T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

論題	拡散モデルを用いた音声強調の計算量削減
Title	
著者	西 悠希, 岩野 広司, 篠田 浩一
Authors	Yuki Nishi, Koichi Shinoda
出典	電子情報通信学会技術研究報告, vol. 123, no. 292, pp. 1-6
Citation	IEICE technical report, vol. 123, no. 292, pp. 1-6
発行日 / Pub. date	2023, 11
権利情報 / Copyright	本著作物の著作権は電子情報通信学会に帰属します。 Copyright(c) 2023 IEICE

Powered by T2R2 (Science Tokyo Research Repository)

一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
NLC2023-14,SP2023-34(2023-12)

拡散モデルを用いた音声強調の計算量削減

西　　悠希† 　岩野　公司†† 　篠田　浩一†

† 東京工業大学　〒152–8550 東京都目黒区大岡山 2–12–1
†† 東京都市大学　〒224–8551 神奈川県横浜市都筑区牛久保西 3–3–1
E-mail: †y-nishi@ks.c.titech.ac.jp, ††iwano@tcu.ac.jp, †††shinoda@c.titech.ac.jp

あらまし　近年拡散モデルと呼ばれる生成モデルが注目されている. GAN と比べ、拡散モデルは安定に学習できるが、生成段階の計算コストが大きいという問題点がある. この傾向は音声強調への拡散モデルの応用に関しても同様である. 本稿では, 音声強調のための拡散モデルにおいて, Encoder,Decoder を用いることによる潜在空間にて音声信号を圧縮し, 圧縮された信号から拡散モデルにより雑音を除去することで, 精度を保ちつつ計算コストの削減することが可能なことを示す. 雑音と音声を同時に用いる訓練で Encoder, Decoder を学習した結果, PESQ を低下させずに生成時間を 50% 以上減少させることに成功した.

キーワード　拡散モデル, 音声強調, 潜在空間, 計算量削減

Yuki NISHI†, Koji IWANO††, and Koichi SHINODA†

† Tokyo Insutitute of Technology　2–12–1 Ookayama, Meguro-ku, Tokyo 152–8550, Japan
†† Tokyo City University　3–3–1 Ushikubonishi, Tuzukiku, Yokohama-shi, Kanagawa 224–8551, Japan
E-mail: †y-nishi@ks.c.titech.ac.jp, ††iwano@tcu.ac.jp, †††shinoda@c.titech.ac.jp

1. ま え が き

雑音除去を目的とした音声強調技術は様々な場面で必要となる. 補聴器による会話の補助や, 音声認識での雑音除去, Web 上での音声を含むコンテンツのための処理などである. 近年はニューラルネットワークを用いた技術が広く使われるようになり, より高い精度での音声強調が実現されている [1]. ニューラルネットワークを用いた技術の中で, 近年は拡散モデルと呼ばれる手法が, データの生成に関し高精度かつ訓練が安定しているということで注目されている. 音声強調においても, この拡散モデルを用いる研究がなされている [2], [3]. しかしこの手法は, 生成段階で繰り返しデータをネットワークに通す必要があるために計算量が多いという欠点を持つ.

本研究では, 拡散モデルにおける生成段階の計算コストを, なるべく精度を落とさずに削減することを目指したものである. Encoder, Decoder を用いた潜在空間を介することで, 入力される音声データを圧縮しその中で拡散モデルの生成処理プロセスを実行することで, 計算コストの削減を精度を落とさずに実現した. また Noisy-Train と名付けた, Encoder, Decoder を訓練する際にノイズのあるデータと無いデータを同時に使用し学習する方法を採用した.

2. 従 来 研 究

2.1 音 声 強 調

今回扱う音声強調とは, 環境雑音 n と人の音声データ x が混合されたデータ $y = x + n$ が与えられたとき, x を推定するタスクを指す. またここでは, シングルチャンネルのみを扱い, マルチチャンネルは考えない. 近年では, 音声強調にニューラルネットワーク (以下 NN) を用いることで, NN を用いない方法, 例えば特定の周波数領域を抑制する方法 [4] や, 人間の声を隠れマルコフモデルでモデリングする方法 [5] などよりも高い精度を達成している.

