Pre-training of Auto-generated Synthetic 3D Point Cloud Segmentation for Outdoor Scenes

Takayuki Shinohara
National Institute of Advanced Industrial Science and Technology
Tsukuba, Japan
shinohara.takayuki@aist.go.jp

Abstract

Acquiring and annotating large datasets for the segmentation of outdoor 3D point clouds observed by airborne light detection and ranging (LiDAR) is resource-intensive and fraught with privacy concerns, limiting the availability of labeled training data. Pre-trained models can help to some extent but their effectiveness hinges on large datasets, and self-supervised learning faces data scarcity challenges. We propose a formula-driven auto-generated terrain and shape point cloud dataset for 3D point cloud segmentation tasks. Our synthetic dataset was created from diverse 3D models with variations in polygon types and shape similarity, and provides a high-quality pre-training alternative to existing datasets. Experiments reveal that models pre-trained on our synthetic data outperform those trained from scratch and rival existing self-supervised learning methods. Our synthetic data aims to supplement the 3D point clouds observed by airborne LiDAR segmentation models and tackle the challenge of limited data availability in this field.

1. Introduction

Airborne light detection and ranging (LiDAR) is a remote sensing technology that utilizes near-infrared light to generate precise 3D models of the Earth's surface. The datasets acquired by airborne LiDAR sensors are typically enormous, often containing billions of points. To process a large number of point clouds, point cloud semantic segmentation (also called point cloud classification) is performed. Point cloud semantic segmentation involves predicting category labels for all points in a given point cloud. This task is highly challenging as a result of the scattered and irregular nature of aerial LiDAR data, which consist of huge numbers of points. Several architectures [13, 23, 31, 40] have been implemented to process point cloud data, including point-based networks, graph-based networks, voxel-based networks, and multi-view networks. Because LiDAR sensors acquire data in the form of 3D points, our focus is on exploring the efficacy of point-based networks for this task. The pioneering work for directly processing point cloud data was PointNet [40]. Qi et al. [41] extended the capabilities of PointNet by incorporating local geometric information through a hierarchical neural network, resulting in the creation of PointNet++. Inspired by these networks, recent studies [10,57,67] have focused on redefining sampling and augmenting features using knowledge from other fields to improve performance. To utilize deep learning methods effectively, it is necessary to develop annotated datasets acquired by airborne LiDAR sensors. Although automated labeling is possible for certain elements such as the ground and planar surfaces like buildings, other objects necessitate manual annotation as a result of their diverse shapes and relatively low representation, often comprising less than 1% of the total data points. However, labeling point cloud data requires a substantial amount of time and effort, leading to a shortage of large, annotated 3D datasets.

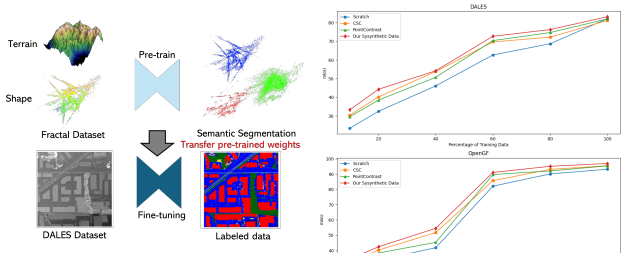
To address this issue, we propose the use of self-supervised learning (SSL) on unlabeled point clouds, drawing inspiration from the success of SSL methods in natural language processing [16, 17, 43] and computer vision [9, 18, 21, 32]. This approach aims to obtain meaningful representations for semantic scene segmentation tasks. This paper introduces a point cloud pre-training method that automatically constructs a synthetic point cloud dataset based on the natural laws governing 3D structures. Specifically, we apply the concept of formula-driven supervised learning (FDSL) to 3D vision, generating infinite training data from a mathematical formula, as proposed by Kataoka et al. for 2D vision [30]. We employ a mathematical formula rooted in fractal geometry [34], which is highly relevant to both natural and artificial objects in real-world 3D scenes. An example of applying a fractal-based training data creation method to 3D data was proposed by Yamada et al., who created a dataset by rendering a generated fractal model into multi-view images [58]. Previous studies in the field

of medical image processing have demonstrated the effectiveness of FDSL on 3D data for segmentation tasks [50]. Given that fractal geometry possesses two essential properties, namely self-similarity and non-integer dimensions, we believe it can be used to generate detailed 3D structures such as vegetation, which cannot be replicated by simple computer-aided design (CAD) models. Our proposed synthetic point cloud fractal dataset significantly enhances representation learning for semantic segmentation. By leveraging fractal geometry, a fundamental aspect of real-world structures, it is possible to generate 3D models and scenes that closely resemble natural environments automatically. This approach eliminates the need for manual labor in constructing outdoor point cloud datasets, as it adheres to natural laws described by mathematical formulations. The main contributions of this study can be summarized as follows.

