Low-overlap window を用いたオンライン Wave-U-Net のアルゴリズム遅延の削減 *

☆中岡 想太郎 [1], 李 莉 [2], 牧野 昭二 [1,3], 山田 武志 [1]
[1] 筑波大学, [2]NTT コミュニケーション科学基礎研究所, [3] 早稲田大学

1 はじめに

音声認識や補聴器，車室内コミュニケーションをはじめ音声を用いるアプリケーションが普及している．入力音声に雑音が含まれる場合に音声の明瞭度や品質が著しく低下するため，これらのアプリケーションにおいては雑音を抑圧し，音声を強調する音声強調技術が不可欠である．

近年，深層学習（Deep Neural Network: DNN）の急速な発展により，モノラル音声強調の性能が飛躍的に向上した [1]．多くの手法では，短時間フーリエ変換（Short-Time Fourier Transform: STFT）で得られた振幅スペクトログラムを用いて DNN を学習し，時間周波数領域において雑音と音声を分離する [2, 3]．これらの手法は従来手法を凌駕する性能を示す一方で，推定信号の位相に混合信号の位相を利用するため，強調音声の明瞭度と品質に限界があるという問題点が残されている [4]．位相を考慮する DNN を学習する手法も多数提案されているが [5, 6]， STFT の冗長性に対する制約が課されておらず，推定された複素スペクトログラムに対応する時間領域の音声信号が存在しない可能性がある [7]．この問題を解決するため，時間領域で音源分離を行う手法が提案されている [8–10]．Wave-U-Net はそのような手法の一つであり，最初は歌声分離の手法として提案され [10]，後に音声強調に応用されている [11].

Wave-U-Net による音声強調は, 雑音を含む観測（混合）音声を入力とし，1 次元畳み込みニューラルネットワーク（Convolutional Neural Network: CNN）で構成される Down sampling (DS) と Up sampling (US) のブロックによって雑音成分の抑圧された音声信号を出力する手法である．推定された音声信号と混合信号の差分を推定の雑音信号とすることで，分離された各信号の和が入力信号と一致し，分離前後での整合性の保持が可能である．これらの性質を保ちながら Wave-U-Net は音声強調タスクにおいて高い性能が示されている．更に，長い窓を用いた STFT による高い周波数分解能が必要である周波数領域での高品質な音声強調に比べ，Wave-U-Net を含む時間領域での音声強調手法には窓長の制約が生じず，低遅延での処理が可能である．特に遅延の少ないオンライン処理が求められる会話型アプリケーションに適していると考えられる．

これらの性質から，我々はこれまでに，Wave-U-Net をオンライン手法に拡張し，teacher-student 学習を用いることにより短いセグメントを入力とするブロック処理でも高品質の音声強調が実現できることを確認できた [12]．そこで，短いセグメントを切り出すために，ハニング窓などオーバーラップ加算制約を満たしながら，周波数特性が解析に適した窓関数を用いることが一般的であるが，オンライン処理において，その窓長がアルゴリズム遅延の下界となる．そこで，本稿では，窓長に課せられたアルゴリズム遅延の下界を削減するために，low-overlap window を用いたオンライン Wave-U-Net を提案する．Low-overlap window はゼロ領域を持ち，窓長より短い信号を切り出すことができるため，アルゴリズム遅延の削減ができる．また，Wave-U-Net は時間領域の信号を直接にネットワークに入力するため，窓関数の周波数特性による影響はネットワークの学習により吸収され，音声強調性能に悪影響しないことが期待できる．提案手法の有効性は，車室内コミュニケーションを想定し，実測した走行雑音を用いた音声強調実験により示す．

2 Wave-U-Net による音声強調

2.1 オフライン Wave-U-Net

本節では，オフライン Wave-U-Net を用いた音声強調 [11] の問題設定とネットワーク構造について説明する．音声信号，雑音信号と観測信号をそれぞれ $s(t) \in [-1, 1]$, $n(t) \in [-1, 1]$, $m(t) \in [-1, 1]$ で表す．Wave-U-Net は観測された信号を分割せずに入力し，ニューラルネットワークを用いて雑音成分の抑圧された音声信号 $\hat{s}(t)$ を求める．

$$\hat{s}(1), \ldots, \hat{s}(T) = \mathcal{F}_\theta(m(1), \ldots, m(T)) \qquad (1)$$

