リアルタイム DNN 音声変換の実装とデータ拡張法による音質改善法*

☆ 荒川 陸 (東京大学), 高道 慎之介, 猿渡 洋 (東大院・情報理工)

1 はじめに

音声変換は入力音声に含まれる言語情報を保持したまま, ある話者が話したパラ言語・非言語情報を, 別の話者が話したかのように変換する技術である. この技術は, 発話者の発声器官の物理的制約を超えて, 多様な音声コミュニケーションを可能にするという点で期待される. 入力音声特徴量と出力音声特徴量の対応関係を, 統計的な音響モデルで表現する統計的パラメトリック音声合成方式は, その汎用性の高さから広く研究されている. Deep Neural Networks (DNN) に基づく非線形な特徴量変換を用いた音声変換は, 従来の Gaussiann Mixture Model (GMM) を用いた音声変換手法 [1] の区分線形変換より高音質な音声変換が可能であることから広く期待されている.

一方で既存の DNN 音声変換手法は, バッチ型の特徴量分析および変換 (例えば, 双方向 recurrent neural network などの非因果的モデル) や波形合成プロセスを有しているため, オンラインでの変換, さらには遅延聴覚フィードバックよりも低遅延な DNN 音声変換技術は確立されていない. そのため本稿では, 従来のリアルタイム GMM 音声変換システム [2] の枠組みを踏襲・改善した実装を提示し, リアルタイム DNN 音声変換アルゴリズムを確立する. 次に音声認識や音響イベント分類といったタスクにおいて DNN の汎化性向上に有効であると確認されている信号処理的データ拡張法を, リアルタイム DNN 音声変換に導入する. 実験的評価により, 提案するリアルタイム DNN 音声変換が, Real Time Factor (RTF) 0.50, アルゴリズム遅延 50 ms で動作すること, およびデータ拡張法により合成音声の品質を改善できることを示す.

2 従来のリアルタイム GMM 音声変換

Toda らはバッチ型の GMM 音声変換 [1] を, フレームごとの特徴量変換に基づきオンライン化することでリアルタイム音声変換を提案している [2]. GMM 音声変換における特徴量変換部は, 演算量の少ない区分線形変換であり, Fast Fourier Transform (FFT) 分析に基づくメルケプストラム分析と再帰的な最尤音声パラメータ生成により, アルゴリズム遅延 50 ms の音声変換システムを実現している. しかしながら, GMM の表現能力の低さから, 合成音声品質および音響モデルの汎化性向上のためのデータ拡張法の効果も限定的である.

3 提案するリアルタイム DNN 音声変換

提案するリアルタイム DNN 音声変換の枠組みを Fig. 1 に示す. Toda らの従来法 [1] を踏襲し, 本枠組みも特徴量分析部・特徴量変換部・波形合成部から構成される. まず特徴量分析部で音声波形から音声特徴量 $\{F_0$, パワー, メルケプストラム $\}$ を抽出する. 得られた特徴量は特徴量変換部で, 事前に学習したモデルを用いて, 出力話者の特徴量 $\{F_0$, パワー, 帯域平均化された非周期性指標, メルケプストラム $\}$ に変換される. 最後に得られた特徴量から波形合成部で音声波形を合成する. 全ての処理は再帰的かつ低遅延に行われる. 各処理の詳細を以下に記す.

3.1 解析部

音声波形から音声特徴量 $\{F_0$, パワー, メルケプストラム $\}$ を抽出する. まず, 従来法と同様に音声波形に対して窓かけを行い, 演算量の少ない FFT 分析に基づき, メルケプストラムと F_0 を計算する. パワーはメルケプストラムの 0 次とする. 次に, このようにして抽出したメルケプストラム時系列に対し, 高変調周波数の変調スペクトル成分を除去する trajectory smoothing [3] を適用する. この成分は音響モデリングが困難であること, および音声知覚への影響が小さいことから, 事前に除去することの有効性が知られている. 従来の trajectory smoothing では, タップ数の大きい非因果的なディジタルフィルタを利用しているが, 本稿はオンラインかつ低遅延システムを目的としているため, 時刻 -1 と 0 に値を持つタップ数 2 の FIR フィルタを用いる. Figure 1 に示す通り, これらの処理によるアルゴリズム遅延は, フレーム分析長の半分である.