- We propose a synthetic fractal dataset automatically generated using natural 3D terrain and objects with fractal geometry to simulate the point clouds observed by airborne LiDAR. This framework notably eliminates the need for data collection and annotation.

- The proposed synthetic fractal dataset facilitates the acquisition of feature representations for 3D segmentation during the pre-training phase, as shown in Fig. 1(a).

- By employing a detector pre-trained on the synthetic fractal dataset, we achieve improved performance on 3D segmentation tasks for representative outdoor point cloud datasets observed by airborne LiDAR, including DALES and OpenGF, as illustrated in Fig. 1(b).



(a) Pre-training 3D semantic segmentation for outdoor point cloud observed by airborne LiDAR with synthetic fractal dataset

(b) Fine-tuning results for a limited DALES training data

Figure 1. Pre-training effects on our synthetic fractal dataset as a family of formula-driven supervised learning. Although the proposed method does not use real data, it is a better pre-training approach to understand a 3D scene, especially in a limited data scenario

2. Related Studies

Datasets Current datasets for large-scale point cloud segmentation can be categorized into three main types: indoor, autonomous driving, and airborne.

Indoor Early datasets in this category such as SUN RGB-D [48], NYUv2 [47], and S3DIS [3] consist of RGB-D sequences captured with short-range depth scanners. These datasets feature low resolution and limited semantic annotations. Additional datasets [11, 15, 46] provide more extensive annotations but their performance on less common classes is limited by the resolution of the laser-scanned ground-truth geometry. ARKitScenes [5] and ScanNet++ [60] address this issue by integrating RGB images with high-resolution 3D scene geometry captured by lasers, offering both sparse and dense semantic annotations.

Autonomous driving This category encompasses datasets tailored for autonomous driving applications, where data are collected using LiDAR scanners and RGB cameras mounted on vehicles [6, 7, 12, 37, 39, 45, 49, 51]. These mobile LiDAR datasets, which are characterized by low-angle viewpoints focused on driving-related segmentation tasks, often exhibit occlusions in their point clouds such as missing building rooftops. Although these datasets fulfill the needs of autonomous driving, they are less suitable for other domains such as public utility management and urban planning.

Airborne These datasets are crucial for advancing research and applications in remote sensing, environmental monitoring, autonomous navigation, public utility management, and urban planning. They are primarily collected using airborne LiDAR [44, 53, 59, 68] or photogrammetry with SfM/MVS [8, 24, 33]. Unlike DALES [53], the ECLAIR dataset [35] offers colorized, large-scale point clouds that include high-resolution 3D geometry with accurate semantic labels and information on the number of LiDAR returns for each point. Additionally, OpenGF [42] was proposed as a dataset for evaluating the task of separating ground and non-ground points, as well as general ground object classification.

Pre-training Contrastive algorithms pre-train a backbone network by discerning similarities and differences between samples. PointContrast [56] is a foundational method in this area, generating two point clouds from varying perspectives and comparing their point feature similarities for pre-training. Subsequent studies have enhanced network performance by improving data augmentation techniques [55, 65] and incorporating cross-modal data [1, 26, 63].

Conversely, generative pre-training techniques focus on reconstructing masked parts of data or their 2D projections to train an encoder. Approaches such as Point-BERT [61] and Point-MAE [38] adapt concepts from bidirectional en-

coder representations from transformers [17] and masked autoencoders [20] to the context of point cloud data. The TAP [54] and Ponder [25] models enhance backbone training by generating 2D projections of point clouds.

Point-M2AE [64] is a hierarchical network designed to capture geometric and feature information incrementally. Joint-MAE [19] explores the interplay between 2D images and 3D point clouds, using hierarchical modules for cross-modal interactions to reconstruct masked elements in both modalities.

In contrast to the structural advancements observed in Point-M2AE and Joint-MAE, our approach focuses on refining the training process. By leveraging the progressive guidance features of conditional diffusion models, our method enables the backbone to acquire hierarchical geometric priors by denoising point clouds across varying levels of noise.

Formula-driven supervised learning. FDSL approaches [27, 29, 30, 36] generate extensive datasets using mathematical formulas, thereby eliminating the need for manual image collection and annotation. Research by Kataoka et al. [30] has demonstrated that a model pre-trained on a synthetic 2D fractal dataset can achieve performance levels comparable with those of models pre-trained on ImageNet for certain image classification tasks.

These approaches are particularly effective because they leverage pre-training on synthetic images generated from fractals, thereby eliminating the need for natural images entirely. Our hypothesis is that the effectiveness of this approach stems from training on fractals, which are prevalent in nature and represent a wide range of real-world patterns more comprehensively than datasets such as ImageNet. Additionally, this study underscores the importance of fractals, as we believe that pre-training on natural 3D structures can significantly enhance the understanding of real-world 3D scenes.