ここで，$\mathcal{F}_\theta(\cdot)$ はパラメータ θ を持つニューラルネットワークによる混合信号から推定された音声信号への非線形写像を表し，$t = 1, \ldots, T$ は時間のインデックスである．混合信号と分離信号間の整合性を満たすために，推定雑音信号は以下のように計算される．

$$\hat{n}(t) = m(t) - \hat{s}(t) \qquad (2)$$

ネットワークパラメータ θ は以下のように，推定された音声信号と雑音信号の組と，混合される前の音声信号および雑音信号の組の二乗誤差を最小化する規準を用いて学習される．

$$\mathcal{L} = \mathbb{E}\left[|\hat{s}(t) - s(t)|^2 + |\hat{n}(t) - n(t)|^2\right] \qquad (3)$$

Wave-U-Net の基本的なネットワーク構造を Fig. 1 中の教師モデルに示す．Wave-U-Net は DS ブロックと US ブロックによって構成されており，DS ブロックと US ブロックの間にボトルネック層，US ブロッ

*Reducing algorithmic latency of online Wave-U-Net using low-overlap Window. Sotaro Nakaoka (University of Tsukuba), Li Li (NTT), Shoji Makino (University of Tsukuba, University of Wasesda), Takeshi Yamada (University of Tsukuba).

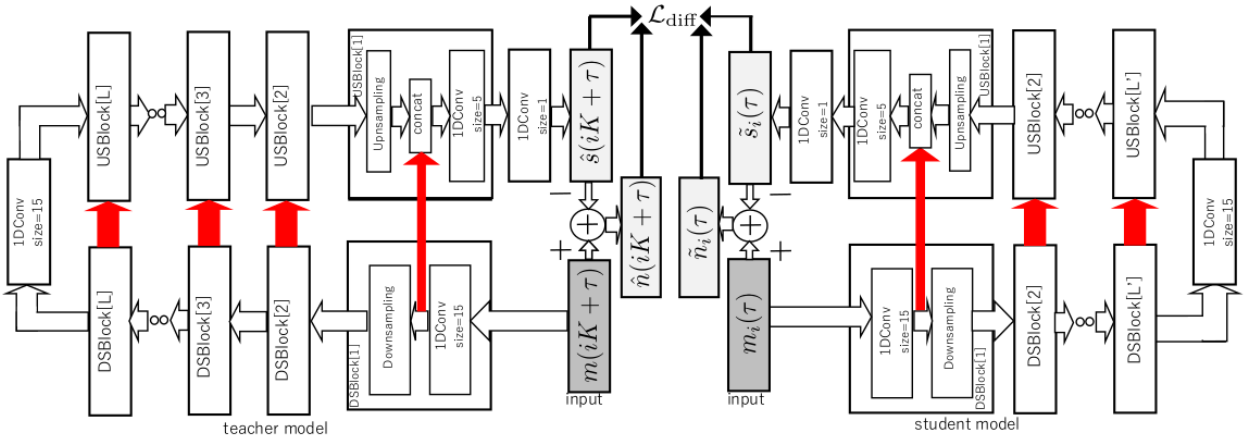

Fig. 1　Teacher-Student 学習に用いられるオフラインおよびオンライン Wave-U-Net のネットワーク構造と学習法.

クの直後に出力層である 1 次元 CNN 層が設けられている．DS ブロックは 1 次元 CNN 層とそれに続く DS 層によって構成されており，CNN 層における畳み込み処理によってチャネル数を増加させた後，DS 層でデータを 1 サンプルおきに排除する事で時間スケール上の分解能を半減させる．US ブロックは US 層とそれに続く 1 次元 CNN 層によって構成されており，US 層で各サンプルの間に中間値を補間してデータ長を倍にし，CNN 層において畳み込みによってチャネル数を減少させる．DS ブロックと US ブロックは対称であり，それぞれ L 個存在する．従って，出力層に入力される特徴量の時間分解能は入力データと同一である．

