第5回定期ミーティング

2024年6月25日(火)

早稲田大学　基幹理工学研究科
電子物理システム学専攻 史研究室
石黒将太郎・野口颯汰

アウトライン

- 拡散モデル(DDPMからの発展)
  - DDIM
  - PNDM

- 参考文献

拡散モデルの生成速度

- DDPMの生成速度GANに比べて圧倒的に遅い
  - 32×32 サイズ、50000枚の画像生成にDDPMは20時間かかり、GANは1分未満で生成可能(RTX 2080Ti)
  - GANはネットワークを一度だけ通過するが、DDPMは100~1000回通過する必要アリ

- LDMは計算コスト低減にはなるが、生成速度の向上には寄与しない
  - デノイズ空間をピクセル空間から圧縮された潜在空間に移すことで、使用メモリ量を低減
  - 低次元での計算、メモリ効率向上でバッチサイズ増加により、生成速度は増加するが、ネットワーク通過回数は変わらない

- ネットワーク通過回数はデノイズ回数に依存するため、ノイズスケジュールの効率化が重要⇨DDIM・PNDM

DDIM

- DDPMは拡散プロセスがマルコフ的であるため、全てのプロセスにおいて、飛ばせる過程が存在しない⇨DDIMは**非マルコフ的**にしたい

  マルコフ過程：$q(x_{1:T}|x_0) = q(x_1|x_0)q(x_2|x_1) \ldots q(x_T|x_{T-1})$

  非マルコフ過程：

$$q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_0 + \sqrt{1-\alpha_{t-1}-\sigma_t^2}\cdot\frac{\boldsymbol{x}_t - \sqrt{\alpha_t}\boldsymbol{x}_0}{\sqrt{1-\alpha_t}}, \sigma_t^2\boldsymbol{I}\right).$$

$$\Rightarrow \quad q_\sigma(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0) = \frac{q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)q_\sigma(\boldsymbol{x}_t|\boldsymbol{x}_0)}{q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}$$

図. 拡散プロセスにおいてのマルコフ過程と非マルコフ過程の違い[1]

DDIM

- 非マルコフ拡散プロセスからの生成プロセス導出

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\epsilon, \quad \text{where} \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}).$$

$$\Rightarrow f_\theta^{(t)}(\boldsymbol{x}_t) := (\boldsymbol{x}_t - \sqrt{1-\alpha_t} \cdot \epsilon_\theta^{(t)}(\boldsymbol{x}_t))/\sqrt{\alpha_t}.$$

$$p_\theta^{(t)}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \begin{cases} \mathcal{N}(f_\theta^{(1)}(\boldsymbol{x}_1), \sigma_1^2 \boldsymbol{I}) & \text{if } t = 1 \\ q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, f_\theta^{(t)}(\boldsymbol{x}_t)) & \text{otherwise,} \end{cases}$$

- $q_\sigma\left(x_{t-1}\middle|x_t, f_\theta^{(t)}(x_t)\right)$を展開すると

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}{\sqrt{\alpha_t}}\right)}_{\text{“ predicted } \boldsymbol{x}_0\text{”}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \epsilon_\theta^{(t)}(\boldsymbol{x}_t)}_{\text{“direction pointing to } \boldsymbol{x}_t\text{”}} + \underbrace{\sigma_t \epsilon_t}_{\text{random noise}}$$

DDIM

- サンプルの更新式において、 σ =0の場合をDDIMと定義

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}{\sqrt{\alpha_t}} \right)}_{\text{“ predicted } \boldsymbol{x}_0\text{”}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \epsilon_\theta^{(t)}(\boldsymbol{x}_t)}_{\text{“direction pointing to } \boldsymbol{x}_t\text{”}} + \underbrace{\sigma_t \epsilon_t}_{\text{random noise}}$$

- DDPMとDDIMのノイズ予測モデルは、同じ目的関数でトレーニングされるため、DDIMに再利用可能
- DDIMの拡散プロセスを全ての潜在変数$\boldsymbol{x}_{1:T}$に対してではなく、長さSの部分集合$\{\boldsymbol{x}_{T1}, \ldots, \boldsymbol{x}_{Ts}\}$上で定義する

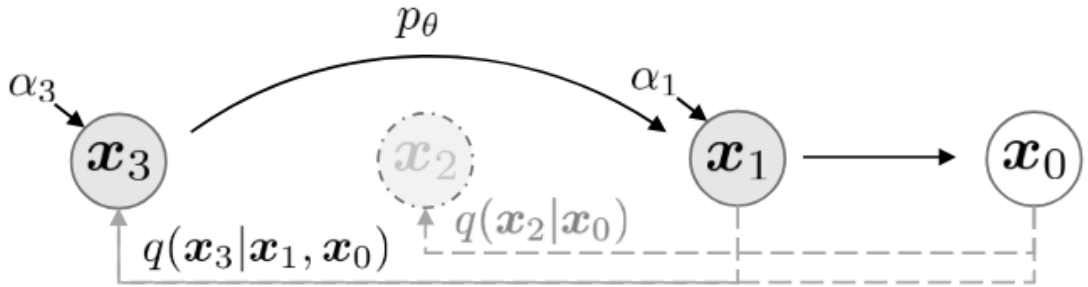


図. DDIMの生成プロセス[1]

DDIM

- 生成プロセスの確率性を操作する η を以下のように定義

$$\sigma_{\tau_i}(\eta) = \eta\sqrt{(1-\alpha_{\tau_{i-1}})/(1-\alpha_{\tau_i})}\sqrt{1-\alpha_{\tau_i}/\alpha_{\tau_{i-1}}},$$

表. η を操作した時の各モデルのFID[1]