3.2 変換部

元話者の音声特徴量から目標話者の音声特徴量を推定する. 具体的には,

1. 元話者 FFT メルケプストラム-目標話者 WORLD メルケプストラム間の変換
2. 元話者 FFT メルケプストラム-目標話者の帯域平均化非周期性指標間の変換
3. 元話者 F_0-目標話者 F_0 間の変換
4. 元話者 FFT パワー-目標話者 WORLD パワー間の変換

の 4 種の変換を行う. 3. および 4. の変換については, 従来の GMM 音声変換と同様に線形変換を適用する [1]. 例えば, 3. の F_0 の変換については, 事前に推定した平均対数 F_0 に基づき, log スケールでの線形変換を行う.

1. および 2. の変換には, 事前に学習した DNN を用いる. DNN の入力は, 当該フレームとその前後 C フレームの FFT メルケプストラム, およびその動的特徴量を結合したベクトルである. 1. の変換では, 目標話者 WORLD メルケプストラムの静的・動的特徴量, 2. の変換では, 目標話者の帯域平均化非周期性指標のみを出力する.

DNN の学習には, mean squared error (MSE) を用いる. すなわち, 出力話者の音声の特徴量系列 $\boldsymbol{y} = [\boldsymbol{y}_1^\top, \cdots, \boldsymbol{y}_t^\top, \cdots, \boldsymbol{y}_T^\top]^\top$ と, 推定された出力話者の音声の特徴量系列を $\hat{\boldsymbol{y}} = [\hat{\boldsymbol{y}}_1^\top, \cdots, \hat{\boldsymbol{y}}_t^\top, \cdots, \hat{\boldsymbol{y}}_T^\top]^\top$ と表記する時, DNN の学習時に最小化する損失関数は,

$$L_G(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{T}(\hat{\boldsymbol{y}} - \boldsymbol{y})^\top(\hat{\boldsymbol{y}} - \boldsymbol{y}) \tag{1}$$

*Implementation of DNN-based real-time voice conversion and its quality improvements by audio data augmentation, by Riku Arakawa, Shinnosuke Takamichi, Hiroshi Saruwatari (The University of Tokyo)