2.2　オンライン Wave-U-net

我々が以前上記のオフライン Wave-U-Net を逐次にブロックごとに音声強調を行うオンライン処理に拡張した [12]．すべての音声信号 $m(t), t = 1, \ldots, T$ の代わりに，固定長の入力セグメント $m_i(\tau) = m(iK + \tau)w_a(\tau)$ に対して逐次的に音声強調を行う．ここで，$\tau = 1, \ldots, K$ と i はそれぞれ時間とセグメントのインデックスであり，$w_(\tau)$ は長さが K の分析窓関数である．セグメント長 K はアルゴリズム遅延の下限であるため，低遅延アプリケーションを考慮する場合には K の短縮が求められる．しかしながら，K の短縮によって起こるセグメント内の利用可能な情報量の減少は音声強調性能の低下を招く，音声強調性能とセグメント長の間に生じるトレードオフを考慮する必要がある．

短いセグメントに対しても高い音声強調性能を得るために，オンライン Wave-U-Net では，teacher-Student 学習を用いる．Teacher-student 学習は，事前に学習された教師モデルが獲得した推論に関する知識を生徒モデルに伝達する学習方法であり，様々な場面に適用されている．例えば，ドメイン適応 [13] や，深いネットワークから浅いネットワーク [14]，異なるアーキテクチャのネットワーク [15] への知識の伝達に用いられる．教師モデル $\mathcal{F}_\theta(\cdot)$ のパラメータ θ は事前学習で最適化され，生徒モデルの学習時には固定される．生徒モデル $\mathcal{G}_\phi(\cdot)$ のパラメータ ϕ は以下の損失関数を最小化する基準の下で学習される．

$$\mathcal{L}_{\text{stu}} = \mathcal{L} + \beta\mathcal{L}_{\text{diff}} \tag{4}$$

ここで，$\mathcal{L}$ は正解データである雑音信号 $n_i(\tau) = n(iK + \tau)w_a(\tau)$ と音声信号 $s_i(\tau) = s(iK + \tau)w_a(\tau)$ の組と生徒モデルにより推定された雑音信号 $\tilde{n}_i(\tau)$ と音声信号 $\tilde{s}_i(\tau)$ の組の間の二乗誤差であり，$\mathcal{L}_{\text{diff}}$ は教師モデルと生徒モデルでそれぞれ推定された信号の組の間の二乗誤差である．$\beta(\geq 0)$ は後者の項に対する重み係数を表す．短い入力セグメント $m_i(\tau)$ を入力とした生徒モデル $\mathcal{G}_\phi(\cdot)$ の出力を $\tilde{s}_i(1), \ldots, \tilde{s}_i(K) = \mathcal{G}_\phi(m_i(1), \ldots, m_i(K))$ とし，推定雑音を $\tilde{n}_i(\tau) = m_i(\tau) - \tilde{s}_i(\tau)$ とする．式 (4) の第一項および第二項は以下のように定義できる．

$$\mathcal{L} = \mathbb{E}\Big[|\tilde{s}_i(\tau) - s(iK+\tau)|^2 + |\tilde{n}_i(\tau) - n(iK+\tau)|^2\Big] \tag{5}$$

$$\mathcal{L}_{\text{diff}} = \mathbb{E}\Big[|\tilde{s}_i(\tau) - \hat{s}(iK+\tau)|^2 + |\tilde{n}_i(\tau) - \hat{n}(iK+\tau)|^2\Big] \tag{6}$$

Fig. 1 にオフライン処理の教師モデルとオンライン処理の生徒モデルを用いた Wave-U-Net の Teacher-Student 学習を表す．教師モデルに対する生徒モデルの相違点は，DS ブロックと US ブロックの数 L' が異なる点と，入力セグメント長が限られている点である．

3　Low-ovarlap window による Online-Wave-U-Net

3.1　オンライン Wave-U-Net における窓処理

セグメントを滑らかに接続するため，入力音声は S サンプルずつシフトし，オーバーラップした時間間隔で分析窓関数 ω_a によって I 個のフレームに切り出され，Wave-U-Net による音声強調処理を経たフレームは合成窓関数 ω_s を乗じた上で加算される．この操作はオーバーラップ加算 (Overlap Add; OLA) などと呼