		CIFAR10 (32×32)					CelebA (64×64)				
	S	10	20	50	100	1000	10	20	50	100	1000
η	0.0	**13.36**	**6.84**	**4.67**	**4.16**	4.04	**17.33**	**13.73**	**9.17**	**6.53**	3.51
	0.2	14.04	7.11	4.77	4.25	4.09	17.66	14.11	9.51	6.79	3.64
	0.5	16.66	8.35	5.25	4.46	4.29	19.86	16.06	11.01	8.09	4.28
	1.0	41.07	18.36	8.01	5.78	4.73	33.12	26.03	18.48	13.93	5.98
	$\hat{\sigma}$	367.43	133.37	32.72	9.99	**3.17**	299.71	183.83	71.71	45.20	**3.26**

PNDM

- 決定論的なDDIMによる生成プロセスは、離散形から微分形に変換することで、同義の常微分方程式(ODE)を導出可能

$$\frac{dx}{dt} = -\bar{\alpha}'(t)\left(\frac{x(t)}{2\bar{\alpha}(t)} - \frac{\epsilon_\theta(x(t), t)}{2\bar{\alpha}(t)\sqrt{1-\bar{\alpha}(t)}}\right).$$

- PNDMはこのODEを古典的数値手法を用いて解こうとする手法

  ⇨PNDMもDDIMの一種と言える
  - Forward Euler Methd
  - Ruge-Kutta Method
  - **Linear Multi-Step Method**

PNDM

- DDPMによる生成データの変化過程
  - Normは生成データのベクトルの大きさ
  - デノイズされるに従って、曲線を描きながら、目標分布に向けて一定の範囲内に収束する

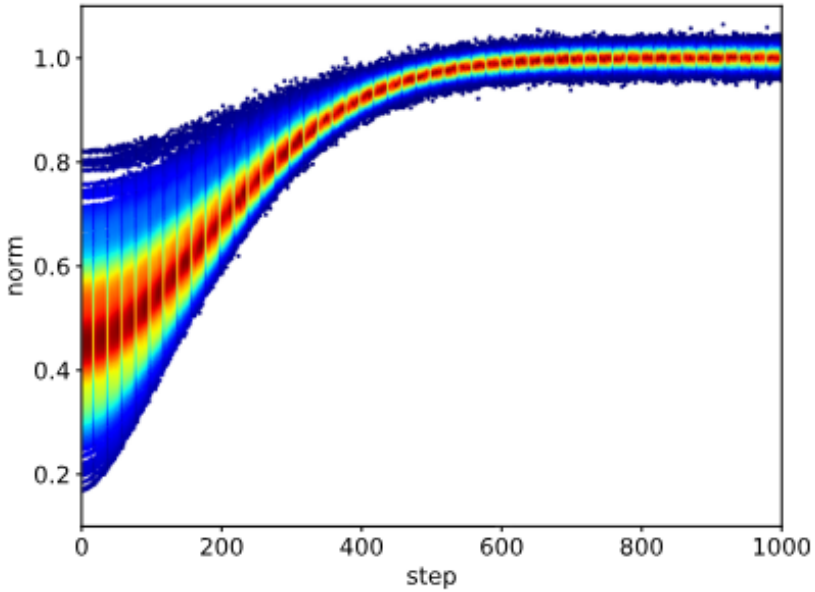


図. 生成データのノーム変化[2]

- 古典的数値手法の問題点
  - 直線的にデータを更新するため、ステップ数が少ないと、データの曲線的な変化に付いていけない可能性がある

- 擬似数値手法
  - 数値手法を勾配を計算する勾配部と、勾配を使ってデータを更新する転送部に分ける
  - 転送部を線形から非線形にすることで、曲線的な変化に対応

PNDM

- 擬似数値手法(Lenear Multi-Steps Methodメインの場合)

$$
\begin{cases}
e_t = \epsilon_\theta(x_t, t) \\
e'_t = \frac{1}{24}(55e_t - 59e_{t-\delta} + 37e_{t-2\delta} - 9e_{t-3\delta}) \\
x_{t+\delta} = \phi(x_t, e'_t, t, t+\delta).
\end{cases}
$$

- 生成アルゴリズム

Algorithm 1 DDIMs

$x_T \sim \mathcal{N}(0, I)$
for $t = T-1, \cdots, 1, 0$ **do**
    $x_t = \phi(x_{t+1}, \epsilon_\theta(x_{t+1}, t+1), t+1, t)$
end for
return x_0

Algorithm 2 PNDMs

$x_T \sim \mathcal{N}(0, I)$
for $t = T-1, T-2, T-3$ **do**
    $x_t, e_t = \text{PRK}(x_{t+1}, t+1, t)$
end for
for $t = T-4, \cdots, 1, 0$ **do**
    $x_t, e_t = \text{PLMS}(x_{t+1}, \{e_p\}_{p>t}, t+1, t)$
end for
return x_0

PNDM

- S-PNDMとF-PNDMは勾配部が異なり、Sは2ステップ前の情報を使い、Fは4ステップ前の情報を使っている
- timeはRTX-3090において、バッチサイズ512、ステップ数50で実験を行った際に、1ステップあたりにかかる平均計算を表す