NN を用いる音声強調においては, 単純に NN に直接 Clean な音声 x を推定させる方法 [1] や, 周波数領域空間において y のフーリエ係数 (またはスペクトログラム) にどの程度 x が由来するかを意味するマスクを生成する方法 [6], [7], VAE [8] を用いた音声強調 [9] などがある.　近年音声強調において, 高い精度を達成した手法は, 敵対的生成ネットワーク (Generative Adversarial Network, GAN [10]) によるものである. 音声強調に GAN を応用した手法には MetricGAN+ [11] がある. しかし GAN には, 音声強調に限らず NN の訓練が不安定となる問題がある. その問題点を解決しつつ GAN に匹敵する精度を達成している手法が次に述べる拡散モデルである.

This article is a technical report without peer review, and its polished and/or extended version may be published elsewhere.
Copyright ©2023 by IEICE

2.2 拡散モデル

2.2.1 拡散モデルの理論的基礎

生成モデルによく使われる手法に, 拡散モデルと呼ばれる手法がある. 拡散モデルとは, 対象とするデータがノイズにより拡散していく過程をモデル化したもので, その逆過程を辿ることで, ノイズが混入したデータから元データを生成する方法である. この逆過程にガウスノイズなどを初期のノイズとして入力することで, ノイズが混入していない何らかの意味のあるデータを出力させることができる. 本稿ではスコアによらない拡散モデルとスコアによる拡散モデルに分類して紹介する (スコアについては後述).

スコアによらない拡散モデル DDPM [12] は, ニューラルネットワークにどのようなガウスノイズが付加されているかをニューラルネットワークに推定させる手法である. まず, T ステップのノイズ付加を定義する. $0 \leqq t \leqq T-1$ から $t+1$ へのデータの遷移 (順方向と呼ぶこととする.) を, $x_{t+1} = \sqrt{1-\beta_t}x_t + \sqrt{\beta_t}z$ とおく. β_t は t ステップにおけるノイズ付加の大きさを表すハイパーパラメータ, $z_t \sim N(0,1)$ である. すると, x_0(つまり Clean なデータ) と t が与えられたときに x_0 を陽に計算できるようになり, また x_{t+1} から x_t への遷移も陽に計算できるようになる. すると, それにより NN の訓練のための, t ステップにおける z_t を推定させる損失関数と, x_T から T ステップの逆遷移を経て x_0 にする計算を導出でき, 後者が推論, つまり画像や音声を生成するプロセスとなる.

スコアによる拡散モデルは, [13] にて導入された. これは, 確率的微分方程式 (Stochastic differential equation, 以下 SDE) を元に定式化がなされる. まず, ノイズの拡散方向である順方向を以下のように定式化する.

$$\mathrm{d}x_t = f(x_t)\mathrm{d}t + g(t)\mathrm{w}, \quad 0 \leqq t \leqq T \tag{1}$$

x_t は t におけるデータの状態で, x_0 がノイズのかかっていないデータに相当する. f はドリフト係数と呼ばれ, ベクトルを返す関数である. g は拡散係数と呼ばれ, スカラー値を返す関数で, 加えるノイズの強さを表す. w はウィーナー過程である. f, g は, 実験設定により様々な定義がなされる. これの逆方向の SDE は, [14] [13] により次のように定式化できる.

$$\mathrm{d}x_t = \left[-f(x_t) + g(t)^2\nabla_{x_t}\log p_t(x_t)\right]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathrm{w}},\ 0 \leqq t \leqq T \tag{2}$$

$\nabla_{x_t}\log p_t(x_t)$ がスコアと呼ばれる. $p_t(x_t)$ は t における x_t の確率分布である。このスコアを推定することが, NN の目的となる. また, $\bar{\mathrm{w}}$ は t を T から 0 まで逆向きにたどった場合の標準ウィーナー過程である. 確率微分方程式 (式 2) を元に, コンピュータにノイズのかかったデータ x_T を逆方向に遷移させ x_0 を得る方法は, 複数提案されている. もっとも単純な方法はオイラー・丸山法 [15] である. これは, T を N 個のステップに分割し, 離散的処理に置き換える手法である. また高精度を達成している手法に, スコアを導入した SDE に導入可能な PC-Sampler [12] がある.

