Pre-training Vision Transformers with Very Limited Synthesized Images

Ryo Nakamura[1,2,*], Hirokatsu Kataoka[1,*], Sora Takashima[1,3],
Edgar Josafat Martinez Noriega[1,3], Rio Yokota[1,3], Nakamasa Inoue[1,3]
[1]National Institute of Advanced Industrial Science and Technology (AIST),
[2]Fukuoka University, [3]Tokyo Institute of Technology

Abstract

Formula-driven supervised learning (FDSL) is a pre-training method that relies on synthetic images generated from mathematical formulae such as fractals. Prior work on FDSL has shown that pre-training vision transformers on such synthetic datasets can yield competitive accuracy on a wide range of downstream tasks. These synthetic images are categorized according to the parameters in the mathematical formula that generate them. In the present work, we hypothesize that the process for generating different instances for the same category in FDSL, can be viewed as a form of data augmentation. We validate this hypothesis by replacing the instances with data augmentation, which means we only need a single image per category. Our experiments shows that this one-instance fractal database (OFDB) performs better than the original dataset where instances were explicitly generated. We further scale up OFDB to 21,000 categories and show that it matches, or even surpasses, the model pre-trained on ImageNet-21k in ImageNet-1k fine-tuning. The number of images in OFDB is 21k, whereas ImageNet-21k has 14M. This opens new possibilities for pre-training vision transformers with much smaller datasets.

1. Introduction

Pre-training has become a standard procedure when training deep neural networks in computer vision [5, 45]. Pre-trained models are known to exhibit superior convergence and generalization for downstream tasks as compared to models that are trained from scratch. However, pre-training large vision models requires enormous data, which makes pre-training state-of-the-art vision models very expensive regarding the required amount of data and computation.

During the past decade, ImageNet has served as a common dataset for pre-training vision models [13]. Models pre-trained on the classification task of 1.28M images in ImageNet were transferred to Object Detection [15, 38, 45], Semantic Segmentation [40, 34], and Video Recognition [8, 2]. In the 2020s, the pre-training of Vision Transformers (ViT) [14] has become increasingly popular. However large vision transformers are said to require datasets that are much larger than ImageNet, such as JFT-300M/3B [36], in order to achieve their true potential.

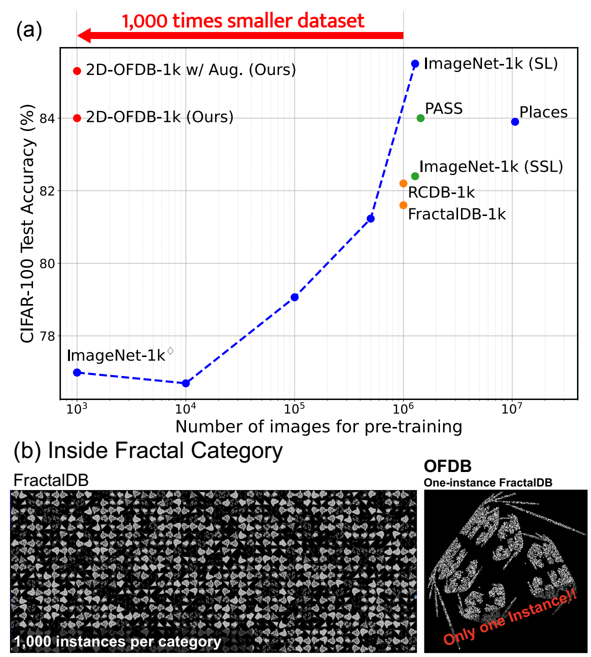


Figure 1: (a) Fine-tuning accuracy on CIFAR-100 for the number of images for pre-training. The line plot for ImageNet-1k indicates the results when using random sampling to reduce the data for pre-training. (b) One-instance fractal database (OFDB) consists of only 1,000 images in total. The figure shows the category representation. OFDB contains a single instance per category.

Pre-training increasingly larger models on increasingly larger datasets has shown a monotonic improvement in the accuracy of downstream tasks. However, creating datasets with billions of labeled images is not the ultimate solution to all our problems. First, the cost of manually labeling such

*equal contribution
GitHub code : `https://github.com/ryoo-nakamura/OFDB/`

huge datasets is prohibitive. Self-supervised learning (SSL) has received great attention due to its competitive performance when used to pre-train large models without requiring labeled images. DINO [7], MoCoV3 [11], and BEiT [4] have shown promising results in this regard. However, SSL still relies on a vast amount of unlabeled data.

Therefore, we should pause and ask the question, "Are we using all this data efficiently?" There have been efforts in this direction to reduce the amount of pre-training data while retaining accuracy on downstream tasks. For ViTs, DeiT [37] has demonstrated that distillation and data augmentation can enhance the pre-training effect of ImageNet-1k to match that of larger datasets. MAE [16] also show a high performance, even when trained on ImageNet-1k. More recent studies, such as "Training Vision Transformers with Only 2,040 Images" [44], show how ViTs can be pre-trained on relatively small datasets.

