

事前学習用画像データセット生成モジュールコンテスト 4 位解法（定性評価）

Kot

Ryo Nakamuraらによる「OFDB : Pre-training Vision Transformers with Very Limited Synthesized Images」では、以下の点について述べています。

1. カテゴリごとに単一のインスタンスでVision Transformerは学習可能
2. ランダムなパッチテクスチャを付与するデータ拡張は有効
3. 三次元点群のYaw軸のみ回転させたものが最も精度がよい

そこで、3D-OFDBをベースに、点群データ拡張、微分可能レンダラーの導入による画像の表現力を向上させることで転移学習の精度向上を狙い、ソリューションを構築しました。

①パラメータ探索

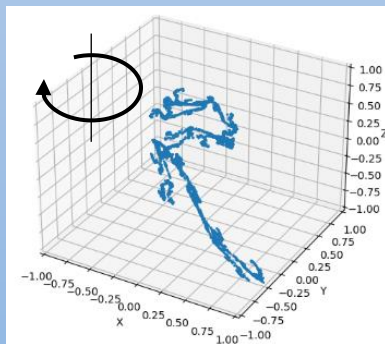
IFS Parameterのランダムサンプル+分散チェック

Iteration Function System

$$W_{n+1} = \begin{bmatrix} a_n & b_n & c_n \\ d_n & e_n & f_n \\ g_n & h_n & i_n \end{bmatrix} W_n + \begin{bmatrix} j_n \\ k_n \\ l_n \end{bmatrix}$$

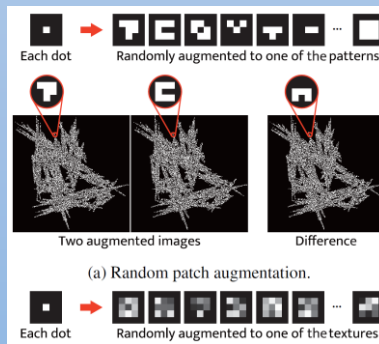
②点群生成

点群生成+回転 (Yaw軸)



③レンダリング

3*3パッチのテクスチャ拡張



④学習

画像データ拡張 (サーバー上で行われるため不明)

OFDBの画像生成のパイプライン

点群データ拡張、微分可能レンダラーの導入とデータセット生成のパイプラインとの対応を図に示す。実際には、IFSパラメータのサンプリング方法、点群データ拡張、レンダラーの変更、事前の画像拡張といった条件を組み合わせる実験を行いました。

①パラメータ探索

- IFS Parameterのランダムサンプル+分散チェック
- ランダムサンプル+行列のランクをチェック+分散チェック
- 特異値サンプリング

②点群生成

- 点群データ拡張
- シフト&スケール変更
 - 回転
 - ガウシアンノイズ
 - 色付き点群
 - 点群数の変更
 - ランダムサンプル
 - 点群変形 (PointWolf)
 - Mixup

③レンダリング

- 微分可能レンダラー (Synsin)
- 点群の透化によるテクスチャ変更
 - 点の大きさの変更
 - 背景付与 (DiamondSquare)

④学習

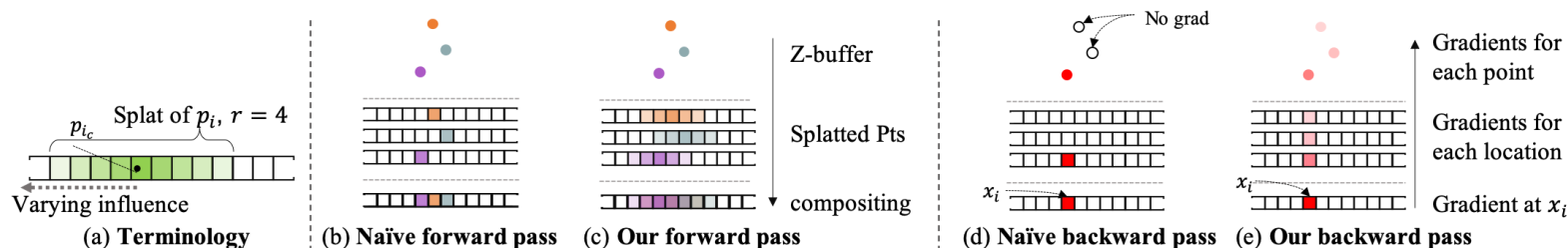
- 事前の画像拡張
- ShiftScaleRotate
 - RandomBrightnessContrast
 - HueSaturationValue
 - Cutout
 - Distortion
 - TrivialAugmentWide

試行した実験条件

先行研究ではOpenGLによるレンダリングを行っていましたが、本ソリューションではFacebook AI Researchによって開発された微分可能な点群レンダラーへの変更を行いました。

