E-035

GPGPU による音声分析合成システム TANDEM-STRAIGHT の高速化

Acceleration of speech analysis/synthesis system TANDEM-STRAIGHT using GPGPU

森勢 将雅† 小澤 賢司†
Masanori Morise Kenji Ozawa

1．まえがき

TANDEM-STRAIGHT[1]は Vocoder 型[2]の音声分析合成システムであり，音声を基本周波数 (F0)，スペクトル包絡，非周期性指標に分解する．分解された 3 つのパラメタから元音声とほぼ等価な品質の音声が合成可能である．元音声を復元するに足る情報が含まれることから，歌声の個人性等，言語以外の情報を扱うための研究に利用されはじめている[3]．しかしながら，計算コストの問題により大規模な音声データベースを用いた統計処理は困難とされていた．TANDEM-STRAIGHT の高速化は，特に個人性や感情等従来の特徴量では認識が難しいパラ言語情報や非言語情報の認識研究における，認識精度の向上に貢献することとなろう．

本研究では TANDEM-STRAIGHT の高速化を目的とした検討を実施しており，本稿では，現在研究者向けに配布されている Matlab 版の TANDEM-STRAIGHT の高速化について述べる．TANDEM-STRAIGHT は音声波形を短時間で切り出し，切り出されたフレーム毎にパラメタを計算する．近年注目される GPU を利用した並列化は FFT 等の信号処理やフレーム毎の独立した処理の並列化と相性が良いことから，本稿でも GPGPU と並列化による高速化について取り組む．

本稿では，Matlab 版の TANDEM-STRAIGHT を，並列処理，および GPU による演算が可能な Parallel Computing Toolbox [4] (以下では PCT と記載する)を用いて高速化した結果について述べる．TANDEM-STRAIGHT は 3 つのパラメタ推定法から構成されるが，本稿では FFT を利用する F0 とスペクトル包絡推定に関する高速化について述べる．

2．TANDEM-STRAIGHT の概説と，GPU による高速化が可能な処理

ここでは TANDEM-STRAIGHT の F0 分析法と，スペクトル包絡推定法について述べ，並列化と GPU により高速化が可能な項目について説明する．なお，音声は時間とともに変化する時系列であることから，各パラメタは短時間毎のフレーム単位で計算する．TANDEM-STRAIGHT では各フレームの処理が独立しているため，全フレームの並列化が可能である．

2.1 F0 推定法

TANDEM-STRAIGHT で採用されている F0 推定法の概要を図 1 に示す．アルゴリズムの詳細については文献[1]を参照されたい．

TANDEM-STRAIGHT における 1 フレームの F0 推定演算では，まず，特定の周波数帯域の F0 候補とその候補の F0 らしさを計算する Extractor を推定する F0 の下限から上限まで配置することが行われる．全ての Extractor で F0 候補と F0 らしさを推定し，そのうち最も F0 らしさの高い F0 候補を最終的な F0 として採用する．

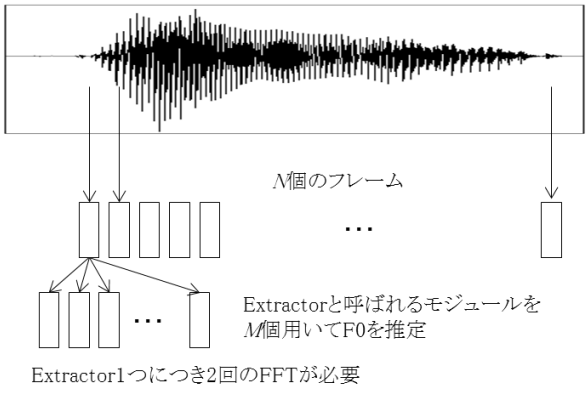

図 1：TANDEM-STRAIGHT の F0 推定の概要．1 フレームあたり M 個の Extractor を用いる．また，各 Extractor について FFT が 2 回実施される．