2.2.2 拡散モデルを用いた音声強調

拡散モデルを音声強調に応用する試みも存在する. ただし, 生成モデルに広く使われている拡散モデル [12] ではガウスノイズのみの除去を想定している. 音声強調では, 想定されるノイズは様々な種類, 例えば本研究で用いた VOICEBANK-DEMAND [16] では駅の雑踏や鳥の鳴き声などがあるため, 拡散モデルの生成アルゴリズムをそのまま適用することはできない。

スコアによらない拡散モデルを用いた音声強調の例は, CDiffuSE [2] がある. これは, 順方向にステップが大きくなるにつれ, ガウスノイズだけでなく非ガウスノイズ (環境雑音など) が混入するというように定式化したものである.

スコアによる拡散モデルを用いた音声強調の例は, 本実験で Baseline とした SGMSE+ [3] がある. これは, 式 (1) においての f, g を, 以下のように定義したものである.

$$f(x_t, y) = \gamma(y - x_t) \tag{3}$$

$$g(t) = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^t\sqrt{2\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)} \tag{4}$$

このように定義することで, 式 2 におけるスコアが以下のように導出できる.

$$\nabla_{x_t}\log p_t(x_t) = -\frac{x_t - \mu(x_0, y, t)}{\sigma(t)^2} \tag{5}$$

$$\mu(x_0, y, t) = e^{-\gamma t}x_0 + (1 - e^{-\gamma t}y) \tag{6}$$

$$\sigma(t)^2 = \frac{\sigma_{\min}^2\left((\sigma_{\max}/\sigma_{\min})^{2t} - e^{-2\gamma t}\right)\log(\sigma_{\max}/\sigma_{\min})}{\gamma + \log(\sigma_{\max}/\sigma_{\min})} \tag{7}$$

ここで, y とは環境雑音の混じった音声である. $\sigma_{\max}, \sigma_{\min}$ はハイパーパラメータで, SGMSE+ では $\sigma_{\max} = 0.5, \sigma_{\min} = 0.05$ と設定された. γ は「硬さ」と表現される, x_0 から y へ遷移する強さに関するハイパーパラメータで, $\gamma = 1.5$ と設定された. このように定義することで,

$$p_{0t}(x_t|x_0, y) = N_{\mathbb{C}}(x_t; \mu(x_0, y, t), \sigma(t)^2\mathrm{I})$$

となり, t が増加するにつれ, x_t の中心は y へ漸近するようになる. SGMSE+ [3] は CDiffuSE [2] に比べ高精度を達成した。

2.2.3 計算量削減

拡散モデルでは, 推論時はデータを複数回 NN に通すことで, 徐々にノイズ除去するという過程であることから, 計算コストが大きくなるという問題点がある. これの解決を目指した様々な研究がある. 計算量削減の先行研究として, 推論時のステップ数を減らすアルゴリズムなどが考察されている. スコアによらない拡散モデルについて, DDIM [17] がある. これは拡散過程の定義式を変えることにより, DDPM [12] を包括するより一般性の高い過程を導出したうえで, それが任意のステップ間隔で逆過程を構成できることを示し, 標準的な DDPM [12] で訓練された NN にそのまま適用可能な任意のステップ間隔を設定できる逆過程を導出したものである.

スコアによる拡散モデルに関する計算量削減の先行研究とし

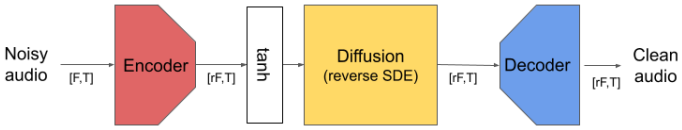

図 1　圧縮率 r における提案手法の推論プロセス. [ ] はデータの次元を表す.

て, [18] は, 様々な拡散モデルに応用できる Sampler を NN として学習する方法を提案した. また, DMCMC [19] は, 各ステップにおけるノイズの大きさを推定する比較的小さな NN を新たに導入した. この 2 つは逆過程で小さいステップ数を設定した場合でも精度低下を軽減したことを示し, 小さいステップ数での生成をより実用的なものとした.

また, 機械学習における知識蒸留 [20] を応用した研究がある [21]. これは, 大きなステップ数での設定で学習された教師モデルを元に, 少ないステップ数を生徒モデルに学習させることを繰り返すことにより, 小さなステップ数で生成するモデルを訓練する手法である.

