Vol.2017-MUS-116 No.3
2017/8/24

楽曲中の歌声とユーザ歌唱の リアルタイムアラインメントに基づく 伴奏追従型カラオケシステム

和田 雄介[1,a)]　坂東 宜昭[1,b)]　中村 栄太[1,c)]　糸山 克寿[1,d)]　吉井 和佳[2,e)]

概要：本稿では，入力された音楽音響信号から伴奏音を抽出し，ユーザ歌唱のテンポ変化に自動で追従して再生するカラオケシステムを提案する．このシステムによって，ユーザは任意の楽曲を，テンポを自由にアレンジしながら歌うことが可能になる．このシステムの主な利点は，ユーザが楽譜 (MIDI ファイル) を用意する必要がないことと，システムを起動した後すぐにカラオケを楽しめることである．これらを実現するために，このシステムでは音源分離手法および audio-to-audio アラインメント手法をオンラインで並列に実行する．まず，入力された音楽音響信号が，ロバスト非負値行列因子分解 (RNMF) のオンライン版を用いて歌声と伴奏音に分解される．その後，分離された歌声信号とユーザ歌唱が，動的時間伸縮 (DTW) によって時間方向に同期される．最後に，DTW によって推定されたワーピングパスを用いて伴奏音が伸縮され，再生される．被験者実験により，このシステムの有効性が確認され，このシステムは新しい歌唱の楽しみ方を提示しうることが示された．

1. はじめに

カラオケは，歌唱の楽しみ方の一つであり, ユーザはあらかじめ用意された伴奏音に合わせて好きな曲を歌える．現在のカラオケ産業では，伴奏音の生成に楽譜 (MIDI ファイル) が用いられている．この MIDI ファイルの作成には，専門家による楽曲の楽譜化が必要であり，新しい CD をカラオケに収録するごとに，その音源を楽譜化するという作業を行わなければならない．この方法の問題点は 2 つあり，1 つは楽譜化に多大な時間と専門的な技術が必要となること，もう 1 つは MIDI ファイルを用いて合成される伴奏音の音質が元の音源に劣ることである．

近年，CGM (Consumer Generated Music) という音楽の楽しみ方が広まっており，多数のアマチュアが自作の楽曲を Web 上に公開している．音楽視聴支援サービス Songrium [1] によると，2007 年 7 月時点で，120 万曲を超える楽曲が Web 上に公開されている．そのような楽曲群を全て楽譜化するのは現実的ではなく，任意の音楽音響信号から，楽譜や歌詞の情報を用いることなく高品質な伴奏音を生成することが重要となる．

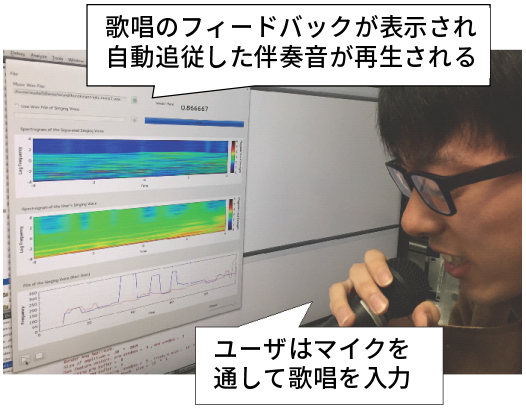

図 1　提案システムの使用例．ユーザは自分の歌唱のテンポ変化に追従する伴奏音を聞きながら，自由に歌唱のテンポをアレンジして歌える．画面には，ユーザ歌唱と元の音源中の歌唱それぞれのスペクトログラムと F0 軌跡がリアルタイムに表示される．その他に，音源分離の進行状況も表示される．

その他に考えられる現在のカラオケシステムの問題点は，ユーザが伴奏音のテンポを自分で設定しなければならないということである．これは，テンポが一定のポピュラー・ソングなどでは問題にならないものの，オペラやゴスペル，フォークソングといったジャンルの楽曲では，表現の一環

[1] 京都大学 大学院情報学研究科
[2] 京都大学/理研 AIP
[a)] wada@sap.ist.i.kyoto-u.ac.jp
[b)] yoshiaki@sap.ist.i.kyoto-u.ac.jp
[c)] enakamura@sap.ist.i.kyoto-u.ac.jp
[d)] itoyama@sap.ist.i.kyoto-u.ac.jp
[e)] yoshii@sap.ist.i.kyoto-u.ac.jp