Another approach to those above is pre-training a ViT using formula-driven supervised learning (FDSL) [28, 21, 19, 32, 14], in which not even a real image is required. In a follow-up study, Kataoka *et al.* [19] created an alternative synthetic dataset with emphasis on the contours in the image and showed that it is possible to surpass the accuracy of a ViT pre-trained on ImageNet-21k by using a synthetic dataset of the same size. FDSL method can generate the labels automatically from the parameters used to generate the images, so there is no labeling cost. Furthermore, unlike SSL, FDSL does not even require real images. The fact that a ViT pre-trained on synthetic datasets can outperform a ViT pre-trained on a fairly large human-labeled dataset ImageNet-21k is significant. However, synthesizing more than a million images is costly. It is possible that many of the images in these synthetic datasets are redundant or are simply not contributing to the pre-training since these synthetic images are expanded with basic procedures from a single image. In case of FractalDB [21], a single image inside of category was augmented to 1,000 instances with image rotation, parameter fluctuation, and patch patterns. It is natural to rely on the combination of data augmentation methods in pre-training phase. In this context, it may be possible to reach the same fine-tuning accuracy for both one instance with the same data augmentations and preprocessed 1,000 instances.

In this paper, we present an FDSL approach to pre-train a ViT with a single instance per category. Therefore, we require only 1,000 images when the dataset contains 1,000 categories. The proposed dataset, i.e., the one-instance fractal database (OFDB), significantly improves data efficiency under the assumption that data augmentation is used during pre-training. In previous FDSL datasets, the different instances for each category are created through some form of manipulation of the original image that defines that category. Therefore, it is quite natural to wonder whether these instances can be created through data augmentation techniques. Although a detailed description is provided in Section 4, we disclose that the fractal instances can be replaced by data augmentation, e.g., image rotation. Based on the above considerations, we use a novel FDSL dataset that only requires a single image per category and training with data augmentation including random pattern augmentation and random texture augmentation as proposed data augmentation methods for FDSL pre-training. We validate this hypothesis by creating small, yet effective, pre-training datasets, 2D-OFDB and 3D-OFDB, respectively.

Our main contributions can be summarized as follows:
Conceptual contribution. We propose two datasets, 2D-OFDB and 3D-OFDB, which consist of only one representative fractal per category. For example, 2D-OFDB-1k consists of only 1,000 images but enables ViT to effectively learn visual representations for image classification. Along this line, we also implement random pattern augmentation and random texture augmentation for fractal pre-training.
Experimental contribution. We show that OFDBs achieve comparable performance to well-defined million-scale datasets (Figure 1 and Table 1). Furthermore, we show that the computational time of pre-training is reduced by 78.7%. In ImageNet-1k fine-tuning, 2D/3D-OFDB-21k performed at equal or better rates than baseline pre-training datasets with only 21k images. (see Table 2). We also show that OFDBs perform better than state-of-the-art methods for training ViTs on small datasets [44] (Table 3).

2. Related Work

2.1. Formula-driven supervised learning (FDSL)

FDSL is a form of learning strategy in which images and their corresponding labels are generated from a mathematical formula [21, 20, 28, 19, 32, 1, 42, 41]. Training on such synthetic datasets frees us from various ethical issues, e.g., societal biases and handling of copyrights and personal information [43, 3]. One of the representative datasets for FDSL is FractalDB, which generates fractal images from an iterated function system (IFS) [21].

Nakashima *et al.* [28] demonstrated that a ViT can be successfully pre-trained on FractalDB, which results in competitive accuracy on downstream tasks to a ViT pre-trained on ImageNet. More recently, Kataoka *et al.* [19] extended the FractalDB to two other datasets (ExFractalDB and RCDB), which comprise images with more emphasis on contours rather than textures. When fine-tuned on ImageNet-1k, the accuracy of the synthetic datasets (ExFractalDB-21k and RCDB-21k) exceeds that of ImageNet-21k. This is a significant result that raises fundamental questions regarding the role of real images when pre-training ViTs. Furthermore, there is ample room for improvement regarding the quality of these synthetic datasets. The ExFractalDB-21k and RCDB-21k dataset each have

21k categories and 1k instances per category. The instances are created by manipulating the original image, using a process similar to data augmentation. In the present study, we consider the possibility of replacing these instances with standard or our proposed data augmentation techniques in image classification.