ばれる．周波数領域上で動作する音声信号処理においては，切り取られたフレームが，その外の時間においてもフレーム長を周期として同じ形状を繰り返す事を前提としたSFFTが行われるため，両端が0となる山型の関数を分析窓関数として選ぶ必要があり，なおかつ分析窓関数が与えるスペクトルへの影響を考慮する必要がある．分析窓で切り出した波形に何も加工を行わず合成窓での接続を行った際に元の波形に戻る事を保証するためには，以下の式を満たす必要がある．ここでτは時刻，iはフレームのインデックスである．

$$\sum_i w_{\mathrm{a}}(\tau - iS) w_{\mathrm{s}}(\tau - iS) = 1 \tag{7}$$

また，フレーム単位でオーバーラップ加算による二乗誤差の影響を最小にする最適な合成窓関数は以下となる [16].

$$w_{\mathrm{s}}(\tau) = \frac{w_{\mathrm{a}}(\tau)}{\sum_{i=-(Q-1)}^{Q-1} w_{\mathrm{a}}^2(\tau - iS)} \tag{8}$$

ここで，Qは窓長Kをシフト長Sで割った商K/Sである．

オンラインWave-U-Netにおいて，分析窓はデータをモデルに入力する前に乗算されるため，モデルの学習も分析窓を乗じた学習データで行われる．また，学習データと検証データに対応する正解データも，分析窓を乗じたものを用いる．ただし，分析窓はハニング窓を用いる．

3.2 Low-overlap windowによる窓処理

本節では, リアルタイム音声圧縮において用いられている窓関数を用いた, オンラインWave-U-Netのさらなる低遅延処理への拡張を提案する．Fig. 2に，[17]において提案されたlow-ovarlap windowを示す．この窓関数は音声圧縮のために提案され, 後に接続される離散コサイン変換 (MDCT) のための分析窓として用いられている．この窓関数はゼロ領域, オーバーラップ領域, 定数領域の3つの部分から構成されており, 窓長の半分をシフト長とする事を前提に設計されている．ゼロ領域の値は全て0であり，定数領域の値は全て1である．オーバーラップ領域にはVorbisコーデック [18] に使われている式 (9) の窓関数の値を用いる．

$$w(\tau) = \sin\left[\frac{\pi}{2}\sin^2\left(\frac{\pi\left(\tau + \frac{1}{2}\right)}{2L}\right)\right] \tag{9}$$

ただし，Lはオーバーラップ領域のサンプル長である．Low-overlap windowは，窓長の半分ずつシフトしてオーバーラップ加算される時，実際に隣同士のフレームの両方の値が計算に使われるのはオーバーラップ領域のみであるため，信号にこの窓関数を乗ずる時, ゼロ領域にあたる情報は消失するため, ゼロ領域の情報は元々準備する必要が無く, 結果として窓長からゼロ領域長を除いたサンプル長がアルゴリズム遅延の下限となる．例えば，窓長の25%をゼロ領域に設定すると，アルゴリズム遅延は25%低減する．Fig. 3にゼロ領域の長さを変化させた際のlow-overlap windowの変化を示す．また，次のフレームに影響される区間はオーバーラップ領域のみであるため，オーバーラップ領域を短くすることにより，リアルタイム処理を満たすための次のフレームの処理時間の上限を緩めることができる．

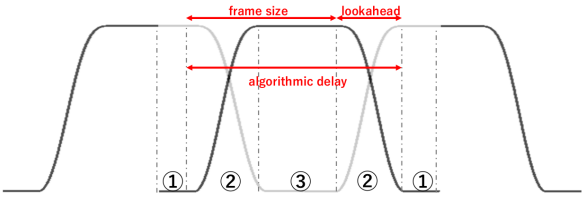

Fig. 2　Low-overlap windowの外観

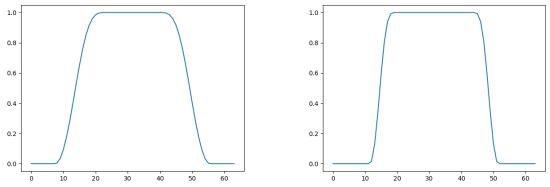

Fig. 3　ゼロ領域の長さを変化させた際のlow-overlap windowの形状の変化．ゼロ領域の割合が窓長の25%の場合（左）とゼロ領域の割合が窓長の40%の場合（右）．