1 つの Extractor では，2 つの異なる時刻で切り出された波形のパワースペクトルを平均するプロセスがあることから，2 回の FFT が行われる．本実装では，この FFT を GPU による FFT に置き換えることで高速化する．F0 推定では，フレーム数を N，Extractor の数を M とした場合，$N \times M$ の処理が並列化されることとなる．

2.2 スペクトル包絡推定法

TANDEM-STRAIGHT のスペクトル包絡推定は，詳細は文献[1]に記載されているが概ね図 2 のように行われる．図 2 には，並列化が可能な処理と FFT に関する処理のみ記載している．F0 推定と同様にフレーム単位でスペクトル包絡を推定するが，各フレームに関する処理には逐次処理が含まれる．

本実装では，N フレームに関する並列化，および 4 回の FFT を GPU により処理させることで高速化を実現している．

3．高速化の工夫

GPGPU プログラミングでは C 言語が一般的に用いられるが，本稿では，TANDEM-STRAIGHT が Matlab で配布されていることを鑑み，Matlab 版での高速化について述べる．ここでは，Matlab の PCT，特に FFT に関する速度について，メモリ転送を含む処理全体の観点から述べる．

TANDEM-STRAIGHT では，音声のサンプリング周波数により FFT 長が決定する．ここでは，サンプリング周波数が 22.05 kHz, 44.1 kHz のデフォルト値である 2048, 4096 点の FFT について議論する．実験では Matlab R2013a，CPU は Intel の Xeon E5-2690 2.9 GHz，GPU は NVIDIA の Tesla C2075 を用いた．

図 3 は，FFT に必要な 3 つの処理 (CPU から GPU への

† 山梨大学工学部コンピュータ理工学科

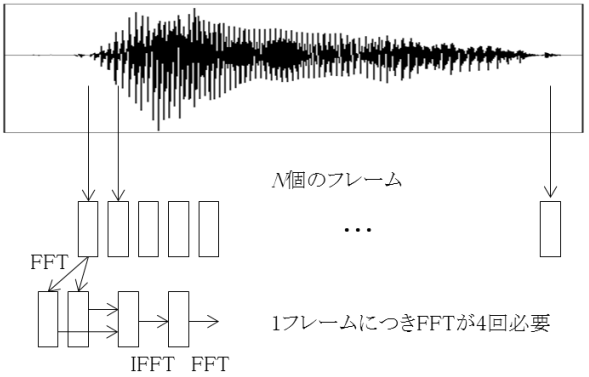

図 2：TANDEM-STRAIGHT のスペクトル包絡推定の概要．1 フレームにつき 4 回の FFT が行われる．最初の 2 回の FFT は並列化が可能であるが，残り 2 回の FFT は逐次処理である．

メモリ転送，FFT，GPU から CPU へのメモリ転送)にかかる計算コストの内訳を示している．GPU を利用した FFT では特に GPU から CPU へのメモリ転送にかかるコストが大きいため，FFT を実施する毎にメモリ転送することを避ける必要がある．本実装では，GPU 側に必要なメモリを事前に確保して計算を行い，全ての計算の終了後に必要なデータのみ CPU に転送することでメモリ転送にかかるコストの低減を図っている．

4. 評価と考察

本稿では，実装された TANDEM-STRAIGHT の実行速度を，従来のものと比較した結果について述べる．フレーム数 *N* に対応する分析シフト量は 5 ms とした．サンプリング周波数は 22.05 kHz (FFT 長 2048)と 44.1 kHz (FFT 長 4096)の 2 種類，入力信号は継続長が 0.5 s から 4 s まで 0.5 s 刻みのホワイトノイズとした．F0 分析では，F0 探索範囲を 32 Hz から 650 Hz とし 1 オクターブに 3 つの Extractor を配置したため，Extractor の総数 *M* は 15 となる．本信号に F0 は存在しないが，分析にかかる計算時間は F0 の有無に影響されない．1 つの条件につき 100 回試行し，その平均値を各条件の結果とした．実験に用いた CPU と GPU は 3 章の実験と同様である．