3. Proposed Method

We introduce a fractal dataset for pre-training to enhance feature extraction from outdoor point clouds observed by airborne LiDAR. The proposed dataset consists of auto-generated 3D fractal shapes and 3D fractal terrain. The construction of the fractal dataset involved five key procedures.

- **Automatic 3D Fractal Terrain Generation:** Based on fractional Brownian motion (fBm), we present a method for the automatic generation of 3D terrain models.
- **Automatic 3D Shape Generation:** Based on a 3D iterated function system (3D IFS) [4], we present a method for the automatic generation of 3D fractal shapes.
- **Category Definition:** We define categories based on the data distribution of the 3D fractal shapes.
- **Instance Generation:** For each category, we generate instances using a novel augmentation method called FractalNoiseMix.
- **3D Fractal Scene Generation:** Finally, we automatically generate 3D fractal scenes utilizing the 3D fractal shapes and fractal terrain.

An overview of our framework is presented in Fig. 2.

3.1. 3D Fractal terrain generation

Fractals are well-suited for terrain generation because they can naturally replicate the complexity and detail observed in real-world landscapes. Their self-similar property, where the structure appears similar at any magnification level, makes them ideal for creating terrains that exhibit realism across various scales.

Fractional Brownian Motion One common approach to fractal-based terrain generation is fBm. fBm involves layering noises of multiple frequencies to create a height field that mimics natural terrain. By sampling noise at different scales and combining these samples, fBm generates a height map representing a terrain's surface. This method ensures that the terrain possesses both large, smooth features and small, intricate details, resulting in a more realistic appearance, as shown in Fig. 3.

Hydraulic and Thermal Erosion To enhance the realism of fractal-generated terrains further, additional erosion processes can be simulated. Hydraulic erosion simulates the effect of water flowing over terrain, transporting sediment from elevated areas to lower areas, thereby creating valleys and riverbeds. This process mimics the natural erosion caused by rainfall and river flow.

In contrast, thermal erosion simulates the effect of material collapsing from steep slopes and accumulating at the base. This process smooths out sharp ridges and cliffs, creating more natural-looking hills and valleys. By applying these erosion techniques, the terrain can be further refined to achieve a closer resemblance to real-world landscapes.

Iterated Function System Another powerful method for fractal terrain generation is the IFS. The IFS utilizes a set of affine transformations such as rotations, translations, and scaling to generate self-similar structures recursively. This method can create highly detailed and intricate terrains by repeatedly applying these transformations to an initial shape or point cloud.

The IFS process begins by defining multiple transformations with their associated probabilities. An initial point

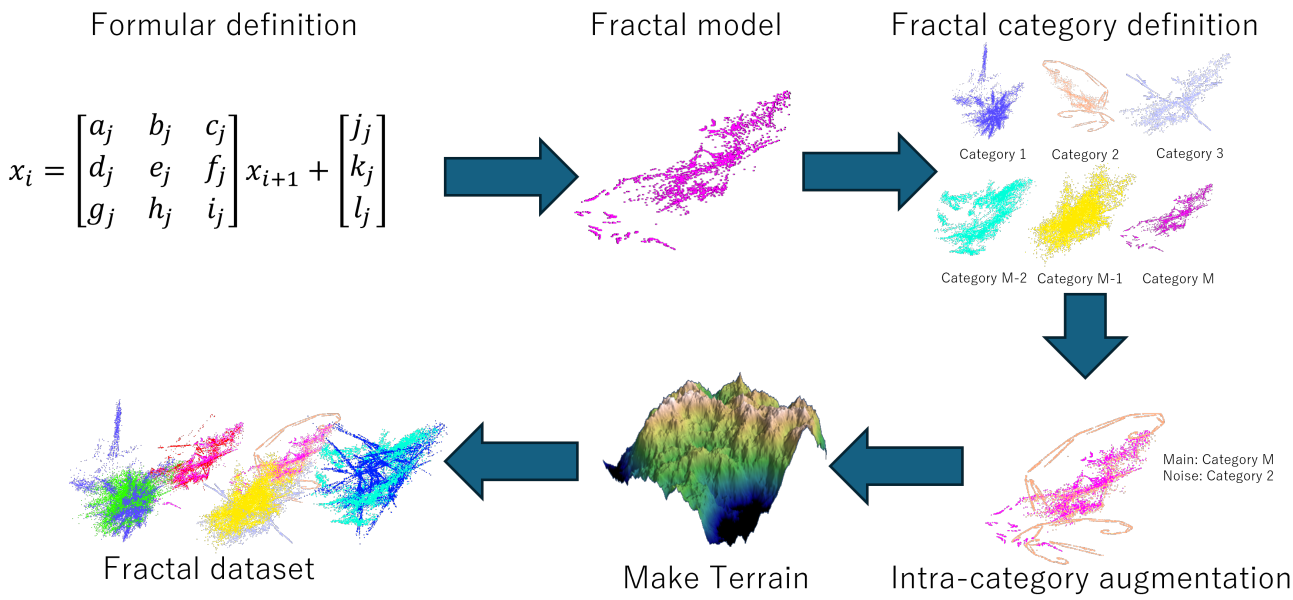


Figure 2. Overview of the formula-driven supervised learning(FDSL) framework for 3D semantic segmentation with 3D point clouds observed by airborne LiDAR. We generate synthetic 3D point clouds of terrain and shapes using fractal geometry. The proposed fractal shapes are automatically constructed by defining fractal categories using variance thresholds and instance augmentation with FractalNoiseMix. A 3D fractal scene is generated by randomly selecting 3D fractal shapes and translating them from the origin on the fractal terrain.