2.2.4　潜在空間の利用

AutoEncoder [22], [23] や VAE [8], [24] における潜在空間とは, データを Encoder により写像したことにより形成される空間のことである. 潜在空間を用いた拡散モデルとして, 画像生成を目的とした Latent DIffusion Model [25] がある. これは, 画像生成が圧縮率の低い知覚的ステージ (Perceptual stage) と, 圧縮率が高い意味的ステージ (Semantic stage) に分割されるという仮説から, AutoEncoder や VAE を採用したものである. また, 音声データの処理に潜在空間と拡散モデルを用いた研究として AudioLDM [26], AudoLDM2 [27] がある. これらは Latent Diffusion Model [25] と同様の仮説のもと, 周波数領域に変換した音声データを, VAE を用いて潜在空間に写像して, 時間軸と周波数軸に関し圧縮するという手法を取った. ただしこれらはいずれも Channel の次元を増やしており, 計算量削減の方向性ではなく, 精度向上を主目的とした研究である.

3.　提 案 手 法

3.1　潜在空間の利用

潜在空間の次元を, 元の次元よりも小さくする Encoder, Decoder(以降この 2 つを表現する際は EncDec と略す) を用意する. その潜在空間の中で逆拡散を実行する. 逆拡散における, 扱うデータの次元が小さければ, その分計算コストは軽くなるのでそれを目的とした処理プロセス (図 1) となる.

本実験では EncDec は VAE ではなく通常の AutoEncoder であるが, そのまま用いると潜在空間においてのデータの分散が, Encoder による変換前のデータと大きく異なる可能性があるという問題がある. 本研究では, Encoder の出力に tanh 関数を挟み, またそれによるデータの標準偏差の変化も検証した.

3.2　Noisy-Train

EncDec を訓練する際には, EncDec にも雑音除去に関し何らかの役割を課すことができることを期待して, 単に音声信号を潜在空間に写像したのちに復元できるように NN を訓練したのではなく, 図 2 で示されるように, Noisy な音声を Clean な音声に変換させられるような訓練をした. この処理により, 単に Clean な音声のみで訓練した場合 (図 2 で, 常に $\alpha = 1.0$ とした場合) との精度比較は 5. で示す.

図 2　Noisy-Train のアルゴリズム

Algorithm 1 Noisy-Train

while not Converge **do**
  $\alpha \leftarrow \mathrm{Uniform}([0, 1])$
  $Clean, Noisy \leftarrow \mathrm{Dataset.Sample}()$
  $x \leftarrow \alpha * Clean + (1 - \alpha) * Noisy$
  $z \leftarrow \mathrm{Encoder}_\theta(x)$
  $\hat{x} \leftarrow \mathrm{Decoder}_\theta(z)$
  $loss \leftarrow \mathrm{Loss\ function}(Clean, \hat{x})$
  $\mathrm{Update}_\theta(loss)$
end while

4.　実　　験

4.1　モデル構造

提案手法は, 実装に必要な NN が EncDec, Diffusion の 3 つとなる. これらのアーキテクチャは全て [12] に提案されていた NCSN++ を用いる. NCSN++ の構造は 図 3 で表されるものであり, U-Net に似た構造となっている. 図 3 における, 例えば入力信号の次元は $[4, F, T]$ であるように, Residual Block を表す四角形の上または下にある数字は Channel の次元を示す.

図 3 における Residual Block は [3] にて使われているものを使用した. Uplayer, Downlayer は, 単純に次元を変化させる演算ではなく, Groupnorm [28] や有限インパルス応答フィルタ (FIR) [29] などの複数の処理を組み込んだモジュールで, BigGAN [30] にて提案された構造である. ProgUp, ProgDown は, FIR のみを用いて次元の圧縮や拡大を実行する演算である. また, Bottlenecklayer は Residual Block や Attention 層を組み合わせた構造である.

図 3 の入力次元の Channel の軸の 4 は, 拡散モデルに NCSN++ を適用した場合の数値である. Clean, Noisy なデータのそれぞれの実部と虚部を実数として NN に入力するので, その 4 パターンが対応する.　図 3 下段の各 ResudualBlock の入力の Channel 軸においては, 事前に「128 の何倍にするか」を設定してある. Downsample する段階での Residual Block の並び (つまり図 3 の下段左側) の Channel 次元は, $N = 6$ の場合は $(1, 1, 2, 2, 2, 2, 2)$ 倍である. Upsample する段階ではこの逆順となる. なお N に比べて ( ) の数字列が一つ多いのは, 最も左側の Residual Block の部分を繰り返し構造にカウントしていないからである.