4 評価実験

本節では，車室内コミュニケーションアプリケーションを想定し，走行雑音を用いて車内環境を模して，分析窓に従来のハニング窓を用いたオンラインWave-U-Netと，提案手法であるlow-overlap Windowを用いたオンラインWave-U-Netの音声強調性能とアルゴリズム遅延を評価する．音声強調性能の評価指標として，signal-to-distortions ratio (SDR)，signal-to-interferences ratio (SIR)，およびsignals-to-artifacts ratio(SAR) [19] を用いる．また，強調された音声信号の品質を評価する指標として，perceptual evaluation of speech quality (PESQ) [20] とshort-time objective intelligibility (STOI) [21] を用いる．

4.1 データセット

教師モデルおよび生徒モデルのネットワークパラメータの学習には，ドライソースの音声信号と車室内で収録された雑音信号を正解データとし，それらの混合信号をネットワークの入力データとして用いた．ドライソースの音声信号はCMU Arctic database [22] より，10話者の音声発話を用いた．学習と検証には“aew”, “ahw”, “aup”, “axb”, “eey”, “fem”の6話者（男性4人, 女性2人）の音声発話を用い，評価には“awb”, “bdl”, “clb”, “slt”の4話者（男性2人, 女性2人）を用いた．各発話の長さは3-7秒である．車室内雑音はJEIDA-NOISEデータベース [23] より，2種類の車両で収録された走行中の車室内における雑音信号を用いた．各車両で1時間の走行雑音が収録されており，1台の走行雑音を学習データとし，別の1台の

Table. 1　窓関数の種類による平均 SDR [dB], SIR [dB], SAR [dB], PESQ, STOI の比較.

分析窓 ω_a	ゼロ領域の比率 [%]	アルゴリズム遅延 [ms]	SDR	SIR	SAR	PESQ	STOI
low-overlap window	10	57.6	**15.15**	**27.64**	**15.56**	**3.18**	**0.95**
	25	48.0	**14.73**	**26.91**	**15.17**	**3.14**	**0.95**
	40	38.4	**14.19**	**26.33**	**14.62**	**3.10**	**0.94**
ハニング窓		64.0	**15.34**	**27.08**	**15.80**	**3.19**	**0.95**

走行雑音を評価に用いた．ネットワークパラメータの学習と検証に用いた混合信号は，音声と雑音の振幅の比率を [0.2, 0.9] の間でランダムに選択し，無作為に選択された区間の雑音信号と発話音声を加算して生成した．音声品質の評価に用いた混合信号は signal-to-noise ration（SNR）を {-3,0,3} dB の条件で各 50 発話，合計 150 発話を生成した．入力混合信号の平均 SDR は 0.07 dB であった．各信号の標本化周波数は 16 kHz である．

4.2　実験設定

Wave-U-Net モデルには，8 つの DS ブロックおよび US ブロックで構成したモデル ($L = 8$) を用いた．1 次元 CNN 層におけるフィルタサイズは DS ブロック，US ブロック，ボトルネック層と出力層の順に 15, 5, 15, 1 とした．l 番目の DS ブロックや US ブロックのチャネル数は $20l$ であり，ボトルネック層と出力層のチャネル数は $20(L + 1)$ および d1 とした．学習には学習率 0.0001 の Adam を用い，バッチサイズは 32 とした．学習済みのオフラインモデルを教師モデルとして，入力セグメント長 $K = 1,024$ の生徒モデルの学習を実施した．教師モデルであるオフラインモデルの入力セグメント長は $64,000$ とした．Teacher-Student 学習における重み β は 1 に設定した．分析窓に窓長全体に対するゼロ領域の比を $10, 25, 40$ とした 3 つの low-overlap window と，従来手法であるハニング窓を使用した，合成窓には式 (8) に示した各分析窓に対応する最適合成窓を使用した．窓関数は $1,024$ サンプルで，入力セグメント長と同じとしている．窓関数のシフト長はいずれも 512 サンプルで，窓長の半分としている．

