulatory Vocal Tract Model, with Applications to the Synthesis of Singing, Ph.D. Thesis, Stanford Univ. (1991).
6) Cook, P.R.: Singing Voice Synthesis: History, Current Work, and Future Directions, *Computer Music Journal*, Vol.20, No.3, pp.38–46 (1996).
7) Sundberg, J.: The KTH Synthesis of Singing, *Advances in Cognitive Psychology, Special issue on Music Performance*, Vol.2, pp.131–143 (2006).
8) 吉田由紀，中嶌信弥：歌声合成システム：CyberSingers，情報処理学会研究報告音声言語情報処理 99-SLP-25-8，Vol.99, No.14, pp.35–40 (1998).
9) Bonada, J. and Xavier, S.: Synthesis of the Singing Voice by Performance Sampling and Spectral Models, *IEEE Signal Processing Magazine*, Vol.24, No.2, pp.67–79 (2007).

10) Kenmochi, H. and Ohshita, H.: VOCALOID – Commercial Singing Synthesizer based on Sample Concatenation, *Proc. 8th Annual Conference of the International Speech Communication Association* (*INTERSPEECH 2007*), pp.4011–4010 (2007).
11) 酒向慎司，宮島千代美，徳田恵一，北村　正：隠れマルコフモデルに基づいた歌声合成システム，情報処理学会論文誌，Vol.45, No.7, pp.719–727 (2004).
12) Saitou, T., Goto, M., Unoki, M. and Akagi, M.: Speech-To-Singing Synthesis: Converting Speaking Voices to Singing Voices by Controlling Acoustic Features Unique to Singing Voices, *Proc. 2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (*WASPAA2007*), pp.215–218 (2007).
13) Fukayama, S., Nakatsuma, K., Sako, S., Nishimoto, T. and Sagayama, S.: Automatic Song Composition from the Lyrics Exploiting Prosody of the Japanese Language, *Proc. 7th Sound and Music Computing Conference* (*SMC2010*), pp.299–302 (2010).
14) 森山　剛，小沢慎治：好みの歌唱様式による歌詞朗読音声からの歌唱合成，情報処理学会研究報告音楽情報科学 2008-MUS-74-6，Vol.2008, No.12, pp.33–38 (2008).
15) 河原英紀，片寄晴弘：高品質音声分析変換合成システム STRAIGHT を用いたスキャット生成研究の提案，情報処理学会論文誌，Vol.43, No.2, pp.208–219 (2002).
16) Janer, J., Bonada, J. and Blaauw, M.: Performance-driven Control for Sample-Based Singing Voice Synthesis, *Proc. 9th Int. Conference on Digital Audio Effects* (*DAFx-06*), pp.41–44 (2006).
17) 剣持秀紀，大下隼人：歌声合成システム VOCALOID—現状と課題，情報処理学会研究報告音楽情報科学 2008-MUS-74-9，Vol.2008, No.12, pp.51–58 (2008).
18) Camacho, A.: SWIPE: A Sawtooth Waveform Inspired Pitch Estimator for Speech And Music, Ph.D. Thesis, University of Florida (2007).
19) 工藤　拓：MeCab: Yet Another Part-of-Speech and Morphological Analyzer, 入手先⟨http://mecab.sourceforge.net/⟩.
20) 河原達也，住吉貴志，李　晃伸，坂野秀樹，武田一哉，三村正人，伊藤克亘，伊藤彰則，鹿野清宏：連続音声認識コンソーシアム 2002 年度版ソフトウェアの概要，情報処理学会研究報告音声言語情報処理 2001-SLP-48-1，Vol.2003, No.48, pp.1–6 (2003).
21) Digalakis, V. and Neumeyer, L.: Speaker Adaptation Using Combined Transformation and Bayesian Methods, *IEEE Trans. Speech and Audio Processing*, Vol.4, No.4, pp.294–300 (1996).
22) Young, S., Evermann, G., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V. and Woodland, P.: *The HTK Book* (2002).
23) 中野倫靖，後藤真孝，平賀　譲：楽譜情報を用いない歌唱力自動評価手法，情報処理学会論文誌，Vol.48, No.1, pp.227–236 (2007).
24) Saitou, T., Unoki, M. and Akagi, M.: Development of an F0 Control Model Based on F0 Dynamic Characteristics for Singing-Voice Synthesis, *Speech Communication*, Vol.46, pp.405–417 (2005).
25) Mori, H., Odagiri, W. and Kasuya, H.: F_0 Dynamics in Singing: Evidence from the Data of a Baritone Singer, *IEICE Trans. Inf. & Syst.*, Vol.E87-D, No.5, pp.1068–1092 (2004).
26) Seashore, C.E.: A Musical Ornament, the Vibrato, *Psychology of Music*, pp.33–52, McGraw-Hill (1938).
27) 森勢将雅，平地由美，坂野秀樹，入野俊夫，河原英紀：STRAIGHT を用いたビブラート歌唱音声の統計的性質，日本音響学会 2005 年周期講演論文集 3-P-15，pp.269–270 (2005).
28) 齋藤　毅，鵜木祐史，赤木正人，榊原健一：歌声の基本周波数変化に含まれるオーバーシュートの知覚への影響に関する検討，日本音響学会聴覚研資，H–2006–109, pp.611–616 (2006).
29) 後藤真孝，橋口博樹，西村拓一，岡　隆一：RWC 研究用音楽データベース：研究目的で利用可能な著作権処理済み楽曲・楽器音データベース，情報処理学会論文誌，Vol.45, No.3, pp.728–738 (2004).
30) Toda, T., Black, A. and Tokuda, K.: Voice Conversion Based on Maximum Likelihood Estimation of Spectral Parameter Trajectory, *IEEE Trans. Audio, Speech and Language Processing*, Vol.15, No.8, pp.2222–2235 (2007).
31) 大谷大和，戸田智基，猿渡　洋，鹿野清宏：STRAIGHT 混合励振源を用いた混合正規分布モデルに基づく最ゆう声質変換法，電子情報通信学会論文誌，Vol.J91-D, No.4, pp.1082–1091 (2008).
32) Nakano, T., Ogata, J., Goto, M. and Hiraga, Y.: Analysis and Automatic Detection of Breath Sounds in Unaccompanied Singing Voice, *Proc. 10th International Conference of Music Perception and Cognition* (*ICMPC 10*), pp.387–390 (2008).