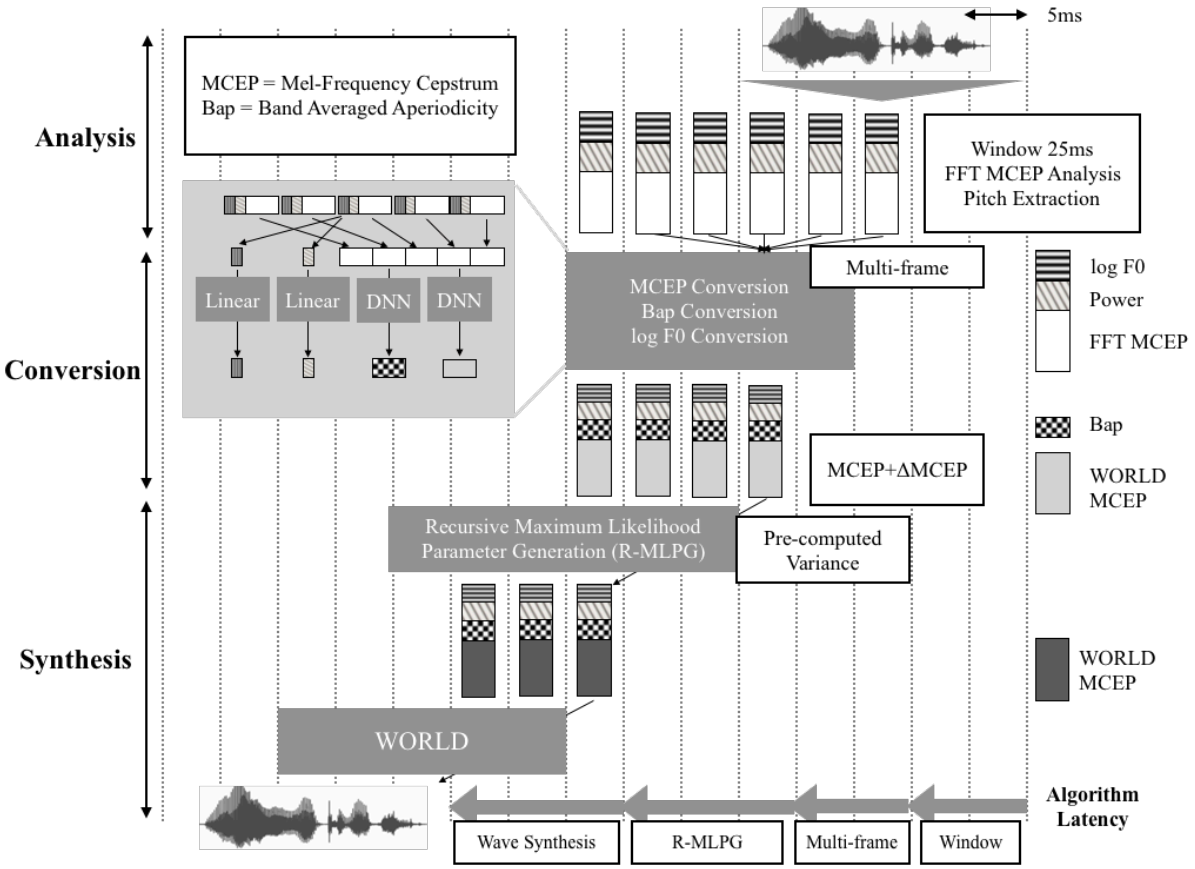

Fig. 1　リアルタイム DNN 音声変換のための提案手法

で与えられる. ここで, t はフレームインデックス, T はフレーム数である. 1. の変換では, input-to-output residual 構造 [4] を用いることで効果的な変換が可能である.

3.3　波形合成部

得られた音声特徴量から波形を生成する. 本稿では従来法と同様に, 推定した WORLD メルケプストラムに再帰的最尤パラメータ生成 [5] を適用し, メルケプストラムを生成する. この際, 事前に計算した共分散行列を使用する.

こうして得られたメルケプストラムに対し, 音声分析合成システム WORLD [6] (D4C edition [7]) の再帰的波形生成アルゴリズムを用いて, 音声波形を生成する. 以降では, 再帰的最尤パラメータ生成時に考慮するフレーム数を M, 再帰的波形生成時のバッファフレーム数を W とする.

3.4　アルゴリズム遅延

提案するリアルタイム DNN 音声変換におけるアルゴリズム遅延について整理する. フレームシフトを 5 ms, フレーム分析長を 25 ms, $C = 2$, $M = 3$, $W = 3$ (70 Hz の F_0 に対する 1 ピッチ波形のタップ長をカバーするフレーム数) とすると, 分析・変換・合成部の遅延はそれぞれ, 10 ms, 10 ms, 30 ms である (Fig. 1 参照).

4　音声変換のためのデータ拡張法

日常的な音声変換システムの利用を考えると, 単一の元話者であろうと日々変化する音声に対して, 変換の頑健性を有する必要がある. 直接的な方法は, 複数日に渡って音声を収録し学習データとすることであるが, 現実的ではない. そこで本稿では信号処理的なデータ拡張法を考える.

データ拡張法は, DNN の性能が学習データ量に応じて敏感に変わることを利用し, 人為的なデータの水増しで DNN の性能を改善する方法として知られており [8], その有効性は音声認識 [9], 音響イベント分類 [10, 11, 12], といった分野で広く確認されている.

本研究ではこれらの手法を音声変換の枠組みに適応し, その音質改善を確認する. 具体的には, 他タスクで有効性が確認されているピッチシフト・タイムストレッチに加え, リアルタイム DNN 音声変換のために新たに提案するタイムシフト法を用いる. Figure 2 は, データ拡張の各手法を図示したものである.

4.1　ピッチシフト (P-Sh)

単一の話者が同じテキストを発話した場合でも, 発話毎に F_0 レンジが異なる. また, FFT メルケプストラムは, F_0 の変動に対して頑健ではない. そこで, WSOLA アルゴリズム [13] とリサンプリングを用いて, 元話者の F_0 を定数シフトし, データ拡張を行う. ここで, ピッチシフトによって話速は変化しない.

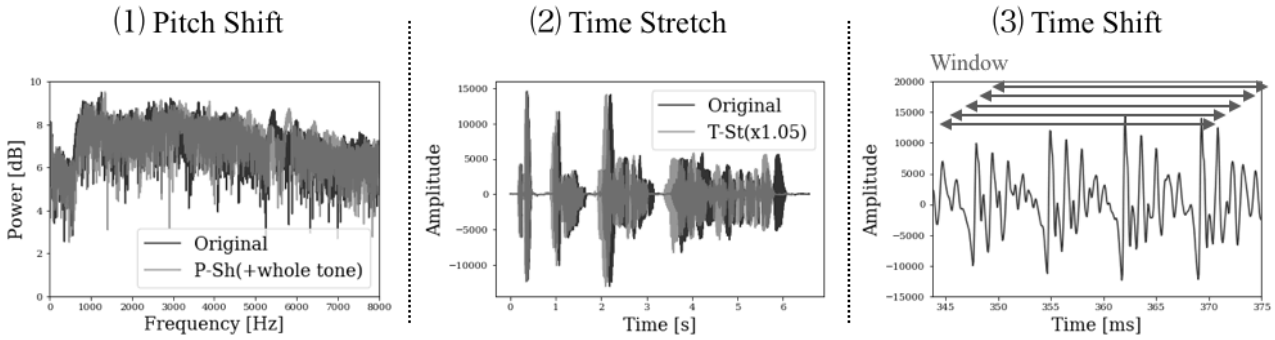

Fig. 2　各データ拡張法の例

4.2　タイムストレッチ (T-St)

4.1 節と同様に, 話速に対する摂動を信号処理的に与える. WSOLA アルゴリズムに基づいて, 音声波形の長さを波形ドメインで定数倍する.