4.3　実験結果

Table 1 に，それぞれの窓関数を用いた際のオンライン Wave-U-Net のアルゴリズム遅延, 平均 SDR, SIR, SAR, PESQ および STOI を示す．全ての窓関数の中で, 従来手法に用いられるハニング窓の場合は SDR の指標では最も良い音声品質となった．提案手法である low-overlap window を使用した中では，ゼロ領域の比率が高いもの程低い評価指標値となったが，いずれも 14 dB 以上の SDR が得られた．これは，ゼロ領域を拡大する事によるアルゴリズム遅延の縮小が，音声品質および分離品質とのトレードオフの関係にある事が示されている．実験中で最もアルゴリズム遅延の小さい，ゼロ領域の長さが窓長の 40% となる low-overlap window を用いた場合とハニング窓を用いた場合の結果を比較し，提案手法は SDR の低下を 1.15 dB に抑えながらオンライン Wave-U-Net のアルゴリズム遅延を 40% 減少させる事が分かった．

5　おわりに

本研究では，従来のオンライン Wave-U-Net で用いられていた窓処理を見直し，音声圧縮手法において用いられる low-overlap window の使用を提案した．走行中の車室内を模した評価実験により，提案手法は SDR の低下を 1.15 dB に抑えながらオンライン Wave-U-Net のアルゴリズム遅延を 40% 減少させる事が分かった．

謝辞　本研究は JSPS 科研費 19H04131 の助成を受けて行われた。

参考文献

[1] D. Wang, *et al.*, *IEEE/ACM Trans. ASLP*, 26(10), pp. 1702–1726, 2018.
[2] Y. Xu, *et al.*, *IEEE/ACM Trans. ASLP*, 23(1), pp. 7–19, 2014.
[3] K. Tan, *et al.*, *Interspeech*, pp. 3229–3233, 2018.
[4] M. S. E. Langarani, *et al.*, *ISSPA*, pp. 1446–1447, 2012.
[5] H. Erdogan, *et al.*, *ICASSP*, pp. 708–712, 2015.
[6] D. S. Williamson, *et al.*, *IEEE/ACM Trans. ASLP*, 24(3), pp. 483-492, 2016.
[7] S. Wisdom, *et al.*, *ICASSP*, pp. 900–904, 2019.
[8] S. Pascual, *et al.*, *Interspeech*, pp. 3642–3646, 2017.
[9] Y. Luo, *et al.*, *ICASSP*, pp. 696–700, 2018.
[10] D. Stoller, *et al.*, *ISMIR*, 2018.
[11] C. Macartney, *et al.*, *arXiv preprint:1811.11307*, 2018.
[12] S.Nakaoka, *et al.*, *ICASSP*, pp. 661-665, 2021.
[13] Z. Meng, *et al.*, *ICASSP*, pp. 6445–6449, 2019.
[14] A. Polino, *et al.*, *ICLR*, 2018.
[15] K. J. Geras, *et al.*, *ICLR*, 2016.
[16] D.R.Griffin, *et al.*, *IEEE Trans. ASSP*,, 32(2) pp. 236–243, 1984
[17] J.Mark, *et al. IEEE Trans. ASLP*, 18(1), pp. 58–67, 2009.
[18] Vorbis I specification, accessed in July. 14, 2021, http://www.xiph.org/vorbis/doc/Vorbis_I_spec.html, 2004.
[19] E. Vincent, *et al.*, *IEEE Trans. ASLP*, 14(4), pp. 1462–1469, 2006.
[20] A. W. Rix, *et al.*, *ICASSP*, pp. 749–752, 2001.
[21] C. H. Taal, *et al.*, *ICASSP*, pp. 4214–4217, 2010.
[22] J. Kominek *et al.*, *SSW*, pp. 223–224, 2004
[23] JEIDA Noise Database, accessed in Oct. 21, 2020, http://research.nii.ac.jp/src/en/JEIDA-NOISE.html