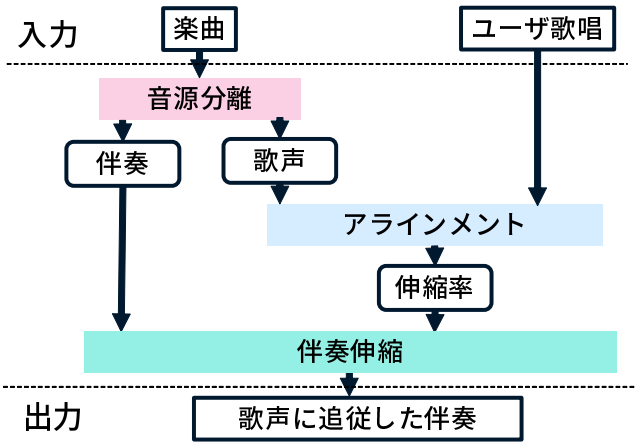

図 2　提案システムの実装概要図.

として歌唱のテンポが動的に変化することが多い．ユーザがそのような表現を意図した場合に，伴奏音のテンポを自分の歌唱に合うように手動で変更するのは手間がかかる．

これらの問題を解決するため，本稿では音楽音響信号からオンラインに伴奏音を抽出し，それをユーザ歌唱のテンポ変化に自動で同期させて再生するカラオケシステムを提案する．図 1 は，ユーザが実際に提案システムを使用する様子を撮影したものである．ユーザが歌いたい曲を選択すると，すぐにその音源からの伴奏音の分離が開始し，ユーザは伴奏音を聞きながら歌を歌える．ユーザが歌唱のテンポを速くしたり遅くしたりすると，伴奏音のテンポもそれに合わせて変化する．画面にはユーザ歌唱と元の楽曲それぞれのスペクトログラムおよび音程 (基本周波数 F0) が表示され，ユーザはそれらをリアルタイムに比較できる．提案システムを使用するにあたってユーザが用意しなければならないのは，自分が歌いたい曲の音源のみである．

提案システムは，3 つの構成要素から成る．1 つ目は，歌声分離によってカラオケの伴奏音を生成する部分である．2 つ目は，歌声同士の audio-to-audio アラインメントを計算する部分である．3 つ目は，伴奏音を時間方向に伸縮する部分である．提案システムの概要は，図 2 に示されている．まず，入力された音楽音響信号から，RNMF [2] のオンライン版を用いて伴奏音が分離される．次に，ユーザ歌唱と分離された歌唱同士をオンライン DTW によって時間方向に同期することで，伴奏音の伸縮率が計算される．最後に，計算された伸縮率に従って伴奏音が時間方向に伸縮され，再生される．これらの処理は並列に実行されるため，ユーザは歌声分離の処理時間を気にせず歌唱を楽しめる．

本研究の主な技術的貢献は，歌声同士のリアルタイム audio-to-audio アラインメントに取り組んだことである．歌声の音程，音色，テンポはどれも時間ごとに著しく変化するため，歌声信号同士の直接的なアラインメントは困難な問題である．事実，これまでの歌声アラインメントに関する研究は，歌声信号と，楽譜や歌詞などの記号的情報のアラインメントに注目したものがほとんどである．また，もう一つの貢献は，この基礎技術を，伴奏追従型カラオケシステムという実用的な技術に応用したことである．

2. 関連研究

本章では，歌声情報処理および自動伴奏に関する研究について述べる．

2.1 カラオケシステム

立花ら [3] は，楽譜や歌詞の情報を用いず，音楽音響信号のみから伴奏音を生成するカラオケシステムを提案した．このシステムでは，歌声抑圧技術を用いて音楽音響信号から伴奏音が生成される．また，伴奏音のピッチを手動で変更できる．その他に，井上ら [4] は，伴奏音のテンポがユーザ歌唱に自動で追従するカラオケシステムを提案した．このシステムは，入力に楽譜と歌詞の情報を必要とし，MIDI ファイルから伴奏音を合成して再生する．

2.2 自動伴奏

これまで，自動伴奏に関して数多くの研究がなされている [5–10]．自動伴奏に関する研究のうち初期のものには，Dannenberg [5] によって提案された，動的計画法によるオンラインでの自動伴奏システムや，Vercoe [6] によって提案された，ライブ演奏に対するリアルタイム自動伴奏システムがある．その後，統計的手法に基づく自動伴奏システムが多く提案された．Raphael [7] は，与えられた楽曲に対して，隠れマルコフモデル (HMM) を用いて最適な楽譜片の割り当てを推定する手法を提案した．Cont [8] は，ライブ演奏に対する楽譜位置推定とテンポ推定を，HMM および隠れセミマルコフモデル (HSMM) を用いて同時に行う手法を提案した．中村ら [9] は，楽器演奏において弾き直しおよび弾き飛ばしが生じたとき，その前後において，楽譜位置に対する事前分布は独立であるという仮定を置いた高速な楽譜追跡アルゴリズムを提案した．Montecchio ら [10] は，パーティクルフィルタを用いて，楽譜情報なしにリアルタイムに多重音同士のアラインメントを行う手法を提案した．