4.3　タイムシフト (T-Sh)

定形の窓関数を用いた FFT メルケプストラムは, フレーム分析開始時刻によって異なる値が抽出される. しかし, リアルタイム音声変換では, そのような変動に対して頑健に動作する必要がある. そこで, 元話者音声波形における当該フレームの分析開始時刻に対し, フレームシフト長の超えない範囲（フレームシフト長が 5 ms なら, ±2.5 ms）のオフセットを与え, FFT メルケプストラムを抽出して元話者の学習データとする. なお, F_0 に応じた窓関数を設計することで分析開始時刻に依存しない分析が可能 [14] だが, 入力話者の F_0 レンジによっては過多のアルゴリズム遅延が生じるため, 本稿では採用しない.

5　実験的評価

5.1　実験条件

実験では声優統計コーパス (全 100 文, 約 12 分) を発話した男性 2 名の音声データをを用いた. 音声データのサンプリング周波数は 16 kHz であり, 入力話者, 出力話者それぞれの全発話の 9 割を学習に, 残りを評価に用いた.

学習する DNN は 500 次元, 2 層の中間層からなる Multi Layer Perceptron 構造であり, 活性化関数は Leaky ReLU を用い, 各活性化関数の直後には Batch Normalization [15] を適用した. 最適化手法は Adam [16] を用いた. 学習時には, 元話者と目標話者のメルケプストラムを, 平均 0, 分散 1 に正規化する.

データ拡張法について, ピッチシフトは { 半音・全音 },{ 低くする･高くする } の計 4 パターンを, タイムストレッチは 話速を {0.95, 1.05} 倍する計 2 パターンを適用した. タイムシフトはフレームシフト長を 5 等分して、開始点を摂動させた. なお, フレームシフト長は 5 ms, フレーム分析長は 25 ms である.

また本システムが CPU 環境下での利用が可能であることを目標とし, 処理遅延の評価では Intel (R) Core (TM) i7-3770K CPU @ 3.50 GHz を用いた.

5.2　リアルタイム DNN 音声変換の処理遅延

特徴量解析･特徴量変換･波形合成の各部について, CPU 処理により遅延が生じる. フレームシフト長分の音声波形の変換ごとに解析部・変換部・合成部においてかかった時間を計測し, 平均・標準偏差を算出した. Table 1 に各部での処理時間およびデコード時間と入力音声の比である RTF をまとめる.

Table 1　処理遅延

	解析部	変換部	合成部
処理時間 [ms]	0.85±0.092	1.12±0.51	0.57±0.10
RTF	0.17	0.22	0.11

表の通り, 提案するリアルタイム DNN 音声変換の RTF は 1.0 を下回っている. また, アルゴリズム遅延は 50 ms であるため, 本変換がリアルタイムで動作することを確認できる.

5.3　データ拡張法の効果

5.3.1　客観評価

提案したデータ拡張法の有効性を確認するため, DNN 学習の反復数に対する評価データの予測精度を計算する. Figure 3 に反復学習で使用した総フレーム数に対する, 変換音声と目標音声間のメルケプストラムの MSE を示す. ただし, ここでは正規化されたメルケプストラムを計算に利用している. また, DNN が十分学習された際の MSE の値を Table 2 にまとめる. “No” はデータ拡張法を使用していないもの, “All” は全てのデータ拡張法を使用したものを表す. この図と表から, データ拡張法間で大きな差異は見られないが, データ拡張により二乗誤差が劇的に改善されることが分かる.

5.3.2　主観評価

変換音声の品質評価のためにクラウドソーシングを利用した AB テストを行った. 参加者はリファレンスとして出力話者の自然音声を与えられ, 全てのデータ拡張法を用いて学習した変換音声と用いなかった変換音声のどちらがより自然音声に近いかをブラインドで選択した. 参加者は 35 人で, それぞれ 10 文の評価データについて評価をした.