2.2. Pre-training ViT on Limited Data

Since Dosovitskiy *et al.* published the ViT [14] paper in 2020, they have been replacing convolutional neural networks (CNN) [25, 24] for various computer vision tasks. However, large ViT models require large datasets, such as ImageNet-21k (14M images) and JFT-300M (300M images), to reach their full potential. The DeiT [37] uses data augmentation and distillation from CNNs to achieve the same pre-training effect using only ImageNet-1k (1.28M images). SSL, such as that by DINO [7], MoCoV3 [11], BeiT [4], and MAE [16], is also able to achieve similar performance using ImageNet-1k for pre-training. Other studies have attempted to reduce the size of the pre-training dataset even further. Cao *et al.* [44] modified the structure of ViT in order to enhance the information extracted from images and were able to pre-train ViT using only 2,040 – 8,144 images. This is three orders of magnitude smaller than ImageNet-1k, which opens new possibilities for pre-training vision transformers on small datasets.

We propose a one-instance fractal database (OFDB), where each category has only one instance. The hypothesis here is that the 1,000 instances of the original FractalDB can be replaced with standard or proposed data augmentation techniques. In this case, we only need a single instance per category, whereas the other instances are created during training through data augmentation.

3. Method

This section presents small and powerful datasets for FDSL. In contrast to previous million-scale datasets, such as ImageNet-1k, one of the proposed datasets, namely 2D-OFDB-1k, contains only 1,000 images for pre-training, but 2D-OFDB-1k enables ViT to effectively learn visual representations for image classification.

On the other hand, at the beginning of this section, we describe **'why fractal pre-training with one-instance per category can train visual representation'** as a curious scenario in FDSL datasets, especially on FractalDB. First, we consider instance augmentation of FractalDB. In FractalDB, the number of images is increased by (i) image rotation (x4), (ii) IFS parameters fluctuation (x25), and (iii) patch patterns (x10) from a representative image of the category found with category search. FractalDB [21] is pre-trained as a dataset of 1,000 instances of pre-processed images, but in a simple view, these could be replaced by data augmentation. Details will be discussed later, and Table 6 shows that the accuracy is almost the same with and without Rotation (2D-OFDB w/o rotation 84.0 vs. w/ rotation 84.1 on CIFAR-100). On the other hand, the performance rates with IFS fluctuation are significantly lower (2D-OFDB w/o IFS 84.0 vs. w/ IFS 81.6 on CIFAR-100). From these observations, we believe that a pre-training dataset with fewer instances like a dataset consists of one-instance per category and using basic data augmentation are the key technologies to reduce training time to an equivalent or better accuracy level. As the result, an image augmentation with patch patterns were newly implemented as random patch/texture augmentation so that they could be implemented within data augmentation, and it became clear that accuracy could be improved while reducing the data size to 0.1% amount.

3.1. Problem Settings

FDSL. The goal of FDSL is to pre-train neural networks without real images. One of the most successful approaches to achieve this goal is to synthesize a labeled dataset $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$ based on some mathematical formulas, such as fractals [21], where $\boldsymbol{x}_i$ is a synthesized image, $\boldsymbol{y}_i$ is a one-hot label vector, and N is the number of images. For FDSL, the cross-entropy loss is used, which is given by

$$\mathcal{L}_{\text{ce}}(\theta; D) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log p_{i,c}, \tag{1}$$

where $\boldsymbol{p}_i = f_\theta(\boldsymbol{x}_i) \in \mathbb{R}^C$ is the output vector of a learnable network f_θ, such as a ViT, θ is a set of parameters, and C is the number of categories. Typically, the number of images N should be equal to or more than one million in order to achieve good pre-training performance.

One-instance FDSL. The present paper proposes a challenging learning framework, namely *one-instance FDSL*, in an attempt at efficient and effective pre-training. The goal of one-instance FDSL is to pre-train neural networks with a dataset of representative images $D = \{\boldsymbol{x}_c\}_{c=1}^C$, where $\boldsymbol{x}_c$ is a single image that represents category c, *i.e.,* dataset D involves only one image instance per category. With this setting, the cross-entropy loss reduces to the following negative log-likelihood loss:

$$\mathcal{L}_{\text{nl}}(\theta; D) = -\frac{1}{C}\sum_{c=1}^{C}\log p_{c,c}. \tag{2}$$

This setting dramatically improves data-efficiency of pre-training because the setting omits N in Eq. (1).

3.2. One-instance Fractal Databases (OFDBs)

We propose OFDBs involving representative images of fractals. There are two variants, 2D-OFDB and 3D-OFDB, which consist of representative images of 2D and 3D fractals, respectively.