2.3 歌声アラインメント

歌声信号と，楽譜および歌詞のアラインメントについて，これまでに多くの研究がなされている [11–15]．Gong ら [11] は，メロディと歌詞の情報を用いた，HSMM による歌声と楽譜のアラインメント手法を提案した．藤原ら [12] は，歌声分離および音素アラインメントを用いて，音楽音響信号とそれに対応する歌詞のアラインメントを行った．Iskandar ら [13] は，動的計画法に基づく音節レベルでの歌声信号と歌詞のアラインメント手法を提案した．Wang ら [14] は，歌声から抽出した特徴量と，音楽音

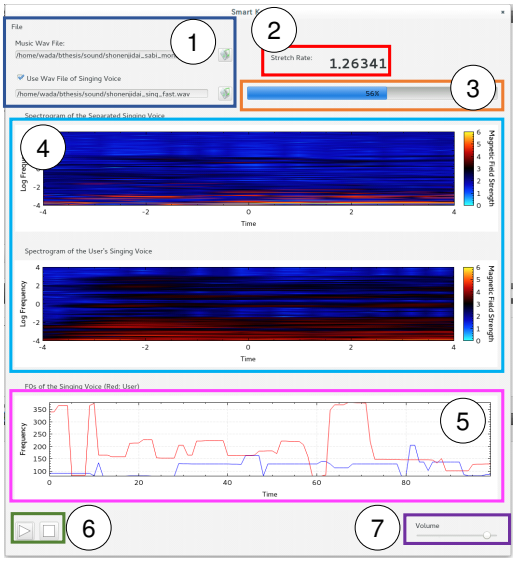

図 3　提案システムのユーザインターフェース．

響信号から推定したリズム構造の情報を組み合わせて音楽音響信号と歌詞のアラインメントに利用する手法を提案した．Dzhambazov ら [15] は，メル周波数ケプストラム係数 (MFCC) を観測とする HMM を用いて，歌声中の音素の間隔を明示的にモデル化する手法を提案した．

2.4　歌声分離

歌声分離に関して，入力された混合音スペクトログラムを，歌声と伴奏それぞれのスペクトログラムに分離するような時間-周波数領域のマスクを推定する手法 [16–19] が広く用いられている．Huang ら [16] は，ロバスト主成分分析 (RPCA) を用いて伴奏音スペクトログラムを低ランク行列で近似する手法を提案した．池宮ら [17] は，RPCA を用いた歌声分離と，歌声に対する F0 推定を相補的に行うことで，分離精度の向上を達成した．Rafii ら [18] は，類似度に基づいて混合音中の伴奏音の繰り返し構造を推定する手法を提案した．Yang ら [19] は，ベイジアン非負値行列分解を用いた手法を提案した．この他に，再帰型ニューラルネットワークを用いた手法 [20] も提案されている．このように，入力された混合音全体に対する歌声分離手法は多数提案されているものの，オンラインで動作する歌声分離手法に関する研究は少ない．

3.　提案システム

本章では，まず提案システムのユーザインターフェースについて述べる．次に，歌声分離および歌声信号同士の audio-to-audio アラインメントからなる提案システムの実装について述べる．

3.1　ユーザインターフェース

図 3 は，提案システムのユーザインターフェースを表す．

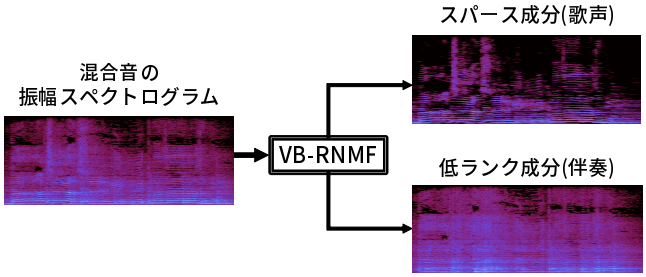

図 4　VB-RNMF を用いた歌声分離の概要図．混合音のスペクトログラムに対応する行列が，歌声に対応するスパース行列と，伴奏音に対応する低ランク行列に分解される．