主観評価の結果を Fig. 4 に示す. 自然性・話者性

Table 2　各データ拡張法を用いて学習した DNN に対する評価データの MSE

No	P-Sh	T-St	T-Sh	All
0.878	0.859	0.858	0.859	**0.853**

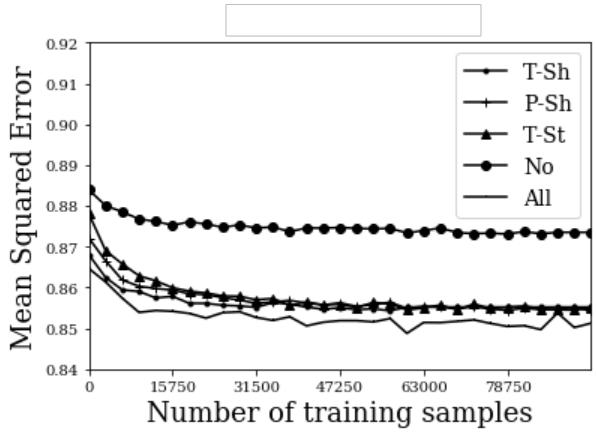

Fig. 3　学習に用いた総サンプル数に対する評価データの MSE

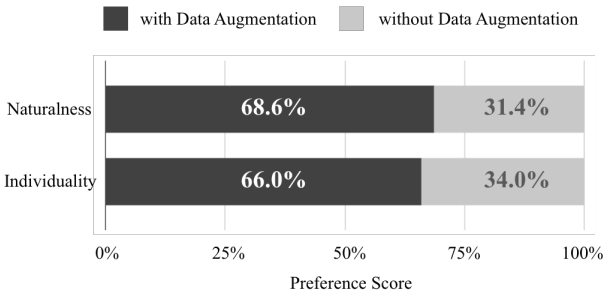

Fig. 4　主観評価結果

に対する評価それぞれにおいて p 値は 2.54×10^{-24}, 3.94×10^{-18} で有意にデータ拡張法の有効性が確認された.

6　まとめと考察

本稿ではまず, 従来研究のやり方を踏襲・改善する形で, リアルタイム DNN 音声変換システムの実装を提案した. 実験的評価により, RTF が 0.5, アルゴリズム遅延 50 ms と十分に小さいことを確認し, リアルタイムシステムとしての実現が可能であることを示した.

次に元話者の音声が発話のたびに変化することへの頑健性を高めるために, 信号処理的摂動を入力話者の学習データに加え, 学習データを水増しするデータ拡張法を提案した. 客観評価により, 元話者と目標話者のメルケプストラムの距離が小さくなるという点において, 各手法の有効性が確認された一方で, 個別の手法についての音質改善効果の比較は難しかった. また主観評価によってもデータ拡張法は有意にその音質改善効果が確認された.

謝辞: 本研究は, 東京大学情報理工学系研究科と合同会社 DMM.com(旧:株式会社 DMM.com ラボ) による社会連携講座「時空間解析技術の応用研究」において実施した.

参考文献

[1] T. Toda et al., "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[2] ——, "Implementation of computationally efficient real-time voice conversion," in *Proc. INTERSPEECH*, Portland, U.S.A., Sep. 2012.

[3] S. Takamichi et al., "The NAIST text-to-speech system for the Blizzard Challenge 2015," in *Proc. Blizzard Challenge workshop*, Berlin, Germany, Sep. 2015.

[4] Y. Saito et al., "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 755–767, Jun. 2018.

[5] K. Tokuda et al., "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, Istanbul, Turkey, Jun. 2000, pp. 1315–1318.

[6] M. Morise et al., "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 7, pp. 1877–1884, 2016.

[7] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Communication*, vol. 84, pp. 57 – 65, 2016.

[8] A. Krizhevsky et al., "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira et al., Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[9] T. Ko et al., "Audio augmentation for speech recognition," in *Proc. INTERSPEECH*, 2015, pp. 3586–3589.

[10] J. Salamon, J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Process. Letter*, vol. 24, no. 3, pp. 279–283, 2017.

[11] B. McFee et al., "A software framework for musical data augmentation," in *Proc. of ISMIR*, 2015, pp. 248–254.

[12] N. Takahashi et al., "Deep convolutional neural networks and data augmentation for acoustic event recognition," in *Proc. INTERSPEECH*, San Fransisco, U.S.A., pp. 2982–2986.

[13] W. Verhelst, M. Roelands, "An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech," in *Proc. ICASSP*, vol. 2, April 1993, pp. 554–557 vol.2.

[14] 森勢将雅, 音声分析合成 *(音響テクノロジーシリーズ 22)*. コロナ社, 2018.

[15] S. Ioffe, C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[16] D. P. Kingma, J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.