2D-OFDB. The first variant is a dataset that consists of representative 2D-fractal images. In order to create fractals, the iterated function system (IFS) [21] is used:IFS =

$\{\mathcal{X}; w_1, w_2, \cdots, w_M; p_1, p_2, \cdots, p_M\}$ where $\mathcal{X} = \mathbb{R}^2$ is a 2D Euclidean space, $w_j : \mathcal{X} \to \mathcal{X}$ is an affine transformation function, and p_j is a probability. Given an IFS and an initial point $v_1 \in \mathcal{X}$, a fractal S is obtained as a set of points $S = \{v_t\}_{t=1}^{\infty} \subset \mathcal{X}$ by $v_{t+1} = w^*(v_t)$, where w^* is a transformation sampled at each t under the probability distribution $p(w^* = w_j) = p_j$.

The 2D-OFDB $D_{\text{2D}} = \{\boldsymbol{x}_c\}_{c=1}^{C}$ consists of C representative images that are synthesized in the following three steps: First, a set of iterated function systems $\{\text{IFS}_c\}_{c=1}^{C}$ is randomly sampled. We use the sampling algorithm proposed in [21]. Second, with each IFS_c, a fractal S_c is randomly sampled. Finally, S_c is rendered into $\boldsymbol{x}_c$. Note that the original FractalDB [21] samples 1,000 fractals from each IFS_c. However, we found that most fractals are redundant, especially with image rotation (see Table 6 for details), if a data augmentation function is used when pre-training ViT. For comparison with ImageNet, we create two datasets, 2D-OFDB-1k/21k, by setting $C = 1,000/21,000$, respectively.

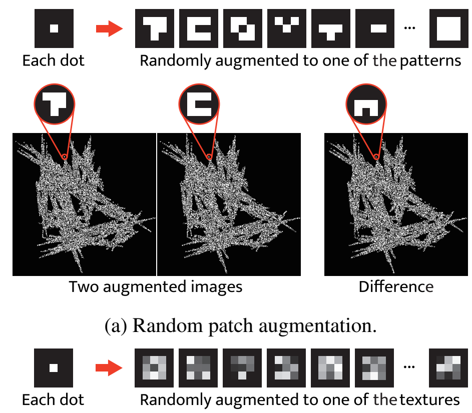
3D-OFDB. The second variant is a dataset consisting of 3D fractals. This dataset uses the 3D Euclidean space $\mathcal{X} = \mathbb{R}^3$ and 3D affine transformations with IFSs. The representative fractals are chosen by considering variance δ of point scattering in 3D space. We follow the previous study [19] to conduct the procedure. We create two datasets, 3D-OFDB-1k/21k, by setting $C = 1,000/21,000$, respectively.

3.3. Data augmentation for fractal images

In one-instance FDSL, we empirically found that the augmentation configuration proposed for the DeiT [37] is effective. However, given representative fractal images, there is still room for improvement because the configuration of the DeiT is empirically optimized for pre-training with real images. Here, we present two additional augmentation functions that boost pre-training with OFDBs. Also, note that our proposed data augmentation is OFDB-specific and cannot be applied to natural images.

Random pattern augmentation (Figure 2a). The image $\boldsymbol{x}_c$ of the representative fractal S_c is a binary (black-and-white) image of dots, each of which corresponds to a point v_t of the fractal. Given an image, $\boldsymbol{x}_c$, the random pattern augmentation augments each dot to a 3×3 pattern. The patch patterns are randomly sampled from the uniform distribution over the set of all binary patterns (there are $2^{3\times3} = 512$ patterns). Figure 2a shows an example of two augmented images and the difference between the images. We see that the overall fractal shape is the same for the two images, but the local patterns are different. Interestingly, the difference image obtained by this augmentation method makes the same fractal.

Random texture augmentation (Figure 2b). The random texture augmentation augments each dot to a 3×3 gray-scale texture, where each pixel value is randomly sampled



(a) Random patch augmentation.

(b) Random texture augmentation.

Figure 2: Proposed data augmentation methods.

from the uniform distribution over $\{0, 1, \cdots, 255\}$. Unlike random pattern augmentation, this augmentation makes dense images, where most of the nine pixels in each texture have non-zero values.

4. Experiments

4.1. Comparison with State-of-the-art Datasets

Fine-tuning results (Table 1). We conducted fine-tuning experiments on the CIFAR-10 (C10) [23], CIFAR-100 (C100) [23], Cars [22], Flowers [29], ImageNet-100 (IN100) [21], Places30 (P30) [21], and Pascal VOC 2012 (VOC12) [15] datasets. The proposed OFDBs are compared with six pre-training datasets: ImageNet-1k , Places-365 , PASS , FractalDB-1k , RCDB-1k , and ExFractalDB-1k . The results using a subset of ImageNet, which consists of 1,000 images obtained by randomly sampling one image per category, are also reported. Note that $^{\diamond}$indicates the subset. This subset is the same size as OFDBs. Here, we assign the ViT-Tiny (ViT-T) model with standard DeiT training configurations, including hyper-parameters.