このユーザインターフェースは，(1) 音楽ファイルの選択，(2) 現在の伴奏音の伸縮率の表示，(3) 歌声分離の進行状況の表示，(4) ユーザ歌唱及び分離された歌声のスペクトログラムの表示，(5) ユーザ歌唱及び分離された歌声の F0 軌跡の表示，(6) 伴奏音の再生と停止，(7) 伴奏音の音量の調節という 7 つの機能を備えている．

図 3 中の 2, 4, 5 番の要素は，ユーザ歌唱および分離された歌唱に対する視覚的なフィードバックを提供する．図 3 中の 3 番の，赤色の枠で囲われた部分によって，現在の伸縮率がユーザの意図にどれだけ合っているかを確認できる．また，図 3 中の 4 番の，水色の枠で囲われた部分に表示されたスペクトログラムを見ることによって，元の音源において歌手がどのように歌っているかを視覚的に捉えられる．例えば，原曲の歌手がビブラートをかけて歌っている部分が視覚的に分かるようになる．これらに加えて，図 3 中の 5 番の，ピンク色の枠で囲われた部分に表示された F0 軌跡を見ることによって，ユーザは自分の歌唱のピッチがどれだけ原曲と合っているかを確認できる．

3.2　実装方針

提案システム使用時のユーザの待ち時間を削減し，ユーザに提案システムを快適に利用してもらうため，我々はシステムの実装に際して 3 つの要件を定めた．1 つ目は，ユーザがシステムを起動してからすぐにカラオケを楽しめることである．2 つ目は，歌声分離が事前学習なしにリアルタイムで動作することである．3 つ目は，伴奏の自動追従もまたリアルタイムで動作することである．

我々は，これら 3 つの要件を満たすように，システムの各部分に用いる手法を選択・実装した．より詳細には，歌声分離，ユーザ歌唱の録音，歌声信号同士のアラインメント，追従した伴奏音の再生のそれぞれが独立したスレッドで行われるように実装した．

3.3　音楽音響信号に対する歌声分離

ユーザが指定した音楽音響信号を，歌声と伴奏音のそれぞれに分離するにあたって，我々は変分ベイズロバスト NMF (VB-RNMF) [2] のオンライン版を提案する．バッチ

での歌声分離に関しては，これまで数多くの手法が提案されている [16–19] ものの，提案システムでは歌声分離の処理時間をユーザから隠蔽するため，リアルタイムで動作する手法が必要となる．図 4 は，VB-RNMF が入力された混合音のミニバッチスペクトログラムを歌声のスパーススペクトログラムと伴奏音の低ランクスペクトログラムに分解する様子を表す．

以下に，VB-RNMF の定式化を説明する．VB-RNMF では，式 1 に示すように，入力された混合音の振幅スペクトログラム $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T]$ は，低ランクスペクトログラム $\mathbf{L} = [\boldsymbol{l}_1, \ldots, \boldsymbol{l}_T]$ と，スパーススペクトログラム $\mathbf{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_T]$ の和で近似される．

$$\boldsymbol{y}_t \approx \boldsymbol{l}_t + \boldsymbol{s}_t \tag{1}$$

また，低ランク成分 $\mathbf{L}$ は，2 のように，K 個の基底ベクトル $\mathbf{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K]$ と，それらのアクティベーション $\mathbf{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_T]$ の積で表される．

$$\boldsymbol{y}_t \approx \mathbf{W}\boldsymbol{h}_t + \boldsymbol{s}_t \tag{2}$$

低ランク性とスパース性の度合いは，以下に示すようなベイズ推定の枠組みによって決定される．近似の誤差を表す指標として，Kullback-Leibler (KL) ダイバージェンスを用いる．ポアソン分布で表される尤度 ($\mathcal{P}$ とする) の最大化は，KL ダイバージェンスの最小化と等価であるため，尤度関数は，式 3 のように表される．

$$p(\mathbf{Y}|\mathbf{W}, \mathbf{H}, \mathbf{S}) = \prod_{f,t} \mathcal{P}\left(y_{ft} \middle| \sum_k w_{fk} h_{kt} + s_{ft}\right) \tag{3}$$

ガンマ分布 ($\mathcal{G}$ とする) はポアソン分布の共役事前分布であるため，低ランク成分における基底およびアクティベーション行列には，それぞれガンマ分布の事前分布を式 4，5 のように置く．

$$p(\mathbf{W}|\alpha^{wh}, \beta^{wh}) = \prod_{f,k} \mathcal{G}(w_{fk}|\alpha^{wh}, \beta^{wh}) \tag{4}$$

$$p(\mathbf{H}|\alpha^{wh}, \beta^{wh}) = \prod_{k,t} \mathcal{G}(h_{kt}|\alpha^{wh}, \beta^{wh}) \tag{5}$$