4.1.1　次 元 圧 縮

圧縮率を $r = 1/X$(ただし X は自然数) と表記するとする. 計算量圧縮のための次元圧縮は, Encoder の最終出力の直前に, カーネルサイズが $(3, 1)$, ストライドが $(X, 1)$, パディングが $(X + 1, 0)$2 次元畳み込み演算を挿入する形で行う. なおこの出力が F/X

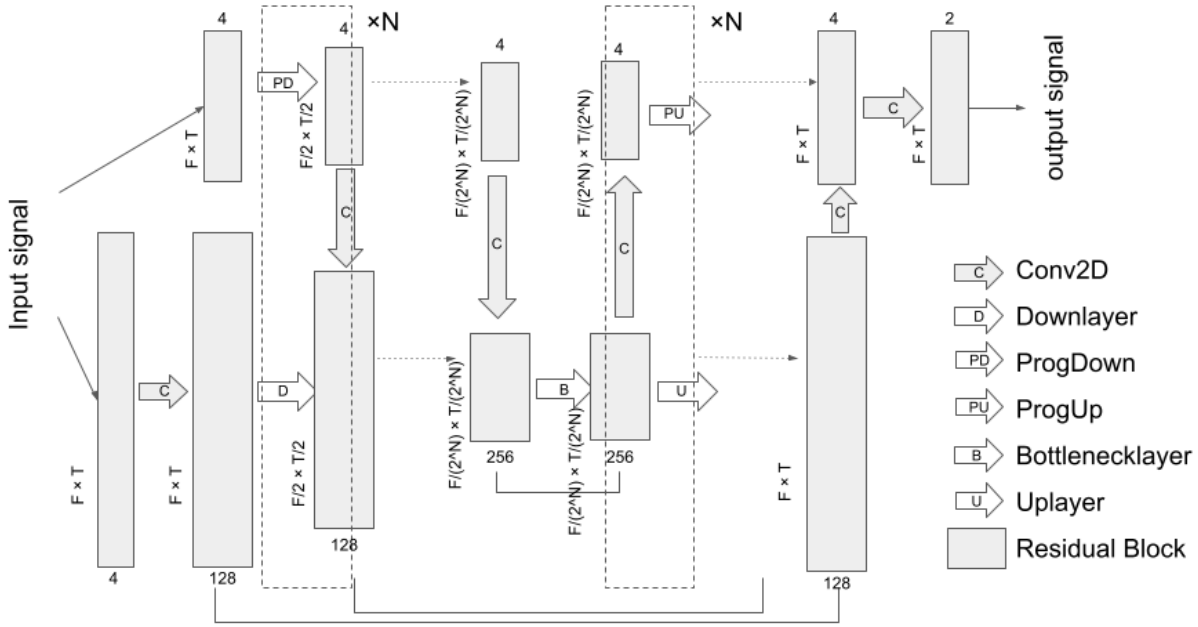

図 3　拡散モデル部分の NCSN++ のアーキテクチャ図. 書かれている数字は信号の次元を表すが, バッチサイズは省略されている. ^ は指数関数の表現で, / は小数点以下を切り捨てする除算である. EncDec では入力の Channel 次元が 2 となる。

より大きくなった場合は, 中心がそろうように次元の前部分と後ろ部分を削除する. それを拡散モデルで処理させたのち, Decoder に入力する直前で TransposeConv2D 演算 [31] を行うことで, 圧縮した次元を元に戻す. この Transpose2D は, カーネルサイズが $(\min(3, X), 1)$, ストライドが $(X, 1)$, output padding が $(\max(X - \min(3, X), 0), 0)$ である.

なお, 圧縮する次元は図 3 における F, つまり周波数の次元に関してのみ行う. T, つまり時間軸に関し圧縮しない理由は, 推論段階つまり実用する段階で入力される音は様々な長さが推定されるためであり, その長さが特定の値よりも短いと NCSN++ の段階的に圧縮する中で次元が 0 になる場合があるからである.