From the experimental results in Table 1, we see that OFDBs achieved a higher accuracy with million-scale FDSL datasets, such as ExFractalDB-1k. Although they did not always surpass SL and SSL methods, note that they often achieved similar performance rates with only 1,000 images for pre-training, which is approximately 0.078% of images as compared to ImageNet-1k (1.28M images). We also see a significant difference between OFDBs and the ImageNet-1k$^{\diamond}$ subset in terms of average accuracy. In the proposed methods, the configuration of 2D-OFDB-1k with random pattern augmentation has a better average rate in the table. Thereafter, in the experiments, we assigned random pattern augmentation for 2D-OFDB.

Scaling experiments on ImageNet-1k (Table 2). In order to investigate the scalability of OFDBs, we increased the number of categories from 1,000 to 21,000 and applied these categories to ViT-T and ViT-Base (ViT-B) in Table 2. For comparison, the results of ImageNet-21k,

Table 1: Comparison of pre-training methods. Best values at each dataset scale are in bold. ◊ indicates a subset that consists of one randomly sampled image per class. ViT-T is used for all experiments. SL: supervised learning using cross-entropy loss, SSL: self-supervised learning using DINO [7]. Fine-tuning accuracies are reported.

Pre-training	#Img	Type	C10	C100	Cars	Flowers	VOC12	P30	IN100	Average
Scratch	–	–	78.3	57.7	11.6	77.1	64.8	75.7	73.2	62.6
Places-365 [45]	1.80M	SL	97.6	83.9	89.2	99.3	84.6	–	**89.4**	–
ImageNet-1k [13]	1.28M	SL	**98.0**	**85.5**	**89.9**	**99.4**	**88.7**	**80.0**	–	–
ImageNet-1k [13]	1.28M	SSL	97.7	82.4	88.0	98.5	74.7	78.4	89.0	86.9
PASS [3]	1.43M	SSL	97.5	84.0	86.4	98.6	82.9	79.0	82.9	**87.8**
FractalDB-1k [21]	1.00M	FDSL	96.8	81.6	86.0	98.3	80.6	78.4	88.3	87.1
RCDB-1k [19]	1.00M	FDSL	97.0	82.2	86.5	98.9	80.9	79.7	88.5	87.6
ImageNet-1k◊	**1,000**	SL	94.3	76.9	57.3	94.8	73.8	78.2	84.3	79.9
ImageNet-1k◊	**1,000**	SSL	94.9	78.0	71.2	94.6	75.5	78.6	84.9	82.5
2D-OFDB-1k (ours)	**1,000**	FDSL	96.9	84.0	84.5	97.1	79.9	79.9	88.0	87.2
2D-OFDB-1k w/ Aug. (ours)	**1,000**	FDSL	**97.2**	**85.3**	**87.6**	98.3	**81.4**	**80.4**	**89.5**	**88.5**
3D-OFDB-1k (ours)	**1,000**	FDSL	97.1	83.8	85.5	**98.4**	80.8	80.0	89.1	87.8
3D-OFDB-1k w/ Aug. (ours)	**1,000**	FDSL	97.0	84.7	85.6	98.3	81.2	79.8	88.9	87.9

Table 2: Scaled models/datasets with combinations of 21k categories and ViT-T/B models are used in ImageNet-1k fine-tuning. We also list GPU hours, batch size ('Batch'), and number of iterations ('#Iterations') in ViT-B pre-training.

Pre-training	#Img	Type	ViT-T	ViT-B	GPU hours	Batch	#Iterations
Scratch	–	–	72.6	79.8	–	–	–
ImageNet-21k	14M	SL	**74.1**	81.8	3,657	8,192	300k
FractalDB-21k	21M	FDSL	73.0	81.8	5,120	8,192	300k
ExFractalDB-21k	21M	FDSL	73.6	**82.7**	5,120	8,192	300k
RCDB-21k	21M	FDSL	73.1	82.4	5,120	8,192	300k
ImageNet-21k◊	21k	SL	71.0	81.1	1,132	1,024	300k
2D-OFDB-21k	21k	FDSL	**73.8**	82.2	1,088	1,024	300k
3D-OFDB-21k	21k	FDSL	73.7	**82.7**	1,088	1,024	300k

FractalDB-21k, ExFractalDB-21k, and RCDB-21k are reported. ImageNet-21k consists of 14M images, and the other three FDSL datasets consist of 21M images. The results for the ImageNet-21k◊ subset involving one image per category are also reported.