ここで，α^{wh} および β^{wh} は，ガンマ分布の形状母数および尺度母数である．

スパース成分に関しては，それらが非負となるように，ハイパーパラメータに対する Jeffreys 事前分布を置いたガンマ事前分布を式 6，7 のように用いる．

$$p(\mathbf{S}|\alpha^s, \boldsymbol{\beta}^s) = \prod_{f,t} \mathcal{G}(s_{ft}|\alpha^s, \beta^s_{ft}), \tag{6}$$

$$p(\beta^s_{ft}) \propto (\beta^{s_{ft}})^{-1}. \tag{7}$$

ここで，α^s は，ガンマ分布のスパース性を調節するハイパーパラメータである．式 (3)–(7) を用いて，$\mathbf{W}$，$\mathbf{H}$ および $\mathbf{S}$ が変分ベイズ法によってミニバッチごとに推定される．

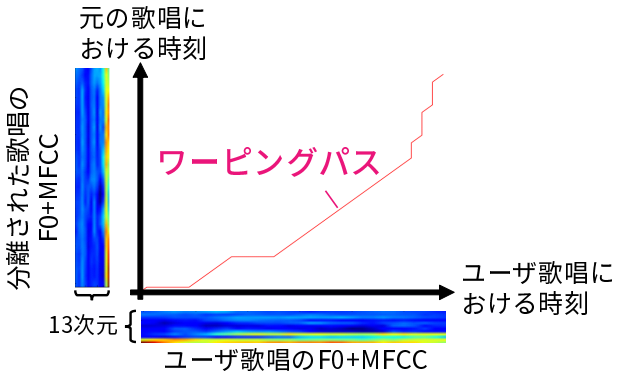

図 5　オンライン DTW で得られるワーピングパスの例．

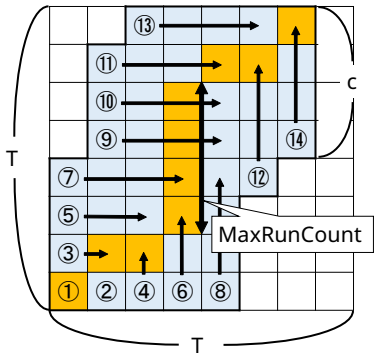

図 6　入力長 $T = 8$ に対して，パラメータを $c = 4$, MaxRunCount = 4 としたときのオンライン DTW の動作例．コスト行列が計算された部分は太枠で囲われた水色で表され，推定されたワーピングパスは橙色で表されている．

3.4　歌声アラインメント

ユーザ歌唱と分離された歌声の audio-to-audio アラインメントは，図 5 に示すような，ユーザ歌唱と分離された歌声に対する最適なワーピングパスを推定する．そのための手法として，提案システムでは，オンライン DTW [21] を用いる．オンライン DTW への入力に用いる特徴量として，提案システムでは，F0 と MFCC の 2 つを組み合わせて用いる．ユーザ歌唱のピッチ情報 (F0) および音韻情報 (MFCC) は，どちらもそのユーザの歌唱力やアレンジによって元の歌唱と大きくかけ離れることがある．そこで，F0 と MFCC の両方を組み合わせて特徴量として用いることで，どちらか一方が元の歌唱と違っていても正しく推定が行われることを狙いとしている．F0 の推定には，Subharmonic Summation [22] を用いる．

以下に，提案システムにおける歌声信号同士の audio-to-audio アラインメントの詳細を述べる．まず，ユーザ歌唱のミニバッチスペクトログラム $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots \boldsymbol{x}_T\}$ および分離された歌唱のミニバッチスペクトログラム $\boldsymbol{Y} = \{\boldsymbol{y}_1, \ldots \boldsymbol{y}_T\}$ から，F0 および MFCC を抽出する．ユーザ歌唱から抽出された F0 軌跡を $f_X = \{f_1^{(x)}, \ldots, f_T^{(x)}\}$ とし，MFCC を $\boldsymbol{m}_X = \{\boldsymbol{m}_1^{(x)}, \ldots, \boldsymbol{m}_T^{(x)}\}$ とする．同様に，分離された歌