結果を図 4 に示す．なお，速度改善の比については，サンプリング周波数によらず同一の傾向を示したため，本稿では 44.1 kHz の結果のみを示すこととする．F0 分析では，信号長に僅かに依存するが概ね 7 倍の高速化が実現されている．スペクトル包絡推定では信号長が 4 s の場合は 4 倍程度高速化されている一方，信号長が 0.5 s の場合にはほぼ等速であることも確認された．

この結果は，スペクトル包絡推定に関して FFT 以外にも GPU による高速化が困難な演算が含まれていることを示している．また，信号長が短い場合は CPU，GPU 間のメモリ転送の影響が相対的に大きくなるため，改善率は信号長に概ね比例する結果になったと考えられる．

5. おわりに

本稿では，GPU と並列処理の組み合わせによる TANDEM-STRAIGHT の高速化について述べた．Matlab 版

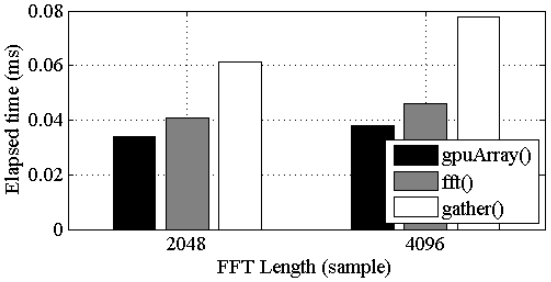

図 3：FFT にかかる演算コストの内訳．特に GPU から CPU に転送する gather()の計算コストが大きい．

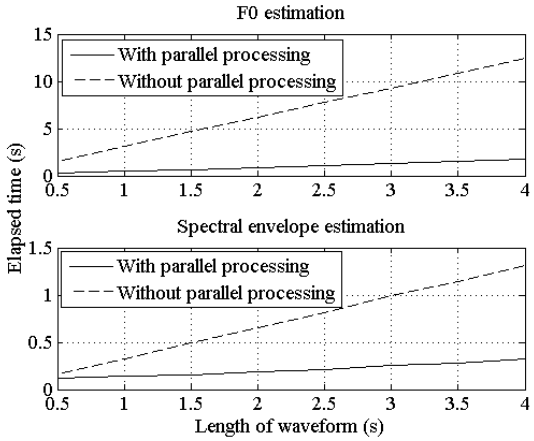

図 4：波形長と計算に要する時間との関連．

について高速化の効果を評価したところ，F0 推定法で約 7 倍，スペクトル包絡推定法で約 4 倍の高速化が実現された．

今後は，非周期性指標の推定に関しても並列化を行う．また，C 言語版の TANDEM-STRAIGHT である STRAIGHT Library [5]についても CUDA を用いた高速化を行う予定である．

謝辞

本研究の一部は，JSPS 科研費 23700221, 24300073, 2460085，および東北大学電気通信研究所 共同プロジェクト (H25/A08)の支援を受けて実施された．

参考文献

[1] H. Kawahara, et al. “TANDEM-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, f0, and aperiodicity estimation,” Proc. ICASSP 2008, pp. 3933-3936, 2008.
[2] H. Dudley, “Remaking speech,” J. Acoust. Soc. Am., vol. 11, no. 2, pp. 169-177, 1939.
[3] 右田尚人他, “歌唱データベースを用いたヴィブラートの個人性の制御に有効な特徴量の検討,” 情報処理学会論文誌，vol. 52, no. 5, pp. 1910-1922, 2011.
[4] http://www.mathworks.co.jp/products/parallel-computing/
[5] 坂野秀樹他，“リアルタイム STRAIGHT の改良と STRAIGHT ライブラリの実装,” 信学技報，SP2007-213, pp. 157-162, 2008.