With ViT-T, 2D-OFDB-21k outperforms FractalDB-21k (73.8 vs. 73.0) and is comparable with ExFractalDB-21k (73.7 vs. 73.6). However, we see the performance gap between 2D-OFDB-21k and ImageNet-21k (73.8 vs. 74.1). This is because the images of ImageNet-21k for pre-training and the images of ImageNet-1k for fine-tuning overlap. This gap is reversed in ViT-B, where 2D-OFDB-21k is recorded 0.4 points higher than ImageNet-21k on ImageNet-1k fine-tuning. Moreover, the 3D-OFDB-21k pre-trained model was recorded 0.9 point higher than the model pre-trained on ImageNet-21k. Note that the computational time for pre-training in terms of GPU hours is reduced by 78.7% (1,088 vs. 5,120 in terms of GPU hours). We clarified that ViT pre-training can be more efficient in terms of both data amount and computational time.

Training on small datasets (Table 3). Instance Discrimination with Multi-crop and CutMix (IDMM), proposed by Cao *et al.* [44], which requires only 2,040 images for pre-training, is one of the most successful approaches for training ViTs on small datasets. In Table 3, we compare OFDBs with IDMM. Here, we used the fine-tuning configuration of IDMM for a fair comparison, *i.e.,* 800-epoch pre-training and 200-epoch fine-tuning on seven datasets of Flowers [29], Pets [31], DTD [12], Indoor-67 [33], CUB [9], Aircraft [27], and Cars [22]. Note that IDMM has two settings: internal pre-training and external pre-training. We used PVT v2[39] for the ViT model, as in Cao *et al.* The former uses the same dataset for pre-training and fine-tuning. This performs well, as reported in [44]. For the latter, we used an ImageNet subset, which consists of 2,040 randomly sampled images (with two or three images per category). We refer to this as IDMM-ImageNet.

The results are shown in Table 3. Although we used a limited number of images to pre-train a ViT, the result is much higher than the accuracy of training from scratch using the same procedure of 200-epoch fine-tuning. The proposed method is also better than the other well-organized pre-training methods, including IDMM [44] and SimCLR [10], with fewer pre-training images. In comparison with IDMM-ImageNet, 2D-OFDB-1k pre-training still performs at a higher accuracy, indicating that 2D-OFDB-1k

Table 3: Pre-training with small datasets. The pre-training and fine-tuning setting in [44] is used for evaluation. Note that the values '2,040 – 8,144' correspond to the number of pre-training images at each dataset. **Best** and **second-best** scores are in underlined bold and bold, respectively.

Pre-training	#Img	Flowers	Pets	DTD	Indoor-67	CUB	Aircraft	Cars	Average
Scratch	–	76.4	67.2	44.2	58.7	54.4	23.0	78.6	57.5
SimCLR [10]	2,040 – 8,144	90.1	82.8	62.3	66.6	**68.5**	74.4	89.3	76.3
IDMM [44]	2,040 – 8,144	92.4	83.2	66.9	68.5	**69.8**	73.4	87.8	77.4
IDMM-ImageNet [44]	2,040	90.5	82.4	66.8	**68.8**	66.8	91.8	87.6	79.2
2D-OFDB-1k (ours)	1,000	**93.7**	**84.6**	**67.5**	66.1	67.7	**95.0**	**91.0**	**80.8**
3D-OFDB-1k (ours)	1,000	**92.8**	**84.6**	**67.5**	**68.6**	67.9	**94.6**	**90.4**	**80.9**

Table 4: Comparison of object detection and instance segmentation. Several pre-trained models were validated on the COCO dataset. The best values for each learning type are shown in bold.

Pre-training	COCO Det AP_{50} / AP / AP_{75}	COCO Inst Seg AP_{50} / AP / AP_{75}
Scratch	63.7 / 42.2 / 46.1	60.7 / 38.5 / 41.3
ImageNet-1k	69.2 / 48.2 / 53.0	66.6 / 43.1 / 46.5
ImageNet-21k	**70.7 / 48.8 / 53.2**	**67.7 / 43.6 / 47.0**
ExFractalDB-1k	69.1 / 48.0 / 52.8	66.3 / 42.8 / 45.9
ExFractalDB-21k	69.2 / 48.0 / 52.6	66.4 / 42.8 / 46.1
RCDB-1k	68.3 / 47.4 / 51.9	65.7 / 42.2 / 45.5
RCDB-21k	67.7 / 46.6 / 51.2	64.8 / 41.6 / 44.7
ImageNet-21k$^\diamond$	63.8 / 42.0 / 45.5	60.7 / 38.3 / 41.0
2D-OFDB-21k	**67.6 / 46.4 / 51.0**	**64.6 / 41.6 / 44.7**
3D-OFDB-21k	67.1 / 46.3 / 51.0	64.4 / 41.4 / 44.4

pre-training is highly beneficial, even when using a limited number of synthesized images.