cloud is then generated and one of the transformations is applied based on the defined probabilities. This process is repeated for a predetermined number of iterations, each time applying a transformation to the current point cloud. The result is a complex terrain that exhibits the self-similar characteristics of fractals.

Figure 3. Examples of fractal terrain

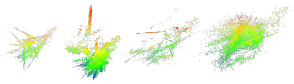


Figure 4. Examples of fractal shapes

3.2. 3D Fractal shape generation

We generate 3D fractal shapes from infinite pairs of 3D fractal parameters and fractal categories using the 3D IFS, which leverages the types of fractal geometry commonly found in the real world. We hypothesize that by utilizing fractal geometry, we can effectively represent complex patterns in 3D shapes using the 3D IFS, thereby enhancing 3D scene understanding in real-world environments. A 3D fractal shape is automatically generated through the following five steps.

1. Multiple affine transformations and their selection probabilities are randomly set.
2. An initial point cloud is defined at the origin coordinates and set as the current point cloud.
3. One of the affine transformations is selected based on the predefined selection probabilities.
4. The current point cloud undergoes an affine transformation to become the next point cloud using the selected affine transformation.
5. Steps 3 and 4 are performed recursively up to a set number of N iterations.

A 3D fractal shape is generated by iteratively applying a 3D affine transform T_j to an initial point cloud. In the present study, for the sake of simplicity, we introduced homogeneous coordinates to handle affine transforms. In homogeneous coordinates, a 3D point cloud $\mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \in \mathbb{R}^3$ is described as $\hat{\mathbf{x}} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \in \mathbb{R}^4$, where the notation $\hat{\cdot}$ indicates that the point is considered in homogeneous coordinates. Note that 3D affine transformations include rotation, translation, scaling, and skewing.

To generate 3D fractal shapes automatically, we apply affine transformations randomly. To construct a 3D IFS

set, affine transformations $\{T_j \in \mathbb{R}^{4\times4} \mid 1 \leq j \leq N\}$ are generated, where the elements of the affine transformation matrices are sampled from a uniform distribution in the range of $[-1.0, 1.0]$. When an initial point $\mathbf{x}_0$ is given, a 3D affine transformation T_j creates a 3D fractal model $P = \{\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_n\}$ by applying

$$\hat{\mathbf{x}}_i = T^i \hat{\mathbf{x}}_{i-1} \tag{1}$$

for i from zero to n, where n is the number of iterations. The probability of selecting T_j is denoted as P_{T_j}. Here,

$$p_j = \frac{|\det T_j|}{\sum_{j=1}^{N} |\det T_j|}.$$

Note that the scaling factor of an affine transform T_j is given by $|\det T_j|$. Next, we set the original coordinate location as the initial point cloud P_0 and select an affine transformation from the 3D IFS according to the probabilities p_j. A 3D fractal shape is generated over 4,000 iterations.

3.3. Augmentation

The process of variance binning defines distinct fractal categories, with each category corresponding uniquely to one 3D fractal shape. To enhance the diversity of these 3D fractal shapes, we introduce Fractal noise mixing (FNM) for augmentation, which is inspired by Point Mixup [14]. Unlike Point Mixup, which enhances data by interpolating between training samples to create intermediate categories, our approach focuses on intra-category augmentation to improve the effectiveness of fractal dataset pre-training.

FNM involves blending major and minor fractal categories or classes. For example, once a major fractal category is chosen, it comprises 80% of the final 3D fractal shape. Subsequently, we randomly select and incorporate 20% of a minor point cloud into the 3D fractal shape to complement the major fractal structure. It should be noted that during the classification of 3D fractal shapes, the major fractal category is considered as the primary fractal category. While random point clouds could be used for augmentation, this approach may result in the loss of critical fractal shape features. Therefore, we use FNM to ensure that these essential features are preserved.

3.4. 3D Scene generation

To create a synthetic outdoor 3D scene, we start by randomly sampling multiple objects from predefined 3D fractal shapes. The number of objects in each scene is determined using a Poisson distribution. We then generate 3D bounding boxes and rotate the fractal shapes about the z-axis.