圧縮率の設定によっては, NCSN++ の処理プロセス内で次元に関する問題が発生する. 例えば $r = 1/8$ においては, 拡散モデルの NN のデータの次元は, Encoder による処理の前の次元は $[F, T]$ だとするとき, $[F/8, T]$ となる. ベースライン [3] では, $F = 256, N = 6$ であるから, 図 3 においての最も中心にあたる部分では, 圧縮が関わる後ろから 2 つ目の次元は $(256/8)/(2^6) < 1$ となってしまう. 対応策として, 圧縮すると次元が消失してしまう場合は図 3 においての Downlayer ではなく何もしない, と処置をした.

4.2　データセット

本実験では, 以下の 3 つのデータセットを用いた.

- VOICEBANK-DEMAND [16]: 音声コーパスである VOICEBANK [32] と, 10 種の雑音が収録された環境音データセット [33] を混合したデータセット. 以降 VB-DMD と略す. VOICEBANK [32] には 28 の話者 (男女それぞれ 14 名) があるセットと、56 の話者があるセット (男女それぞれ 28 名) があるが、そのうち 28 の話者があるセットを選択する. 各話者はそれぞれ約 400 の発話がある. 訓練用では合計 11572 の発話データがあり, 評価用では 824 個ある.

  DEMAND データセット [33] は, 18 種の雑音が収録されている. その中で, DKITCHEN, OMEETING, PCAFETER, PRESTO, PSTATION, TCAR, TMETRO, STRAFFIC, の 8 種を利用した. それに加え, [16] にて独自に作られた Babble noise(未使用の音声データ [32] から作られた小声での話し声), speech-shaped noise(ホワイトノイズを付加された男性の音声) の, 計 10 種を用意した. 訓練データでは, それぞれのノイズについて SN 比が 15dB, 10dB, 5dB, 0dB, 評価用データではそれぞれ 17. 5 dB, 12.5 dB, 7.5 dB, 2.5 dB の中からランダムに選択された音量で混合されたの一つの発話につき, 一つのノイズを混合させており. それぞれの発話は一回のみ使用された.

- WSJ0-CHiME3 [3]: ウォールストリートジャーナルの読み上げである WSJ0 [34] と, 音声認識コンペティションの 3rd CHiME Challenge にて用意された CHiME3 [35] を混合したデータセット. 以降 WSJ-C3 と略す . WSJ0 に, 訓練用には計 12776 個の発話がある si_tr_s データセット、評価用には計 699 個の発話がある si_et_05 のサブセットが用いられた。CHiME3 のノイズデータは, カフェ, ジャンクション, 公共バスの停留所, 歩行者用エリアの 4 種である. SN 比は 0dB から 20dB の一様分布である. それぞれの発話は一回のみ使用された.

なお, 訓練と評価はいずれも同じデータセットに対し行った.

4.3　精 度 指 標

強調された音声がどれほど Clean な音声に近いかを示す指標には, PESQ [36], ESTOI [37], SI-SDR, SI-SNR [38] を計測した.

- PESQ [36]: The Perceptual Evaluation of Speech Quality. 2001 年の ITU-T P. 862 にて勧告された.

- ESTOI [37]: The Extended Short-Time Objective Intelligibility. SOIT [39] に改良を加えたものであり, これは処理された

表 1　VB-DMD における, それぞれの圧縮率における強調処理の結果.

	PESQ↑	ESTOI↑	SI-SDR↑	SI-SIR↑	SI-SAR↑	Time[s]
SGMSE+	2.89	0.86	16.8	28.4	17.6	6.50
r=1/2, NT	3.12	0.87	17.4	29.7	18.2	4.33
r=1/4, NT	2.96	0.87	17.8	29.3	18.7	3.25
r=1/8, NT	2.62	0.80	12.0	32.6	12.1	2.81

表 2　WSJ-C3 における, それぞれの圧縮率における強調処理の結果.

	PESQ↑	ESTOI↑	SI-SDR↑	SI-SIR↑	SI-SAR↑	Time[s]
SGMSE+	2.96	0.91	17.2	32.0	17.4	14.1
r=1/2, NT	3.05	0.91	17.9	31.8	18.1	8.25
r=1/4, NT	3.03	0.91	17.2	32.0	17.4	6.56
r=1/8, NT	2.60	0.85	11.3	34.1	11.3	3.88

音声データの明瞭度や自然さといったものを数値化することを目指したものである.