• 周囲の点を考慮した色の決定

従来の点群レンダラーは最も近い点の色を使用して画像を生成しますが、Pytorch3Dは深度情報を含む色の構成を可能にします。これはSynsin[3]で提案された方法で、勾配情報を持たせ微分可能にするためです。この特徴により、OFDBのランダムテクスチャ拡張を代替し、**立体構造を保持したまま多様なテクスチャをレンダリングが可能**にします。



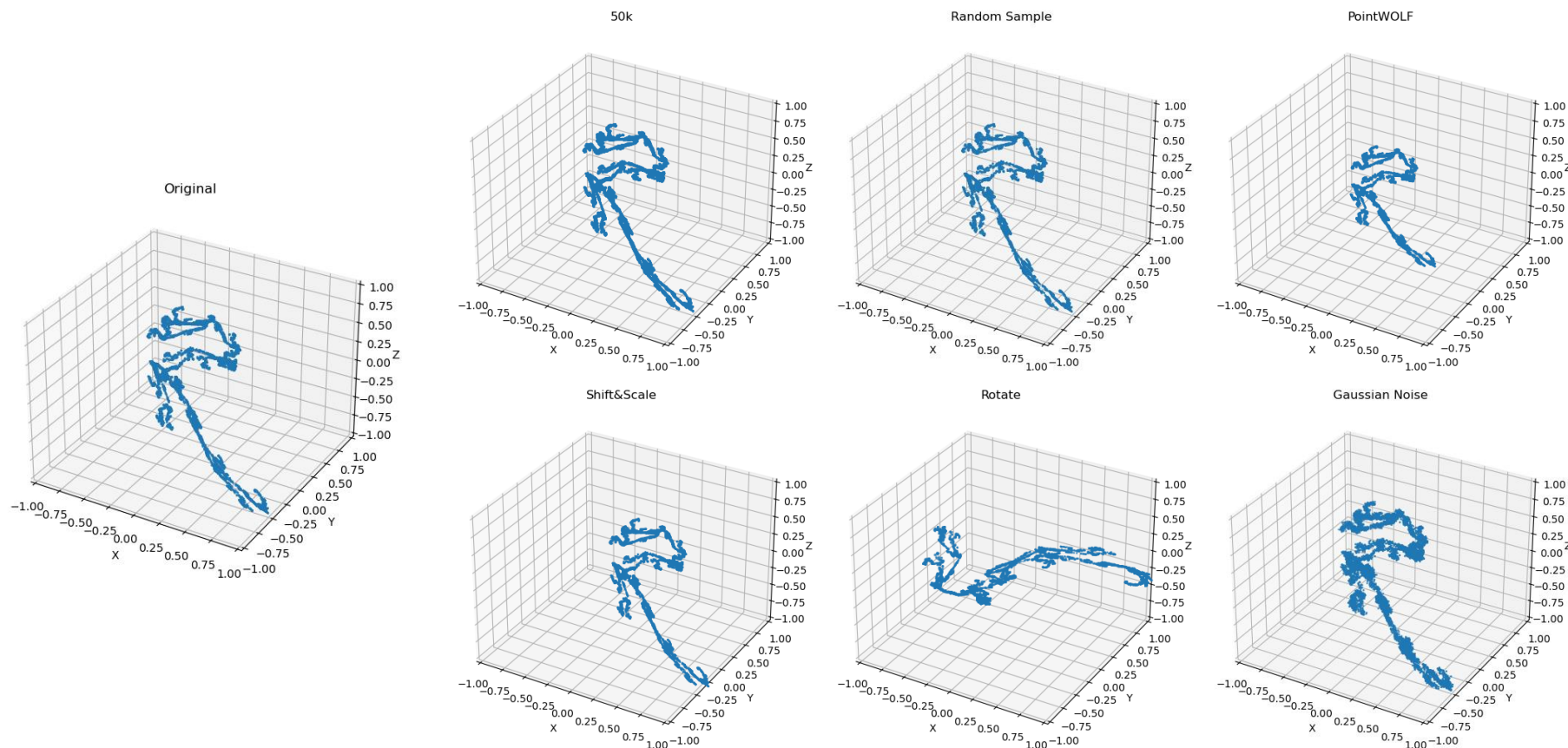
SynSinのレンダラー

• 球体ベース

先行研究において、FractalDB-1kで点数を変えながらのCIFAR100の分類精度実験では、10kの点数では学習が困難であったと報告されています[4]。Pytorch3Dでは点群を球体としてレンダリングすることが可能で、半径の調整により点数を10kに減らし、成功裏に学習を行うことができました。これは、フラクタルの輪郭が破損しなかったことが一因と考えられます。