(平成 23 年 1 月 6 日受付)
(平成 23 年 9 月 12 日採録)

中野　倫靖（正会員）

2003 年図書館情報大学卒業．2008 年筑波大学大学院図書館情報メディア研究科博士後期課程修了．博士（情報学）．現在，産業技術総合研究所研究員．日本音響学会会員．2006 年日本音楽知覚認知学会研究選奨，2007 年インタラクション 2007 インタラクティブ発表賞，2009 年情報処理学会山下記念研究賞（音楽情報科学研究会），2010 年音楽情報科学研究会（夏のシンポジウム 2010）ベストプレゼンテーション賞各受賞．

後藤　真孝（正会員）

1998 年早稲田大学大学院理工学研究科博士後期課程修了．博士（工学）．同年電子技術総合研究所に入所し，2001 年に改組された産業技術総合研究所において，現在，情報技術研究部門メディアインタラクション研究グループ長．統計数理研究所客員教授，筑波大学大学院准教授（連携大学院），IPA 未踏 IT 人材発掘・育成事業未踏ユースプロジェクトマネージャーを兼任．ドコモ・モバイル・サイエンス賞基礎科学部門優秀賞，科学技術分野の文部科学大臣表彰若手科学者賞，情報処理学会長尾真記念特別賞等，25 件受賞．