表. 各モデルのFID・time[2]

dataset	FID / step / model	10	20	50	100	250	1000	time
Cifar10	DDIM	13.4	6.84	4.67	4.16		4.04	
	PF		13.8	3.89	3.69	3.71	3.72	
Cifar10 (linear)	DDIM*	18.5	10.9	6.99	5.52	4.52	4.00	0.337
	FON	13.1	7.41	5.26	4.65	4.12	3.71	0.390
	S-PNDM	11.6	7.56	5.18	4.34	3.91	3.80	0.344
	F-PNDM	7.03	5.00	3.95	3.72	**3.60**	3.70	0.391
Cifar10 (cosine)	DDIM	14.5	8.79	5.86	4.92	4.30	3.69	0.505
	S-PNDM	8.64	5.77	4.46	3.94	3.71	3.38	0.517
	F-PNDM	7.05	4.61	3.68	3.53	3.49	**3.26**	0.595
CelebA	DDIM	17.3	13.7	9.17	6.53		3.51	
CelebA (linear)	DDIM*	16.9	13.4	8.95	6.36	4.44	3.41	1.237
	FON	16.0	11.6	8.13	6.70	5.14	4.17	1.431
	S-PNDM	12.2	9.45	5.69	4.03	3.19	2.99	1.258
	F-PNDM	7.71	5.51	3.34	2.81	**2.71**	2.86	1.433

PNDM

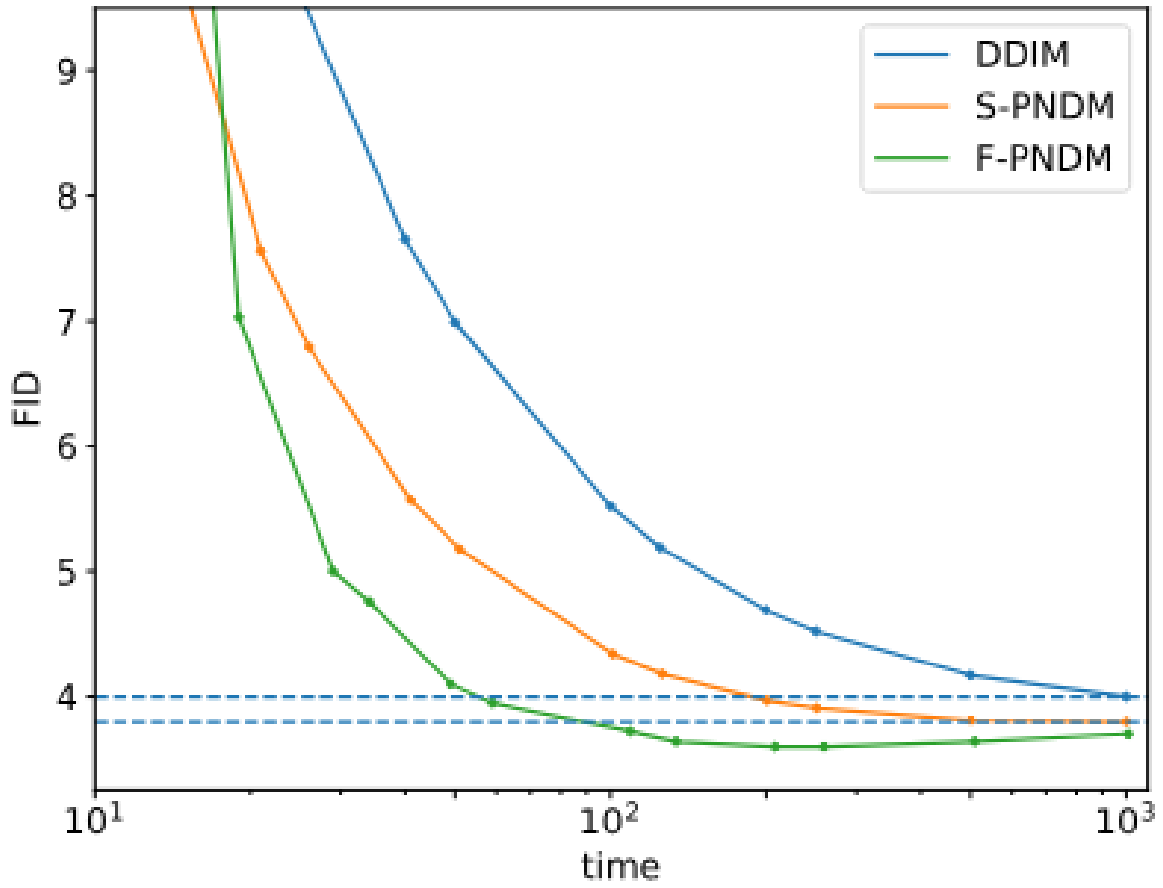
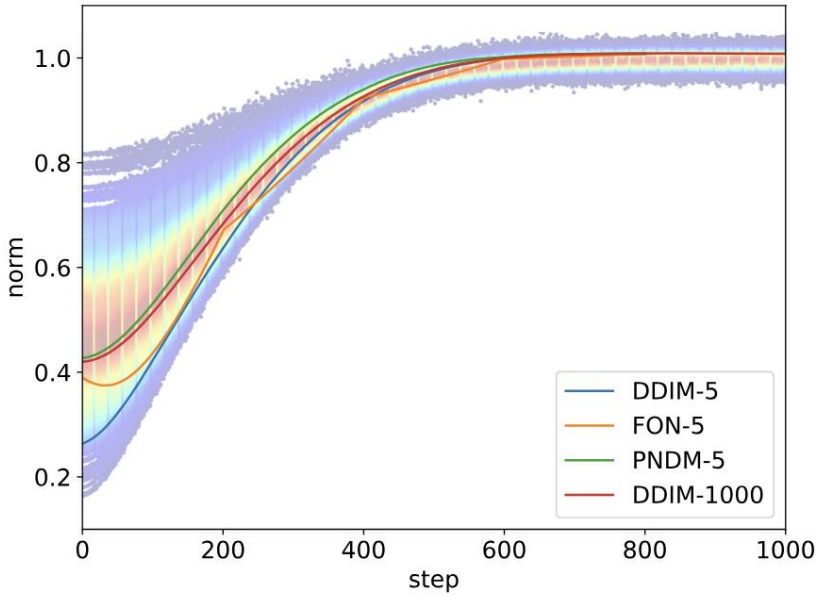
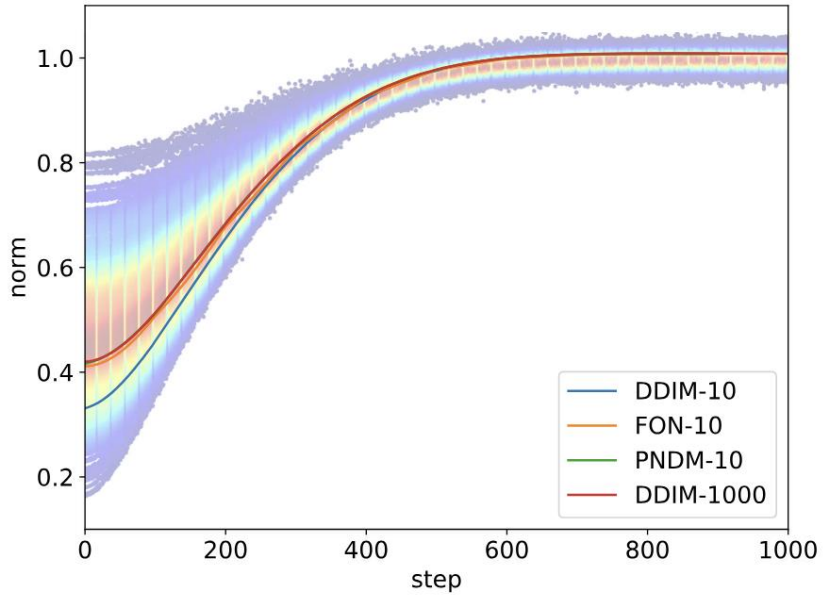