Algorithm 1 オンライン DTW アルゴリズム

```
t ← 0, j ← 0 で初期化
ワーピングパスに (t, j) を追加
while t < T, j < T do
  if GetInc(t, j) ≠ Column then
    t ← t + 1
    for k = j − c + 1, ..., j do
      if k > 0 then
        式 (8) に従って d_{t,k} を計算
      end if
    end for
  end if
  if GetInc(t, j) ≠ Row then
    j ← j + 1
    for k = t − c + 1, ..., t do
      if k > 0 then
        式 (8) に従って d_{k,j} を計算
      end if
    end for
  end if
  if GetInc(t, j) == previous then
    runCount ← runCount +1
  else
    runCount ← 1
  end if
  if GetInc(t, j) ≠ Both then
    previous ← GetInc(t, j)
  end if
  ワーピングパスに (t, j) を追加
end while
```

Algorithm 2 FUNCTION GetInc (t, j)

```
if t < c then
  return Both
end if
if runCount < MaxRunCount then
  if previous == Row then
    return Column
  else
    return Row
  end if
end if
(x, y) = arg min(D(k, l)), where k == t or l == j
if x < t then
  return Row
else if y < j then
  return Column
else
  return Both
end if
```

唱から抽出された F0 軌跡を $f_Y = \{f_1^{(y)}, \ldots, f_T^{(y)}\}$ とし，MFCC を $\boldsymbol{m}_Y = \{\boldsymbol{m}_1^{(y)}, \ldots, \boldsymbol{m}_T^{(y)}\}$ とする．提案システムでは，MFCC の次元数は 12 とする．抽出した F0 と MFCC を組み合わせたベクトルを特徴量とし，ユーザ歌唱に対する特徴量を $\boldsymbol{X}' = \{\boldsymbol{x}'_i\}_{i=1}^T = \{f_i^{(x)}, \boldsymbol{m}_i^{(x)}\}_{i=1}^T$，分離された歌唱に対する特徴量を $\boldsymbol{Y}' = \{\boldsymbol{y}'_i\}_{i=1}^T = \{f_i^{(y)}, \boldsymbol{m}_i^{(y)}\}_{i=1}^T$ とする．すなわち，この特徴量の次元は 13 次元となる．

次に，ユーザ歌唱および分離された歌唱から抽出した特徴量を，オンライン DTW を用いて時間方向に同期させる．オンライン DTW によって，入力された時系列に対する最適なワーピングパスが，コスト行列をバックトラックすることなく求められる．図 6 に，オンライン DTW アルゴリズムによるワーピングパス計算の例を示す．図 6 中に記された丸数字は，コスト行列を計算した順番を表し，そこから伸びる矢印は，コスト行列の成分がどの方向に計算されたかを示している．オンライン DTW は，入力された特徴量を用いて，アルゴリズム 1 に従ってコスト行列 $D = \{d_{i,j}\}(i = 1, \ldots, T; j = 1, \ldots, T)$ を更新する．アルゴリズム 1 中で用いられる，ワーピングパスが進む方向を決定する関数 GetInc は，アルゴリズム 2 に示した．アルゴリズム 1 中の各パラメータについて，(t, j) はコスト行列中の現在位置である．c は，ワーピングパスを求める際にどれだけの範囲のコスト行列を計算するかを決定するパラメータであり，現在位置 (t, j) から左または下 c 個分のコスト行列の要素を計算する．どちらの方向を計算するかは，関数 GetInc の出力によって決定される．runCount は，ワーピングパスが同じ方向に連続してどれだけ進んだかを表すパラメータであり，この値が閾値 MaxRunCount に達すると，ワーピングパスはそれ以上同じ方向に進まなくなる．提案システムでは，各パラメータの値として，$T = 300, c = 4, \mathrm{MaxRunCount} = 3$ を用いた．アルゴリズム 1 におけるコスト行列の成分の計算は，式 8 に従って行われる．

$$d_{i,j} = ||\boldsymbol{x}'_i - \boldsymbol{y}'_j|| + \min(d_{i,j-1}, d_{i-1,j}, d_{i-1,j-1}) \quad (8)$$

式 8 中の $||\boldsymbol{x}'_i - \boldsymbol{y}'_j||$ は，$\boldsymbol{x}'_i$ と $\boldsymbol{y}'_j$ の距離を表し，提案システムでは二乗平均平方根 $||\boldsymbol{x}'_i - \boldsymbol{y}'_j|| = \sqrt{\sum_{k=1}^{13}(x'_{ik} - y'_{jk})^2}$ を用いた．このオンライン DTW アルゴリズムにより，最適なワーピングパス $L = \{(i_1, j_1), \ldots, (i_l, j_l)\}(0 \leq i_k \leq i_{k+1} \leq T, 0 \leq j_k \leq j_{k+1} \leq T)$ が得られる．ワーピングパス中の (i_k, j_k) は，オンライン DTW に入力された特徴量 $\boldsymbol{X}'$ および $\boldsymbol{Y}'$ のうち，$\boldsymbol{x}'_{i_k}$ と $\boldsymbol{y}'_{j_k}$ が対応づけられるということを意味する．