点群用データ拡張の導入

点群のデータ拡張は、回転やスケーリング、平行移動、ノイズ付与といった単純なものからPointWOLFのような局所変形まで含めた手法の組み合わせを探索しました。



データ拡張の例

表は点群データ拡張を変更してCIFAR100の転移学習を測定した結果を示す。
ここでは、点群をランダムにサンプルするものが精度が高く、Pytorch3Dでレンダリングした画像は色が拡張されていることが精度向上に寄与していることがわかる。一方で点群分類に使用されるPointWOLF[5]は精度が改善に寄与しなかった。また、背景の追加や色情報の追加も改善に寄与していないことがわかります。

Experiment							
Shift&Scale	✓	✓	✓	✓	✓	✓	✓
Rotate	✓	✓	✓	✓	✓	✓	✓
Cololize Points		✓					
Add Points 10k->50k			✓				
Random Sample				✓	✓	✓	✓
Gaussian Noise					✓		
PointWolf[7]						✓	
Add Background Diamond[8]							✓
CIFAR100 Accuracy(%)	73.23	71.83	72.95	74.46	72.69	72.08	71.84

表は前頁の条件でレンダリングした画像に対して事前に画像拡張を行った時のLBの結果を示す。生成したデータセットによる事前学習は、サーバーで実施されるためDataLoaderでデータ拡張を変更できない。そのため手元で予め、拡張した画像を生成した。もし、適切なデータ拡張が選択されている場合は、ここでの精度改善は小さいはずだがいくつかのデータ拡張はLBでのスコア改善がみられた。ここでは、損失関数への操作はできないためMix系は含めていない。

Experiment						
Baseline	✓	✓	✓	✓	✓	✓
Flips		✓	✓	✓	✓	
ShiftScaleRotate		✓	✓	✓	✓	
RandomBrightnessContrast		✓	✓	✓	✓	
HueSaturationValue			✓			
Cutout				✓		
Distortion					✓	
TrivialAugmentWide						✓
Leader Board	1,186.22	1164.94	1157.03	1161.29	1171.68	1179.81

Solution

本ソリューションは、3D-OFDBを拡張し、事前学習の効率をさらに向上させる方法を提案します。主要な改善点は以下の2点です。

1. レンダラーをOpenGLからPytorch3Dに変更し、点を球体として扱うことで、必要な点数を削減し、色の多様性を表現しやすくしました。これにより、GPUを効率的に活用し、画像生成を高速化します。
2. 点群に対する様々なデータ拡張技術（例えば、スケーリング、回転、ランダムサンプル）を組み合わせ使用し、事前学習の効率をさらに改善しました。

これらの改善により、CIFAR100の転移学習での精度比較から3D-OFDBを用いた事前学習の効率は向上することを確認しました。

①パラメータ探索

- IFS Parameterのランダムサンプル+分散チェック
- ランダムサンプル+行列のランクをチェック+分散チェック
- 特異値サンプリング

②点群生成

- 点群データ拡張
- シフト&スケール変更
 - 回転
 - ガウシアンノイズ
 - 色付き点群
 - 点群数の変更
 - ランダムサンプル
 - 点群変形 (PointWolf)
 - Mixup

③レンダリング

- 微分可能レンダラー (Synsin)
- 点群の透化によるテクスチャ変更
 - 点の大きさの変更
 - 背景付与 (DiamondSquare)

④学習

- 事前の画像拡張
- ShiftScaleRotate
 - RandomBrightnessContrast
 - HueSaturationValue
 - Cutout
 - Distortion
 - TrivialAugmentWide

ソリューション（提出したものを青文字で示す）

- R. Nakamura, H. Kataoka, S. Takashima, E. J. M. Noriega, R. Yokota, and N. Inoue, 'Pre-training Vision Transformers with Very Limited Synthesized Images', arXiv [cs.CV]. 2023.
- N. Ravi et al., 'Accelerating 3D Deep Learning with PyTorch3D', arXiv [cs.CV]. 2020.
- O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, 'SynSin: End-to-end View Synthesis from a Single Image', CoRR, vol. abs/1912.08804, 2019.
- H. Kataoka et al., 'Replacing Labeled Real-Image Datasets With Auto-Generated Contours', in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 21232–21241.
- R. Yamada et al., 'MV-FractalDB: Formula-driven Supervised Learning for Multi-view Image Recognition', in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 2076–2083.
- S. Kim, S. Lee, D. Hwang, J. Lee, S. J. Hwang, and H. J. Kim, 'Point Cloud Augmentation with Weighted Local Transformations', CoRR, vol. abs/2110.05379, 2021.
- J. Ren, L. Pan, and Z. Liu, 'Benchmarking and Analyzing Point Cloud Classification under Corruptions', arXiv:2202.03377, 2022.
- C. Anderson and R. Farrell, 'Improving Fractal Pre-training', arXiv [cs.CV]. 2021.