図. 各モデルのFID・time[2]

PNDM





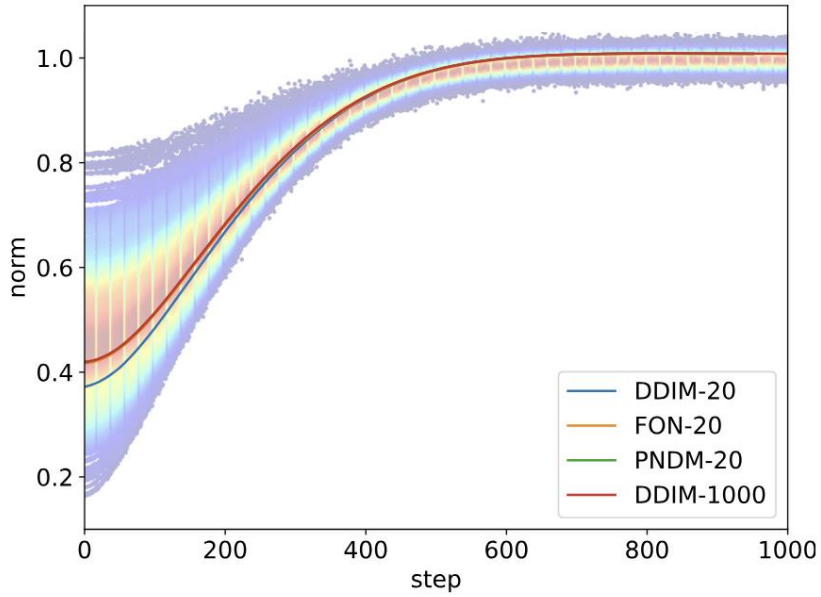


図. 各モデルの生成データの変化[2]

参考文献

1. Jiaming Song and Chenlin Meng and Stefano Ermon, “Denoising Diffusion Implicit Models”, ICLR 2021, 2022-10-05
2. Luping Liu and Yi Ren and Zhijie Lin and Zhou Zhao, ” Pseudo Numerical Methods for Diffusion Models on Manifolds”, ICLR 2022, 2022-10-31

DiffRF: Rendering-Guided 3D Radiance Field Diffusion

Norman Muller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo,
Peter Kontschieder, Matthias Nießner