3.5 伴奏音の伸縮

提案システムでは，オンライン DTW によって推定されたワーピングパス L から，伴奏音のミニバッチスペクトログラムの各フレームに対する伸縮率の系列 $R = \{r_1, \ldots, r_T\}$ を計算する．伴奏音のミニバッチスペクトログラムの k 番目のフレームに対する伸縮率 r_k は，式 9 に従って計算される．

$$r_k = \frac{\{i_1, \ldots, i_l\} \text{ 中の } k \text{ の個数}}{\{j_1, \ldots, j_l\} \text{ 中の } k \text{ の個数}} \quad (9)$$

各 r_k から，伴奏音のミニバッチスペクトログラム全体に

	質問 (1)	質問 (2)
被験者 1	ややそう思う	ややそう思う
被験者 2	そう思う	ややそう思う
被験者 3	ややそう思う	そう思わない
被験者 4	そう思う	ややそう思う

表 1 被験者実験における, 質問に対する被験者の方々の回答.

対する伸縮率 r は，$R = \{r_1, \ldots, r_T\}$ の中央値として計算される．これは，外れ値により全体の伸縮率がユーザの意図しないものとなるのを避けるためである．

以上のようにして計算された伸縮率 r に従って，提案システムは伴奏音のミニバッチスペクトログラムを時間方向に伸縮する．伸縮にはフェーズボコーダ [23] を用いる．

4. 評価実験

提案システムの有効性を確認するため，被験者実験を行った．4 名の被験者の方々に，歌いたい楽曲を自由に挙げてもらい，その楽曲を用いて実際に提案システムを使用してもらった．評価に使用された楽曲は，「日立の樹」 (CM ソング),「リライト」(ASIAN KUNG-FU GENERATION),「少年時代」 (井上陽水),「きまぐれロマンティック」 (いきものがかり) の 4 曲である．システム使用後，被験者の方々に対して，(1) 伴奏音の追従は正確に行われていたか，(2) ユーザインターフェースは適切であったかという 2 つの質問を行い，それぞれに対して，1. そう思う，2. ややそう思う，3. あまりそう思わない，4. そう思わない，の 4 段階で回答してもらった．2 つの質問に対する被験者の方々の回答は，表 1 に示した通りである．この結果から，伴奏の追従は概ね正確であり，ユーザインターフェースは概ね適切であることが示された．

また，被験者の方々から，提案システムに対する自由意見を収集した．その結果，まず，伴奏音の品質が低く，伴奏音が自分の歌唱のテンポ変化に適切に追従しているかどうか分からなかったという意見が得られた．この問題に対して，まずは歌声分離の品質を定量評価する必要がある．この問題の解決策として，バッチの歌声分離手法を用いるということが考えられる．それによって，ユーザに対して処理の待ち時間を生じさせてしまうが，伴奏音の音質向上が期待される．また，ユーザインターフェースに表示されるスペクトログラムが何を意味するか分からなかったという意見が得られた．スペクトログラムは多くの有用な情報を含むものの，ユーザの視点に立つと，現在の伸縮率および F0 軌跡のみを表示することを検討する必要がある．

その他に，被験者実験の人数が少ないという問題があるため，今後さらに人数を増やして実験を行う必要がある．

5. おわりに

本稿では，音楽音響信号から伴奏音を分離し，ユーザ歌唱のテンポ変化に自動で追従させて再生するカラオケシステムを提案した．提案システムの主な構成要素は，オンライン VB-RNMF および歌声同士のオンライン DTW による audio-to-audio アラインメントである．提案システムにより，ユーザは自分が歌いたい任意の曲を，楽譜を用意することなくテンポを自由にアレンジしながら歌うことが可能になる．被験者実験の結果より，提案システムの有効性が確認された．

今後は，歌声アラインメントのさらなる精度向上に取り組む予定である．audio-to-audio アラインメントにテンポ推定の結果を取り入れることで，アラインメントの精度向上が期待できる．その他に，提案システムの発展として，ユーザ歌唱に対して自動でハモリパートを生成・付与する機能や，ユーザの歌唱履歴から苦手な歌唱表現を分析し，ユーザの歌唱力向上に役立てる機能の開発を行いたい．

謝辞 本研究の一部は、JSPS 科研費 26700020, 24220006, 26280089, 15K16654, 16H01744, 16J05486 および JST ACCEL No. JPMJAC1602 の支援を受けた．