Additionally, we set the scale factor for the x axis randomly between 0.75 and 1.25. This scale factor is then adjusted by a coefficient for the aspect ratio (ranging between 0.9 and 1.1) along the y and z axes. This method accounts for the minor variation in object scale typically observed in 3D outdoor datasets. The orientation of each 3D fractal shape can be randomly rotated about the z axis to introduce variability during training. Because these fractal shapes lack a defined front, the rotation angle is randomly set within the range $[-180°, 180°]$.

Finally, to align the 3D fractal shapes with terrain structures, we translate them onto the z plane. This process involves setting the x and y coordinates of each instance generated by the 3D fractal shapes as the centroid of the shape and redefining the coordinates within the range $[-7.5, 7.5]$. Note that the minimum z coordinates for each 3D fractal shape may not align on the same z plane.

In airborne LiDAR observations, real-world objects often appear to float based on the presence of elevated structures such as building roofs and power lines. Therefore, the 3D fractal shapes are placed in non-overlapping positions within the scene.

4. Experiments

In this section, we first describe the pre-training process of our fractal dataset and its fine-tuning for downstream datasets. We then present an analysis of experimental results to demonstrate our method's advantages over 3D scenes composed of CAD models. Finally, based on the results, we compare the performance of the fractal dataset to previous pre-training methods on two 3D segmentation benchmarks.

4.1. Experimental setting

Pre-training on the fractal dataset In this study, we employed KP-Conv and PointTransformer to train an end-to-end 3D semantic segmentation network. Unlike previous methodologies, our approach enables the acquisition of robust feature representations for semantic segmentation during the pre-training phase.

To construct the fractal dataset pre-trained models, we configured the training parameters as follows. Pre-training was conducted for a minimum of 1.8 million iterations with a batch size of 64 and a learning rate of 0.004. Each generated fractal point cloud scene encompassed an area of 30 square meters. First, we generated a fractal terrain of 250 square meters in size, as illustrated in Fig. 3. Each parameter was determined by random numbers for each scene and 10,000 scenes were generated, as illustrated in Fig. 4. Because fractal terrain is output as a surface, a point cloud was randomly generated from this surface with approximately 10 points per square meter. Next, we generated a fractal shape to place on the fractal terrain. The input point clouds from each scene were randomly sampled to extract 40,000 points. Generating the fractal dataset, which contains 1,000 categories with 500 instances per category in 10,000 scenes, required approximately two days. The pre-training process

Table 1. Overview of the selected methods on the DALES data set. We report the mean IoU and per class IoU, for each category.

Method	Pre-train	*mIoU*	*Ground*	*Vegetation*	*Cars*	*Trucks*	*Power Lines*	*Poles*	*Fences*	*Buildings*
KPConv [52]	fractal	83.19	97.4	97.4	87.1	46.5	92.9	79.8	67.6	96.8
	ModelNet	81.15	97.1	94.1	85.3	41.9	95.5	75	63.5	96.6
	S3DIS	81.19	97.2	93.8	85.9	42.6	92.1	76.2	65.1	96.6
Point Transformer [66]	fractal	75.75	97.8	96.5	86.4	39.1	80.2	46.7	67.3	92
	ModelNet	68.26	94.1	91.2	75.4	30.3	79.9	40	46.2	89.1
	S3DIS	68.51	94.4	91.8	75.9	31.1	80.5	39.9	45.9	88.6

was completed in six days using four NVIDIA Tesla V100 GPUs.

Fine-tuning for semantic segmentation Next, we evaluated the fractal dataset pre-trained model using fine-tuning datasets. The fine-tuning datasets we used, which were collected by airborne LiDAR, were DALES [53] and OpenGF [42]. These datasets, which represent outdoor scenes, are frequently employed in 3D semantic segmentation tasks. Fine-tuning was conducted over 180 epochs with a batch size of 64 and an initial learning rate of 0.01. The learning rate was reduced at intervals of 40, 80, 120, and 160 epochs. The input point clouds were randomly sampled to extract 40,000 points for each dataset.

4.2. Results

Effects of pre-training In this subsection, to understand the effects of pre-training tasks and demonstrate our method's advantages over CAD models, we present the results of multiple experiments. We investigated the following question. Which is the most effective for pre-training: 3D fractal models, CAD models, or indoor datasets (see Tables 1 and 2)?. We evaluated the effectiveness of our 3D fractal dataset using 3D fractal models.