DiffRF：ノイズ除去拡散確率モデルに基づいた新しい3D放射フィールド合成手法

左側：
3D教師データと体積レンダリングを利用して
高品質な3Dアセットを生成

右側：
未完成のオブジェクトの形状と外観を復元する
→特定タスクの訓練なしで条件づけすることで可能

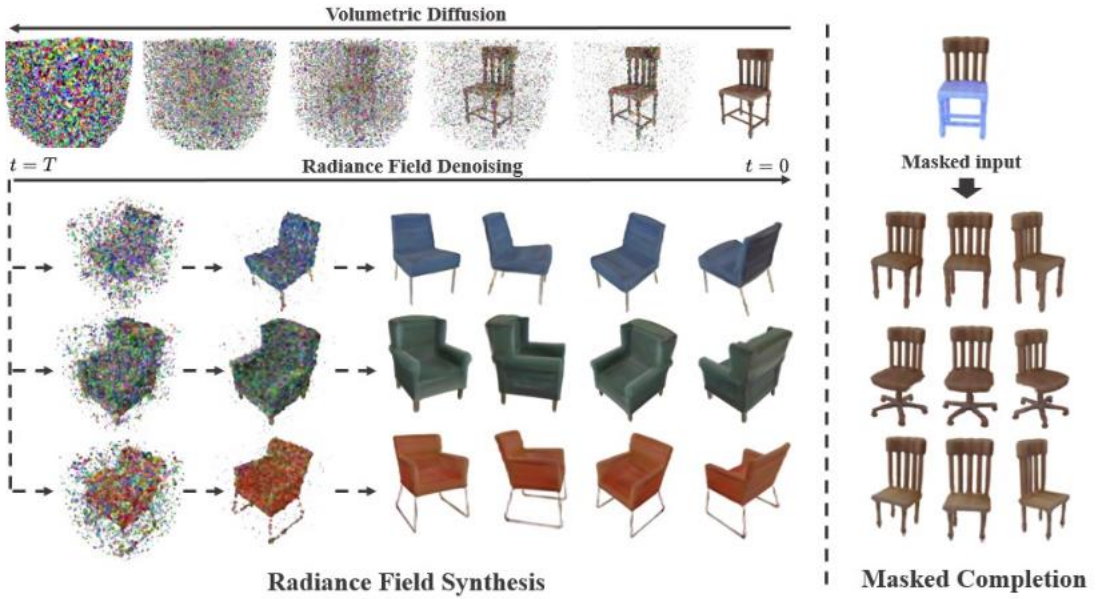


図1. DiffRF出力過程[1]

背景：

- NeRFsは、2D画像から3Dシーンを生成し、新しい視点からリアルな画像を作り出す技術で注目されているが、汎用性が高いneural fieldや、データセットを超えてシーンの事前知識を学習する方法は限られている。
- neural fieldの生成はあいまいさやアーティファクトの問題があり、正確なデータを取得するのは難しい
- 拡散モデルが2Dベンチメークにおける GANを超える性能を有しているが、直接3Dボリューメトリックneural fieldに適応するのは困難

→ノイズベクトルと対応する教師データサンプルの間に一対一のマッピングが必要となり、ほぼ不可能

取り組んだ問題：

- 3D放射フィールド上で直接動作する拡散モデル
- ノイズ除去プロセスをレンダリング損失と組み合わせて、良質な画像を生成するようにモデルを調整
- 3D放射フィールドのマスク付き補完
- 無条件および条件付き設定での結果

Radiance Fields

動画：NeRF生成例[2]

Radiance Fields とは：密度フィールドとカラーフィールドを使用して3Dオブジェクトを表現

有名モデル：NeRF

今回のモデルでは高品質なレンダリングとともに、より高速なトレーニングと推論を可能にするために
ボクセルグリッドでRadiance Fieldsを表現

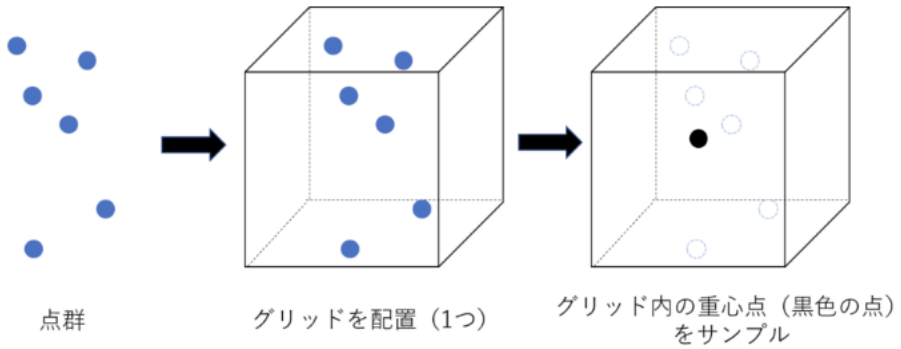


図2. ボクセルグリッドフィルタ[3]

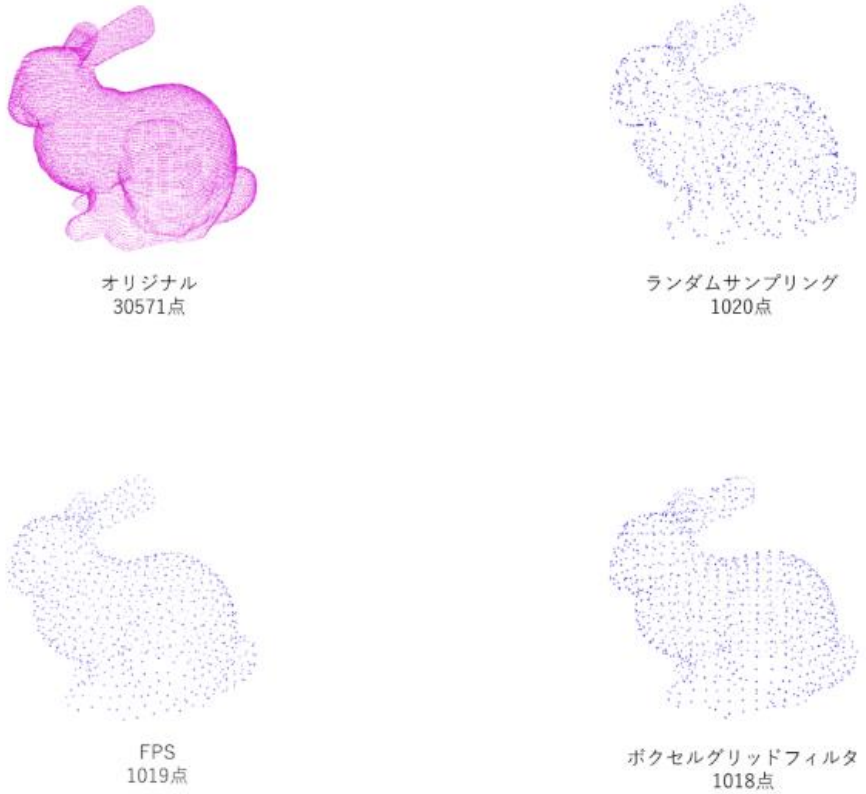


図3. 点群データセットダウンサンプリング処理[3]