参考文献

[1] Hamasaki, M. et al.: Songrium: Browsing and Listening Environment for Music Content Creation Community, *Proc. SMC*, pp. 23–30 (2015).
[2] Bando, Y. et al.: Variational Bayesian Multi-channel Robust NMF for Human-voice Enhancement with a Deformable and Partially-occluded Microphone Array, *Proc. EUSIPCO*, pp. 1018–1022 (2016).
[3] Tachibana, H. et al.: A Real-time Audio-to-audio Karaoke Generation System for Monaural Recordings Based on Singing Voice Suppression and Key Conversion Techniques, *J. IPSJ*, Vol. 24, No. 3, pp. 470–482 (2016).
[4] Inoue, W. et al.: Adaptive Karaoke System: Human Singing Accompaniment Based on Speech Recognition, *Proc. ICMC*, pp. 70–77 (1994).
[5] Dannenberg, R. B.: An On-Line Algorithm for Real-Time Accompaniment, *Proc. ICMC*, pp. 193–198 (1984).
[6] Vercoe, B.: The Synthetic Performer in The Context of Live Performance, *Proc. ICMC*, pp. 199–200 (1984).
[7] Raphael, C.: Automatic Segmentation of Acoustic Musical Signals Using Hidden Markov Models, *IEEE Trans. on PAMI*, Vol. 21, No. 4, pp. 360–370 (1999).
[8] Cont, A.: A Coupled Duration-focused Architecture for Realtime Music to Score Alignment, *IEEE Trans. on PAMI*, Vol. 32, No. 6, pp. 974–987 (2010).
[9] Nakamura, T. et al.: Real-Time Audio-to-Score Alignment of Music Performances Containing Errors and Arbitrary Repeats and Skips, *IEEE/ACM TASLP*, Vol. 24, No. 2, pp. 329–339 (2016).
[10] Montecchio, N. et al.: A Unified Approach to Real Time Audio-to-score and Audio-to-Audio Alignment Using Sequential Montecarlo Inference Techniques, *Proc. ICASSP* (2011).
[11] Gong, R. et al.: Real-time Audio-to-Score Alignment of Singing Voice Based on Melody and Lyric Information, *Proc. Interspeech* (2015).
[12] Fujihara, H. et al.: LyricSynchronizer: Automatic Synchronization System between Musical Audio Signals and Lyrics, *Proc. IEEE Journal of Selected Topics in Signal*

Processing Conference, pp. 1252–1261 (2011).
[13] Iskandar, D. et al.: Syllabic Level Automatic Synchronization of Music Signals and Text Lyrics, *Proc. ACMMM*, pp. 659–662 (2006).
[14] Wang, Y. et al.: LyricAlly: Automatic Synchronization of Textual Lyrics to Acoustic Music Signals, *IEEE TASLP*, Vol. 16, No. 2, pp. 338–349 (2008).
[15] Dzhambazov, G. et al.: Modeling of Phoneme Durations for Alignment between Polyphonic Audio and Lyrics, *Proc. SMC*, pp. 281–286 (2015).
[16] Huang, P.-S. et al.: Singing-Voice Separation from Monaural Recordings Using Robust Principal Component Analysis, *Proc. IEEE ICASSP*, pp. 57–60 (2012).
[17] Ikemiya, Y. et al.: Singing Voice Separation and Vocal F0 Estimation Based on Mutual Combination of Robust Principal Component Analysis and Subharmonic Summation, *IEEE/ACM TASLP*, Vol. 24, No. 11, pp. 2084–2095 (2016).
[18] Rafii, Z. et al.: Music/Voice Separation Using The Similarity Matrix, *Proc. ISMIR*, pp. 583–588 (2012).
[19] Yang, P.-K. et al.: Bayesian Singing-Voice Separation, *Proc. ISMIR*, pp. 507–512 (2014).
[20] Huang, P.-S. et al.: Singing-Voice Separation from Monaural Recordings Using Deep Recurrent Neural Networks, *Proc. ISMIR*, pp. 477–482 (2014).
[21] Dixon, S.: An On-Line Time Warping Algorithm for Tracking Musical Performances, *Proc. the 19th IJCAI*, pp. 1727–1728 (2005).
[22] Hermes, D. J.: Measurement of Pitch by Subharmonic Summation, *J. ASA*, Vol. 83, No. 1, pp. 257–264 (1988).
[23] Flanagan, J. et al.: Phase Vocoder, *Bell System Technical Journal*, Vol. 45, pp. 1493–1509 (1966).