We compared the pre-training performance of our 3D fractal dataset with that of 3D scenes composed of CAD models from ModelNet [62] and an indoor dataset from S3DIS [3]. As shown in Tables 1 and 2, the fractal dataset pre-trained KP-Conv and PointTransformer model outperformed the models pre-trained on the ModelNet dataset. The performance of KP-Conv improved by +2.5% on the DALES dataset and +4.1% on OpenGF in terms of mean intersection over union (mIoU), and the performance of PointTransformer improved by +10.0% on the DALES dataset and +6.0% on OpenGF in terms of mIoU. Additionally, the fractal dataset pre-trained KP-Conv and PointTransformer models outperformed the models pre-trained on the 3DSDIS dataset [2]. The performance of KP-Conv improved by +2.5% on the DALES dataset and +1.4% on OpenGF in terms of mIoU, and the performance of PointTransformer improved by +11.0% on the DALES dataset and +3.0% on OpenGF in terms of mIoU. Both models were highly effective for small objects (e.g., poles and cars) in the DALES dataset.

Between ModelNet and the indoor data, pre-training on the fractal dataset had less effect when using indoor data. ModelNet's weights are tuned for classification tasks and only the encoder parts of KPConv and PointTransformer can be trained, whereas S3DIS represents a segmentation task, so all weights except for those in the classification layer can be reused.

Consequently, we confirmed that our 3D synthetic fractal dataset generated based on fractal geometry is more effective than 3D pre-trained models with well-organized surface data such as ModelNet and indoor scenes, indicating that fractal geometric features are essential for effective pre-training. The proposed fractal dataset better accommodates pre-training on complex geometric shapes compared with CAD models or indoor datasets. This suggests that the fractal dataset can capture more diverse variations and common 3D patterns found in the real world, a significant advantage attributable to its construction based on fractal geometry. There was a noticeable difference in the effect of the fractal dataset between OpenGF and DALES data. This is thought to be because OpenGF is a dataset containing a large amount of vegetation such as forests, making this dataset more compatible with fractal shapes. Additionally, the reason why PointTransformer was more effective than KP-Conv on the fractal dataset is thought to be related to the scaling law of the transformer [28]. With conventional datasets, the transformer structure cannot be fully utilized, so PointTransformer performs worse than KP-Conv, which does not use a transformer. Our pre-training process using a large-scale fractal dataset is thought to be more compatible with PointTransfomer.

Comparison with other pre-training methods Here, we compare the proposed fractal dataset with self-supervised learning methods such as PointContrast [56] and CSC [22] in terms of pre-training effectiveness(see Table 3). In this experiment, we utilized KP-Conv and PointTransformer.

KP-Conv pre-trained with the fractal dataset resulted in a +1.2% improvement on DALES and +2.8% improvement on OpenGF in terms of mIoU compared with training from scratch with random values. PointTrasformer pre-trained with the fractal dataset resulted in a +8.0% improvement

Table 2. Overview of the selected methods on the OpenGF data set. We report the per class IoU, for each category.

Method	Pre-train	*mIoU*	*non-ground*	*ground*
KPConv [52]	Fractal	97.0	96.6	97.3
	ModelNet	93.2	93.7	92.6
	S3DIS	95.6	96.1	95.2
Point Transformer [66]	Fractal	95.6	94.5	96.7
	ModelNet	90.2	89.0	91.4
	S3DIS	92.9	92.5	93.4

Table 3. Comparison of pre-training methods.

PreTrain Method	Dataset	Model	mIoU
PointContrast [56]	DALES	KP-Conv	82.01
		PointTransformer	74.22
	OpenGF	KP-Conv	95.38
		PointTransformer	94.53
CSC [22]	DALES	KP-Conv	81.21
		PointTransformer	73.19
	OpenGF	KP-Conv	95.52
		PointTransformer	94.87
Random value	DALES	KP-Conv	82.15
		PointTransformer	70.12
	OpenGF	KP-Conv	93.19
		PointTransformer	92.99
fractal dataset	DALES	KP-Conv	83.19
		PointTransformer	75.75
	OpenGF	KP-Conv	96.96
		PointTransformer	95.60

on DALES and +4.0% improvement on OpenGF in terms of mIoU compared with training from scratch with random values. Furthermore, we observed that the performance of fractal dataset is relatively higher than that of previous state-of-the-art self-supervised learning methods. Our fractal dataset yields performance approximately equivalent to that of CSC and PointContrast. Notably, the performance of the fractal dataset when using PointTransformer is better than that of PointContrast on DALES and OpenGF. The advantage of our method compared with existing SSL approaches is that it can reproduce various three-dimensional geometric scenarios that occur in nature using fractals, and because the amount of data can theoretically be increased infinitely, it has the major advantage of being able to assist in training deep learning models using large amounts of data.

Limited training data Furthermore, as illustrated in Fig. 1, the fractal dataset yields superior performance under conditions with limited training data and annotations compared with previous SSL methods. This underscores the importance of pre-training with a substantial number of 3D scenes for datasets with limited annotations. Given the high annotation costs in the 3D vision field, the concept of constructing 3D datasets using FDSL without the need for manual data collection and annotation offers a promising direction for 3D vision research.