概要

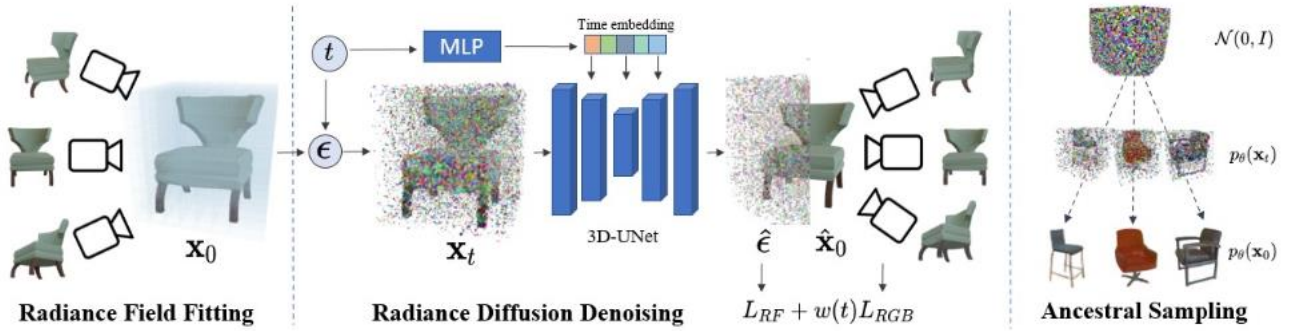


図4. DiffRFパイプライン[1]

① 初期放射フィールド f0 に対して固定ノイズスケジュールに基づいてノイズを加える
② 結果生じた ft を時間条件付き3D-UNetに通して、ノイズの推定値を得る
③ モデルはノイズ予測損失 LRF とレンダリング損失 LRGB によって調整
④ 最終的にノイズを除去した放射フィールド f0 を予測

損失関数

L_{RF}：データ分布に適合しない放射フィールドの生成をペナルティとして与える損失

$$
\begin{aligned}
L^t_{\mathrm{RF}}(f_0|\theta) &:= \mathbb{E}_q\left[\|\epsilon - \epsilon_\theta(f_t, t)\|^2\right] \\
&= \mathbb{E}_\phi\left[\left\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} f_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\right\|^2\right]
\end{aligned}
$$

L_{RGB}：生成された放射フィールドからのレンダリング品質を向上させることを目的としたRGB損失

$$
\begin{aligned}
L^t_{\mathrm{RGB}}(f_0|\theta) &:= \omega_t \mathbb{E}_{\phi,\psi}\left[\ell_v(\tilde{f}^t_0(\epsilon,\theta), I)\right] \\
\ell_v(f, I) &:= \|I_v - R(v, f)\|^2
\end{aligned}
$$

$$
\begin{aligned}
L(\theta) &:= L_{\mathrm{RF}}(f_0|\theta) + \lambda_{\mathrm{RGB}} L_{\mathrm{RGB}}(f_0|\theta) \\
&\propto \mathbb{E}_\kappa\left[L^t_{\mathrm{RF}}(f_0|\theta) + \lambda_{\mathrm{RGB}} L^t_{\mathrm{RGB}}(f_0|\theta)\right]
\end{aligned}
$$

実験概要

データセット

- PhotoShape Chairs ：
  15,576個の椅子をアルキメデス螺旋に沿った200の視点からレンダリング
- ABO Tables：
  異なる環境マップ設定を使用してレンダリングされたテーブルの画像

評価指標

- 画像品質
  FID：生成画像と実画像の統計的類似性
  KID：生成画像と実画像の特徴空間における距離
- ジオメトリ品質
  COV：サンプルのジオメトリの多様性を計測
  CDを用いたMMD：サンプルのジオメトリ品質を評価

無条件のRadiance Field (定量評価)

- ✓ GANベースと比較し、明示的なRadiance Fieldを処理するため、幾何学的な性質と多様性が向上
- ✓ 追加の2Dレンダリング損失により、画像品質が向上し、それなしでは品質が低下する

表1. PhotoShape Chairsデータセットにおける無条件生成の定量的比較[1]

Method	FID ↓	KID ↓	COV ↑	MMD ↓
π-GAN [7]	41.67	13.81	44.23	10.92
EG3D [8]	31.18	11.67	48.15	9.327
DiffRF w/o 2D	35.89	13.94	**63.46**	8.013
DiffRF	**27.06**	**10.03**	61.54	**7.610**

表2. ABO Tablesデータセットにおける無条件生成の定量的比較[1]

	Masking:	20%	40%	60%	80%	Avg
mPSNR↑	EG3D	23.71	24.86	24.92	25.79	24.82
	DiffRF	24.85	26.66	28.23	30.38	**27.53**
FID↓	EG3D	25.91	29.41	33.06	34.31	30.67
	DiffRF	22.36	27.74	31.16	29.84	**27.78**

無条件のRadiance Field (定性評価)

- ✓ EG3Dは良好な画像を生成するが、不正確な形状やアーティファクトが発生
- ✓ DiffRFは微細なフォトメトリック、ジオメトリックを持つRadiance Fieldを生成