- SI-SDR [38]: Scale-Invariant Signal to-Distortion Ratio. 比較する 2 つのデータの音量に関し不変となるように, 音の歪みを計算する指標である.
- SI-SIR, SI-SAR [38]: Scale-Invariant Signal-to-Interference Ratio と, Scale-Invariant Signal-to-Artifacts Ratio. [40] にて提唱された SIR と SAR を, 音量に関わらないように計算式を改変した指標である.

これらの精度指標は, いずれも値が大きいほど精度が良いことを示す.

4.4 実 験 設 定

実験は SGMSE+ [3] をベースに行った. SGMSE+ のコードには Web で公開されているものを利用した[1]. 実験は 2 つの段階に分けられる. 第一段階は, EncDec を訓練する. 第二段階は, 第一段階で訓練した EncDec を用いて, 拡散モデルの NN を訓練する. この際, EncDec のパラメータは固定する. 圧縮率は $r = 1/2, 1/4, 1/8$ を実行した. 拡散モデルの定義に関する各パラメータは 2.2.2 から変化はない. バッチサイズは, EncDec を訓練する際は 8 で, 拡散部分を訓練する場合は 16 ある. Optimizer は Adam [41] を使用し, 学習率は 10^{-4} とした. EncDec は同時に, VB-DMD は 124 epoch,WSJ-C3 は 112 epoch 学習し, Diffusion 部分は VB-DMD は 276 epoch,WSJ-C3 は 250 epoch 学習させた. ただし, Noisy-Train をしない場合の EncDec は, VB-DMD において $r = 1/4$ は 116 epoch, $r = 1/8$ は 123 epoch 学習した.

短時間スペクトル分析時の窓幅は 510 点であり, 周波数の次元数 (F) は 256 となる. また, シフト幅は 128 点とした. 音声データは, まずすべて 16kHz にリサンプリングされた. また訓練する際は, 周波数領域において, 時間の次元が $T = 256$ となるようにトリミングした.

5. 結　　果

表 1,2 は, それぞれ VB-DMD, WSJ-C3 に対する音声強調の性能評価結果である. 表中の Time は, 1 発話当たりの音声強調に要する平均所要時間 (EncDec の処理時間を含む) である. なお, 演算処理に利用した GPU は RTX 5000 である. NT とは, Noisy-Train を EncDec を訓練する際に実行したという意味である. これらの結果により、$r = 1/4$ までは性能を落とさずに生成時間の短縮が可能であることが示された. $r = 1/4$ の場合は, VB-DMD, WSJ-C3 において 50% 以上の生成時間短縮に成功している. 表 3 は, Noisy-Train の有無による性能変化である. 現状では特に性能変化は見られなかった. 表 4 は, Encoder にデータを通す前と後での, 標準偏差の平均である. この結果により, 標準偏差がそれほど変化なく, tanh 関数の妥当性が示されたといえる.

表 3　VB-DMD における, EncDec を Noisy-Train の実行の有無による性能変化

	PESQ↑	ESTOI↑	SI-SDR↑	SI-SIR↑	SI-SAR↑
r=1/4	2.91	0.87	17.2	29.8	17.9
r=1/4, NT	2.96	0.87	17.8	29.3	18.7
r=1/8	2.75	0.82	12.8	30.6	13.1
r=1/8, NT	2.62	0.80	12.0	32.6	12.1

表 4　VB-DMD の評価データにおける, Encoder にデータを処理させた際のデータの標準偏差の平均

変換前	r=1/2, NT	r=1/4, NT	r=1/8 , NT	r=1/4	r=1/8
0.096	0.128	0.189	0.231	0.071	0.231

6. 結　　論

本研究では, SDE による拡散モデルにおいて, 潜在空間を用いることで計算コストを削減しつつ精度を保てることを示した.

今後は, EncDec のアーキテクチャの改善, 潜在空間におけるデータの標準偏差と性能の関係の検証, さらに高い圧縮率の実現などが課題となる. また, 提案手法を 2.2.3 で紹介したステップ数の削減による計算コスト削減手法と組み合わせた場合の効果も検証対象である.

(注1)：https://github.com/sp-uhh/sgmse/tree/main

文　　献

[1] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, Vol. 26, No. 10, p. 1702–1726, oct 2018.