COCO detection/instance segmentation (Table 4). We validate object detection and instance segmentation on the COCO [38] dataset. Here, we switch the backbone model from a ViT to a Swin Transformer [26] with Mask R-CNN [17] head. We perform training for 60 epochs on the COCO dataset. The proposed 2D-OFDB-21k pre-trained model scores are higher than in the case of training from scratch and are similar to those for the model pre-trained with ImageNet-1k. 2D/3D-OFDB-21k recorded similar rates of 67.1 and 64.4 (64.3 in 3D-OFDB-21k) at AP_{50} in detection and segmentation tasks.

4.2. Exploratory Study

In this subsection, we basically apply the data augmentation methods and hyper-parameters used in the paper on the DeiT, unless we mention the changed parameters from those of the DeiT. We use fine-tuning accuracy on CIFAR-100 (C100) as an evaluation measure.

Virtual camera for 3D-OFDB (Table 5). In ExFractalDB implementation, three axes (roll, pitch, yaw angle) are controlled to set the random viewpoint, which is then projected from the 3D model onto the 2D image. Here, we project the 3D model onto 2D images with random viewpoints using $\{1, 2, 3\}$ axes {roll, pitch, yaw}. Table 5 show the experi-

Table 5: Analysis on 3D-OFDB. Although ExFractalDB-1k previously adjusted one-axis with yaw angle, we adjusted three axes with {roll, pitch, yaw} angles.

Pre-Training	Axis	Acc.
3D-OFDB	1 (yaw)	**83.8**
	2 (pitch, yaw)	82.8
	3 (roll, pitch, yaw)	82.7
ExFractalDB	3 (roll, pitch, yaw)	83.1

mental results. We use 12 fixed viewpoints at every 30 degrees in two or three axes. Eventually, an improvement can no longer be expected by creating additional camera angles with roll, pitch, and yaw. Rather, a limited viewpoint from fixed 30-degree angles with only a yaw angle proves better.

Data augmentation (Table 6). Table 6 shows the effects of data augmentation for fractal images. We see that the proposed augmentation methods boost accuracy. In particular, random pattern augmentation is the most effective. With IFS augmentation, we see the pre-training performance decrease significantly. This is because the better performance of 2D-OFDB-1k vs. FractalDB-1k in Table 1 is due to the removal of IFS augmentation. Figure 3 shows examples of image augmented with IFS. In the augmented images, there are images with significantly altered shapes, particularly images with almost meaningless shapes, which may hurt pre-training during DeiT augmentation. Also, with rotation augmentation, which randomly rotates fractal images by 0, 90, 180, or 270 degrees, we see that the performance improvement is not significant. This is because rotation augmentation resembles flipping augmentation in the DeiT setting.

One-instance setting on real-image dataset (Table 7). We conduct some additional experiments on the ImageNet-1k$^\diamond$ subset. Here, we convert RGB images to gray-scale [18], binary [30], and Canny edge [6] images. Table 7 shows the results of pre-training with these converted images. This table also shows the results for 2D/3D-OFDB-1k.

We see that binary and Canny images performed better than RGB images when we used these images in the pre-training phase. However, none of the images outperformed 2D/3D-OFDBs. These results are consistent with the claim that "object contours are what matter in FDSL datasets",

Table 6: Effects of data augmentation for fractal images. Data-efficient image Transformer (DeiT) augmentation setting is used as a default.

DeiT	✓	✓	✓	✓	✓	✓
IFS		✓				
Rotation			✓			
Rand. Pat.				✓		✓
Rand. Text.					✓	✓
2D-OFDB	84.0	81.6	84.1	**85.3**	84.8	84.3
3D-OFDB	83.8	-	-	84.7	**85.1**	**85.1**

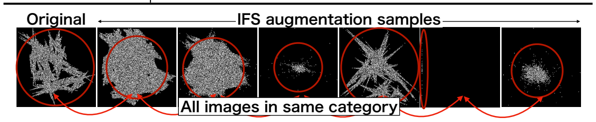


Figure 3: Sample images augmented with IFS

as noted in a previous paper [19]. In fact, in order to pre-train a ViT model with one instance per category, an RGB representation is not enough on ImageNet-1k. A contour-emphasized dataset with a Canny edge detector is more efficient for pre-training a ViT, which is good at learning object contours. In the one-instance setting, we found that the pre-training effect with contour-emphasized image representations was also improved.

Number of instances (Figure 4). Figure 4 shows the relationship between the number of image instances and accuracy in pre-training with ImageNet-1k (SL) and 2D-OFDB-1k (FDSL). In the figure, the transition in accuracy is shown when #instances are set to {1, 10, 100, 500, 1,000}. Formula-driven supervised learning has the highest accuracy when #instances is 1; there is a slight decrease in accuracy up to 100 instances and almost no change thereafter. On the other hand, the lowest accuracy was observed for one instance when ImageNet-1k was used, and the accuracy improved with each increase in the number of instances. Note that since the total image dataset size is 1,000 images for the one-instance setting, the batch size is set to 256 for all settings. However, we assigned a better setting with a batch size of 1,024 on ImagNet-1k pre-training, except for the one-instance setting.