5. Conclusion

In this paper, we introduced fractal-based point cloud generation, a novel FDSL approach inspired by the fractal geometry commonly found in natural 3D structures. Our synthetic fractal dataset, which facilitates pre-training for point clouds observed outdoors, consists of two key elements: fractals representing topography and fractals representing objects such as vegetation. The primary feature of our approach is the automatic construction of 3D datasets, eliminating the need for observed data and human annotation, in contrast to previous supervised learning methods.

Our experimental results demonstrated that the proposed fractal dataset significantly enhances the performance of 3D semantic segmentation tasks such as land-use classifi-

cation and filtering. Notably, the fractal-dataset-pre-trained model proved to be more effective in scenarios with limited training data and annotations compared with previous SSL methods, as it allows for the pre-training of the entire network.

We have established a conceptual framework for constructing effective pre-training datasets for 3D segmentation. We believe that our fractal dataset will serve as an essential tool for advancing the understanding of 3D scenes in the future.

Acknowledgments

Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used. The study was conducted with funding from the research digital transformation (DX) initiative within AIST. We also extend our gratitude to the research teams of OpenGF and DALES for making their datasets publicly available, which greatly contributed to the success of this study.

References

[1] Mohamed Afham, Isuru Dissanayake, Dinithi Dissanayake, Amaya Dharmasiri, Kanchana Thilakarathna, and Ranga Rodrigo. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9902–9912, 2022. 2

[2] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 6

[3] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *CVPR*, pages 1534–1543, 2016. 2, 6

[4] Michael F Barnsley. *Fractals everywhere*. Academic press, 2014. 3

[5] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. 2021. 2

[6] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *ICCV*, pages 9297–9307, 2019. 2

[7] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, pages 11621–11631, 2020. 2

[8] Gülcan Can, Dario Mantegazza, Gabriele Abbate, Sébastien Chappuis, and Alessandro Giusti. Semantic segmentation on swiss3dcities: A benchmark study on aerial photogrammetric 3d pointcloud dataset. *Pattern Recognition Letters*, 150:108–114, 2021. 2

[9] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018. 1

[10] Mariona Caros, Ariadna Just, Santi Segui, and Jordi Vitria. Object segmentation of cluttered airborne lidar point clouds. *Artificial Intelligence Research and Development. 356 (2022) 259-268*, 2022. 1

[11] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. pages 667–676, 2017. 2

[12] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *CVPR*, pages 8748–8757, 2019. 2

[13] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017. 1

[14] Yunlu Chen, Vincent Tao Hu, Efstratios Gavves, Thomas Mensink, Pascal Mettes, Pengwan Yang, and Cees GM Snoek. Pointmixup: Augmentation for point clouds. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 330–345. Springer, 2020. 5

[15] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, pages 5828–5839, July 2017. 2

[16] Andrew M Dai and Quoc V Le. Semi-supervised sequence learning. *Advances in neural information processing systems*, 28, 2015. 1

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1, 3

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 1

[19] Ziyu Guo, Xianzhi Li, and Pheng Ann Heng. Joint-mae: 2d-3d joint masked autoencoders for 3d point cloud pre-training. *arXiv preprint arXiv:2302.14007*, 2023. 3

[20] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Confer-*

ence on Computer Vision and Pattern Recognition (CVPR), June 2020. 1

[22] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15587–15597, 2021. 6, 7

[23] Jordan SK Hu, Tianshu Kuai, and Steven L Waslander. Point density-aware voxels for lidar 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8469–8478, 2022. 1

[24] Qingyong Hu, Bo Yang, Sheikh Khalid, Wen Xiao, Niki Trigoni, and Andrew Markham. Towards semantic segmentation of urban-scale 3d point clouds: A dataset, benchmarks and challenges. In *CVPR*, pages 4977–4987, 2021. 2

[25] Di Huang, Sida Peng, Tong He, Honghui Yang, Xiaowei Zhou, and Wanli Ouyang. Ponder: Point cloud pre-training via neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16089–16098, 2023. 3

[26] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22157–22167, 2023. 2

[27] Nakamasa Inoue, Eisuke Yamagata, and Hirokatsu Kataoka. Initialization using perlin noise for training networks with a limited amount of data. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 1023–1028. IEEE, 2021. 3

[28] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020. 6

[29] Hirokatsu Kataoka, Asato Matsumoto, Ryosuke Yamada, Yutaka Satoh, Eisuke Yamagata, and Nakamasa Inoue. Formula-driven supervised learning with recursive tiling patterns. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4098–4105, 2021. 3

[30] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without natural images. *International Journal of Computer Vision (IJCV)*, 2022. 1, 3

[31] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4558–4567, 2018. 1

[32] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6874–6883, 2017. 1

[33] Xinke Li, Chongshou Li, Zekun Tong, Andrew Lim, Junsong Yuan, Yuwei Wu, Jing Tang, and Raymond Huang. Campus3d: A photogrammetry point cloud benchmark for hierarchical understanding of outdoor scene. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 238–246. Association for Computing Machinery, 2020. 2