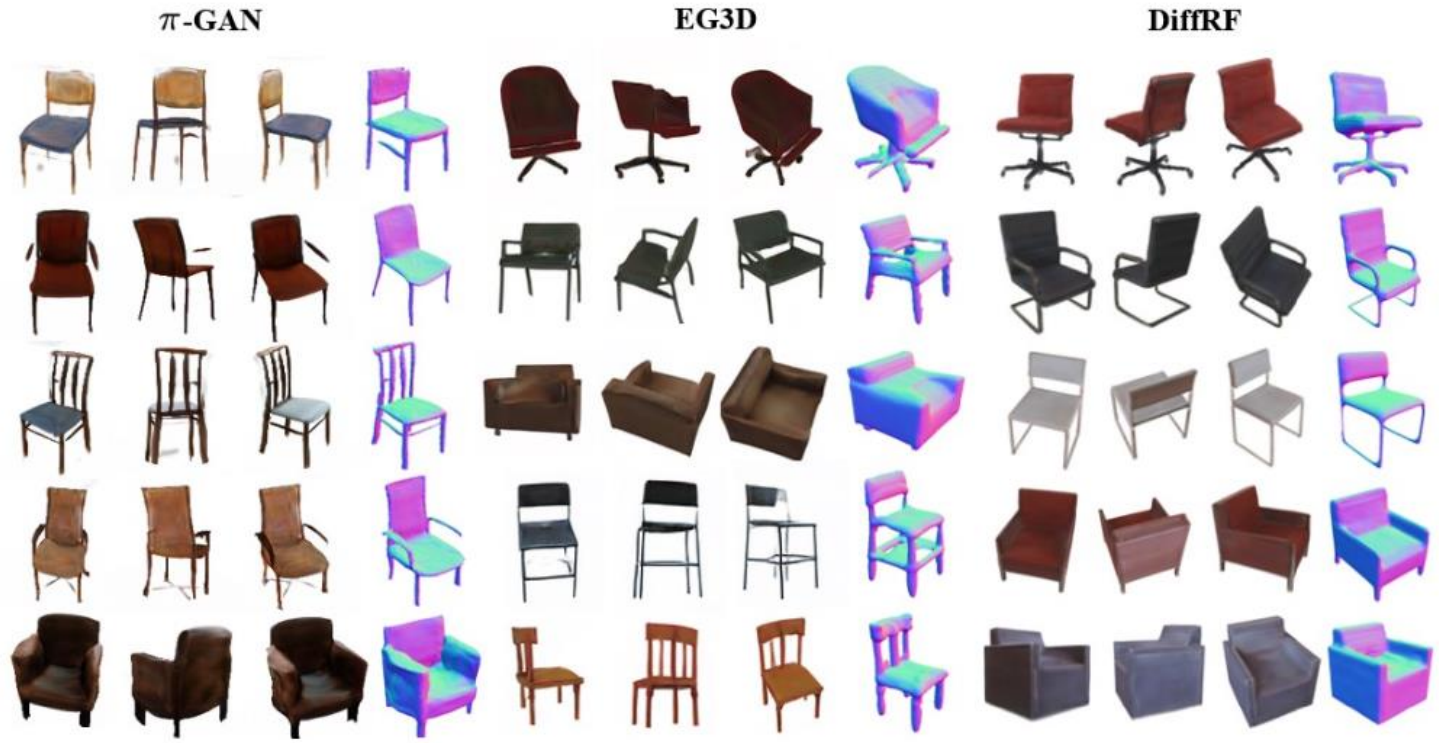


図5. PhotoShape Chairsにおける定性比較[1]

条件付き生成

- ◆ 拡散モデルの追加学習をせずに条件付けできる特性を用い、マスクされたRadiance Field補完を評価
- ◆ RePaintというモデルを参考に既知の領域内で無条件のサンプリングプロセスを徐々にRadiance Fieldに誘導することで補完する

$$f_0^{t-1} = \sqrt{\bar{\alpha}_t}(m \odot \tilde{f}_0^t + (1-m) \odot f^{in})$$
$$f_{t-1} \sim \mathcal{N}(f_0^{t-1}, (1-\bar{\alpha}_t)I)\,,$$