[2] Yen-Ju Lu, Zhong-Qiu Wang, Shinji Watanabe, Alexander Richard, Cheng Yu, and Yu Tsao. Conditional diffusion probabilistic model for speech enhancement. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7402–7406, 2022.

[3] Julius Richter, Simon Welker, Jean-Marie Lemercier, Bunlong Lay, and Timo Gerkmann. Speech enhancement and dereverberation with diffusion-based generative models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 31, pp. 2351–2364, 2023.

[4] S. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 27, No. 2, pp. 113–120, 1979.

[5] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, Vol. 77, No. 2, pp. 257–286, 1989.

[6] Donald S. Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Transac-*

tions on Audio, Speech, and Language Processing, Vol. 24, No. 3, pp. 483–492, 2016.

[7] Szu-Wei Fu, Ting-yao Hu, Yu Tsao, and Xugang lu. Complex spectrogram enhancement by convolutional neural network with multi-metrics learning. pp. 1–6, 09 2017.

[8] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[9] Simon Leglaive, Laurent Girin, and Radu Horaud. A variance modeling framework based on variational autoencoders for speech enhancement. 09 2018.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, p. 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

[11] Szu-Wei Fu, Cheng Yu, Tsun-An Hsieh, Peter Plantinga, Mirco Ravanelli, Xugang lu, and Yu Tsao. Metricgan+: An improved version of metricgan for speech enhancement. pp. 201–205, 08 2021.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[13] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[14] Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, Vol. 12, No. 3, pp. 313–326, 1982.

[15] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg, 2011.

[16] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech. In *Proc. 9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, pp. 146–152, 2016.

[17] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

[18] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022.

[19] Beomsu Kim and Jong Chul Ye. Denoising MCMC for accelerating diffusion-based generative models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, Vol. 202 of *Proceedings of Machine Learning Research*, pp. 16955–16977. PMLR, 23–29 Jul 2023.

[20] Geoffrey Hinton, Jeff Dean, and Oriol Vinyals. Distilling the knowledge in a neural network. pp. 1–9, 03 2014.

[21] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

[22] Umberto Michelucci. An introduction to autoencoders, 01 2022.

[23] Dor Bank, Noam Koenigstein, and Raja Giryes. Autoencoders, 03 2020.

[24] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, Vol. 12, No. 4, pp. 307–392, 2019.

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

[26] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. AudioLDM: Text-to-audio generation with latent diffusion models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, Vol. 202 of *Proceedings of Machine Learning Research*, pp. 21450–21474. PMLR, 23–29 Jul 2023.

[27] Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. AudioLDM 2: Learning holistic audio generation with self-supervised pretraining. *arXiv preprint arXiv:2308.05734*, 2023.

[28] Yuxin Wu and Kaiming He. Group normalization. *International Journal of Computer Vision*, Vol. 128, , 03 2020.

[29] Richard Zhang. Making convolutional networks shift-invariant again. In *ICML*, 2019.

[30] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 09 2018.

[31] Matthew D. Zeiler, Dilip Krishnan, Graham W. Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2528–2535, 2010.

[32] Christophe Veaux, Junichi Yamagishi, and Simon King. The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. In *2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, pp. 1–4, 2013.

[33] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. The Diverse Environments Multi-channel Acoustic Noise Database (DEMAND): A database of multichannel environmental noise recordings. *Proceedings of Meetings on Acoustics*, Vol. 19, No. 1, p. 035081, 06 2013.

[34] et al. Philadelphia: Linguistic Data Consortium Garofolo, John S. Csr-i (wsj0) complete ldc93s6a. web download, 09 1993.

[35] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe. The third 'chime' speech separation and recognition challenge: Dataset, task and baselines. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 504–511, 2015.

[36] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, Vol. 2, pp. 749–752 vol.2, 2001.

[37] Jesper Jensen and Cees H. Taal. An algorithm for predicting the intelligibility of speech masked by modulated noise maskers. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 24, No. 11, pp. 2009–2022, 2016.

[38] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey. Sdr – half-baked or well done? In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 626–630, 2019.

[39] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 19, No. 7, pp. 2125–2136, 2011.

[40] E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14, No. 4, pp. 1462–1469, 2006.

[41] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.