[34] Benoit B Mandelbrot and Benoit B Mandelbrot. *The fractal geometry of nature*, volume 1. WH freeman New York, 1982. 1

[35] Iaroslav Melekhov, Anand Umashankar, Hyeong-Jin Kim, Vladislav Serkov, and Dusty Argyle. Eclair: A high-fidelity aerial lidar dataset for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 2

[36] Kodai Nakashima, Hirokatsu Kataoka, Asato Matsumoto, Kenji Iwata, Nakamasa Inoue, and Yutaka Satoh. Can vision transformers learn without natural images? *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2):1990–1998, Jun. 2022. 3

[37] Yancheng Pan, Biao Gao, Jilin Mei, Sibo Geng, Chengkun Li, and Huijing Zhao. Semanticposs: A point cloud dataset with large quantity of dynamic instances. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 687–693, 2020. 2

[38] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 604–621. Springer, 2022. 2

[39] Quang-Hieu Pham, Ramanpreet Singh Pahwa Pierre Sevestre, Chun Ho Pang Huijing Zhan, Vijay Chandrasekhar Yuda Chen, Armin Mustafa, and Jie Lin. A*3d dataset: Towards autonomous driving in challenging environments. 2020. 2

[40] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1

[41] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 1

[42] Nannan Qin, Weikai Tan, Lingfei Ma, Dedong Zhang, and Jonathan Li. Opengf: An ultra-large-scale ground filtering dataset built upon open als point clouds around the world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1082–1091, 2021. 2, 6

[43] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1

[44] F. Rottensteiner, G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf. The isprs benchmark on urban object classification and 3d building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I-3:293–298, 2012. 2

[45] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Paris-lille-3d: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *The International Journal of Robotics Research*, 37(6):545–557, 2018. 2

[46] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *ECCV*, pages 125–141. Springer-Verlag, 2022. 2

[47] Nathan Silberman and Rob Fergus. Indoor scene segmentation using a structured light sensor. pages 601–608, 2011. 2

[48] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, pages 567–576, 2015. 2

[49] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, pages 2446–2454, 2020. 2

[50] Ryu Tadokoro, Ryosuke Yamada, and Hirokatsu Kataoka. Pre-training auto-generated volumetric shapes for 3d medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4740–4745, 2023. 2

[51] Weikai Tan, Nannan Qin, Lingfei Ma, Ying Li, Jing Du, Guorong Cai, Ke Yang, and Jonathan Li. Toronto-3D: A large-scale mobile lidar dataset for semantic segmentation of urban roadways. In *CVPRW*, pages 202–203, 2020. 2

[52] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 6, 7

[53] Nina Varney, Vijayan K Asari, and Quinn Graehling. Dales: A large-scale aerial lidar data set for semantic segmentation. In *CVPRW*, pages 186–187, 2020. 2, 6

[54] Ziyi Wang, Xumin Yu, Yongming Rao, Jie Zhou, and Jiwen Lu. Take-a-photo: 3d-to-2d generative pre-training of point cloud models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5640–5650, 2023. 3

[55] Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9415–9424, 2023. 2

[56] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020. 2, 6, 7

[57] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. Linking points with labels in 3d: A review of point cloud semantic segmentation. *IEEE Geoscience and remote sensing magazine*, 8(4):38–59, 2020. 1

[58] Ryosuke Yamada, Hirokatsu Kataoka, Naoya Chiba, Yukiyasu Domae, and Tetsuya Ogata. Point cloud pre-training with natural 3d structures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21283–21293, June 2022. 1

[59] Zhen Ye, Yusheng Xu, Rong Huang, Xiaohua Tong, Xin Li, Xiangfeng Liu, Kuifeng Luan, Ludwig Hoegner, and Uwe Stilla. Lasdu: A large-scale aerial lidar dataset for semantic labeling in dense urban areas. *ISPRS International Journal of Geo-Information*, 9(7), 2020. 2

[60] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, pages 12–22, 2023. 2

[61] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022. 2

[62] A. Khosla F. Yu L. Zhang X. Tang J. Xiao Z. Wu, S. Song. 3d shapenets: A deep representation for volumetric shapes. In *Computer Vision and Pattern Recognition*, 2015. 6

[63] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. Clip2: Contrastive language-image-point pretraining from real-world point cloud data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15244–15253, 2023. 2

[64] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *arXiv preprint arXiv:2205.14401*, 2022. 3

[65] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10252–10263, 2021. 2

[66] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. In *ICCV*, pages 16259–16268, 2021. 6, 7

[67] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H.S. Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16259–16268, October 2021. 1

[68] S. M. Iman Zolanvari, Susana Ruano, Aakanksha Rana, Alan Cummins, Rogério E. da Silva, Morteza Rahbar, and Aljoscha Smolic. Dublincity: Annotated lidar point cloud and its applications. In *BMVC*, 2019. 2