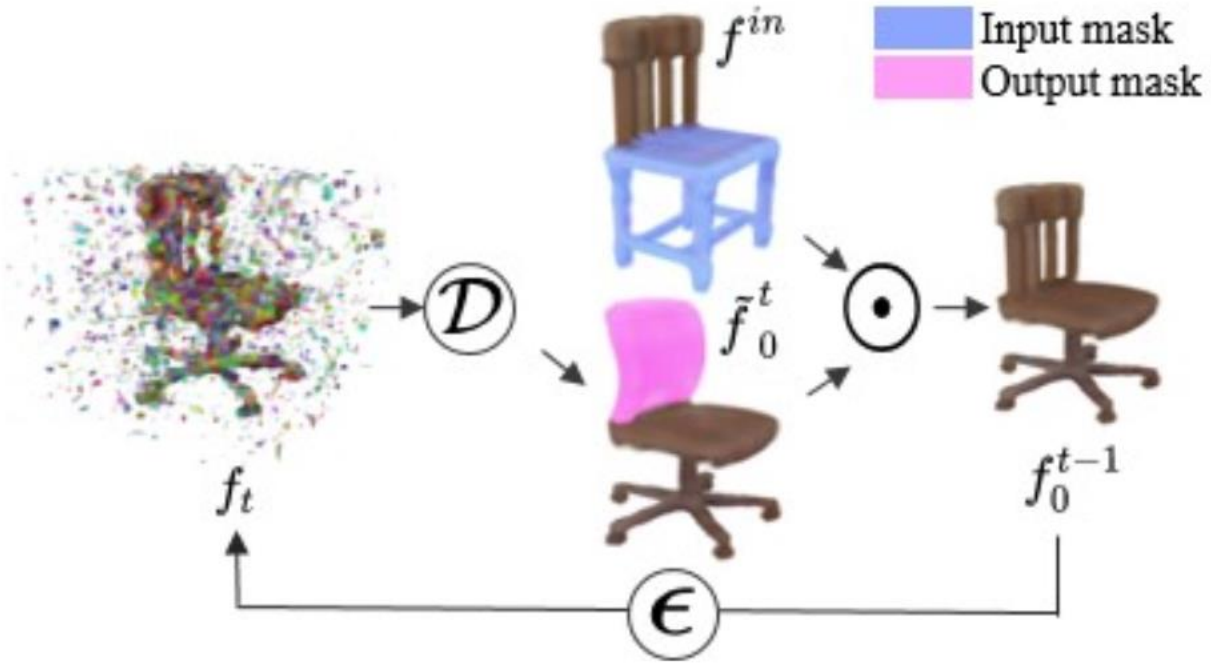


図6.マスク付きRadiance Field補完[1]

条件付き生成

実験概要：様々なマスクレベルの200サンプルに対してテスト
FIDと非マスク領域のPSNRを評価
定量評価：EG3Dは非マスク領域の構造を保持し辛い。
定性評価：非マスク領域の構造がDiffRFでは保持できている

表3. 異なるマスクレベルでの放射フィールド補完タスク評価[1]

	Masking:	20%	40%	60%	80%	Avg
mPSNR↑	EG3D	23.71	24.86	24.92	25.79	24.82
	DiffRF	24.85	26.66	28.23	30.38	**27.53**
FID↓	EG3D	25.91	29.41	33.06	34.31	30.67
	DiffRF	22.36	27.74	31.16	29.84	**27.78**

図7. PhotoShapeからのマスクされた椅子の補完[1]

条件付き生成

Classifer Guidance定式化を採用し、ポーズ付きRGB画像と対応するオブジェクトマスクに対するレンダリングエラーを最小限に抑える方向にノイズ除去を行う

図8. CLIP条件付けの生成結果[1]

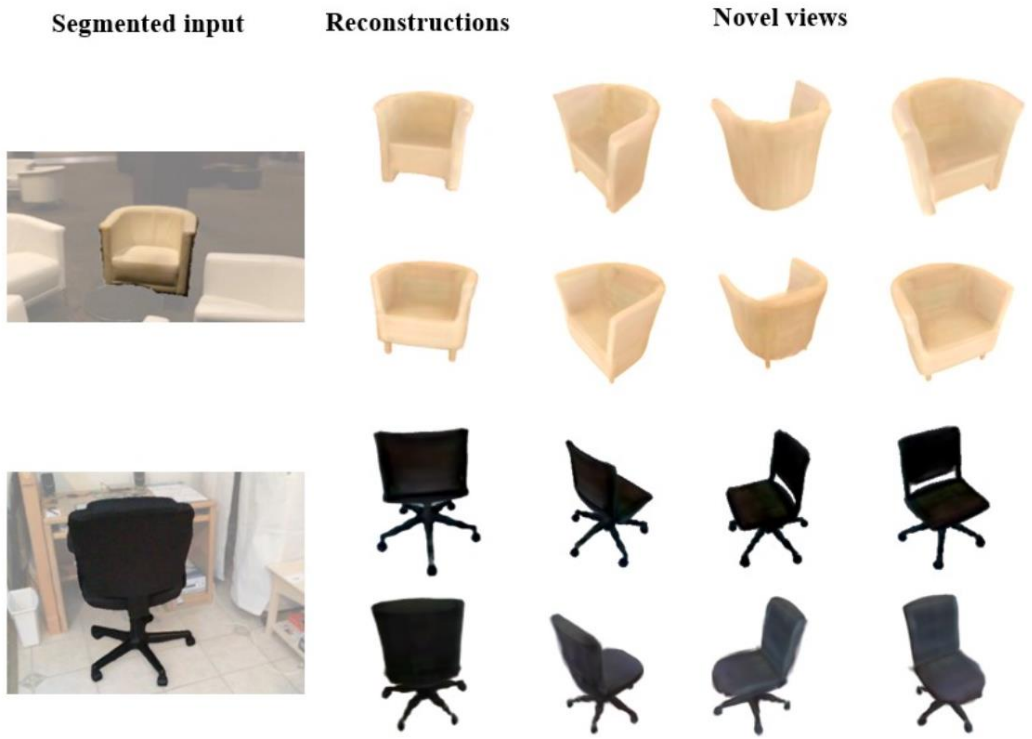


図9. 単一視点再構成の生成結果[1]

考察

3Dデータセットを拡散モデルで処理するには今の現状では考えられないくらい高度データセットが必要
→ARKiTを調査すべき

GANは条件づけで３D生成する場合、不自然になるので拡散モデルが優位

3D-Unetは2D-UNetアーキテクチャの2D畳み込み層とアテンション層を対応する3Dオペレーターに置き換えた
→具体的な方法、RGBD画像に対応したU-Netの調査

GANベースと比較して十分な数のビューボーズが必要
→より高速なサンプリング手法の活用

学習時のメモリ制約により、グリッド解像度に制約がある
→適応的または疎なグリッド構造の活用、factorized neural fields representationsの活用

今回、 Radiance Fieldにおいて拡散モデルを用いることで既存手法に比べて応用性・拡張性の優位が示された。

今後の研究計画

表6.今後の研究計画

	6月	7月	8月	9月	10月	11月	12月
データセット考察	→						
拡散モデル調査	→	→	→	→	→	→	→
実装と検証	→	→	→	→	→	→	→

- 3D-Unetの文献調査
- 3DMM条件付けの実現可能性
- Radiance FieldでStable Diffusionを実装できるのか調査