For ImageNet, the performance of pre-training is reduced because the number of data is reduced, but for FractalDB, the performance is improved by reducing the number of data. The performance reduction can be attributed to the augmentation method of FractalDB's data. FractalDB augments the data with Table 6 Rotation and IFS to create a 1M-scale dataset. The results in Table 6 show that Rotation does not improve the pre-training performance when using DeiT's data augmentation, and for IFS, it rather reduces the pre-training performance. Therefore, since reducing data in FractalDB corresponds to removing data augmented by IFS, we can consider that the accuracy of pre-training performance improves as the data decreases.

Number of pre-training images on 2D-OFDB (Figure 5).

Table 7: Exploratory experiment on one-instance ImageNet pre-training. Preprocessing ImageNet-1k$^{\diamondsuit}$ (1-instance) dataset with RGB images, we convert {gray, binary, canny} images for pre-training for each dataset. In this experiment, ImageNet-1k$^{\diamondsuit}$ (Canny) achieved the best accuracy in ImageNet-1k$^{\diamondsuit}$ sets {RGB, gray, binary, canny} .

Pre-training	C10	C100	IN100	P30
ImageNet-1k$^{\diamondsuit}$	94.3	76.9	84.3	78.2
– Gray scale [18]	96.1	81.2	87.8	79.9
– Binary [30]	96.7	82.7	88.8	79.9
– Canny [6]	96.5	82.8	87.7	80.3
2D-OFDB-1k	96.9	**84.0**	88.0	**80.4**
3D-OFDB-1k	**97.1**	83.8	**89.1**	80.0

We determined whether fewer pre-training images can work in ViT pre-training. Here, we decrease the number at {100, 300, 500, 700, 900} in the pre-training phase. Note that the number of parameter updates is aligned even if the number of images is decreased. And the results are then computed by averaging five times the pre-training effects of different 2D-OFDBs.Figure 5 shows that the accuracy of pre-ViT training with only 100 synthesized images was 82.7 on C100, which is much higher than the accuracy for training from scratch (57.7) and still better than that for pre-trained on ImageNet with DINO supervision (82.4).

Category selection with data pruning (Figure 6). We employed 'accidentally' found fractal categories in FDSL pre-training. However, a one-instance setting in FDSL does not require augmented image instances; that is, it makes it easier to evaluate image categories on the FDSL dataset. Therefore, we tested whether category selection can improve the pre-training effects of OFDB with the data pruning method [35].

We analyzed the tendency of selected categories from the 21k to 1k category dataset. Figure 6 shows the relationship between easy sample usage with data pruning and fine-tuning accuracy. As the figure confirms, the balanced dataset (50:50 with easy:hard samples) recorded the best accuracy among all the settings. On the other hand, a dataset mainly consisting of hard samples (10:90 or 30:70 with easy:hard samples) had lower fine-tuning accuracy than the OFDB-1k pre-trained ViT-T. See supplementary material for the samples of selected categories with data pruning.

Processing time on dataset rendering. Table 8 shows a time comparison for dataset rendering in FDSL datasets. The table indicates that the proposed one-instance FDSL datasets, 2D/3D-OFDB-1k, are $\times$6.9/10.4 faster than the rendering time with FractalDB-1k and ExFractalDB-1k, respectively. By considering the rendering time, the speed increases are $\times$41.1/135.7 faster. Note that the number of 3D points in an image is much smaller than that of 2D rendering points. Therefore, the 3D-OFDB-1k is more efficient than 2D-OFDB-1k for total time.

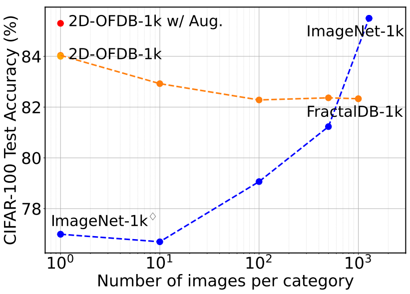


Figure 4: Relationship between accuracy and number of image instances per category.

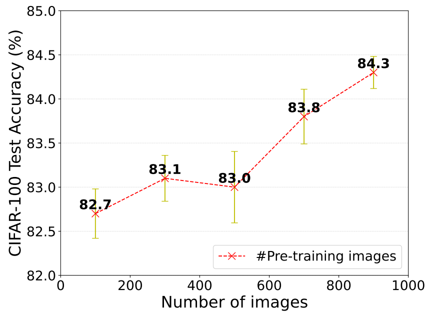


Figure 5: Relationship between number of pre-training images on OFDB and recognition accuracy.

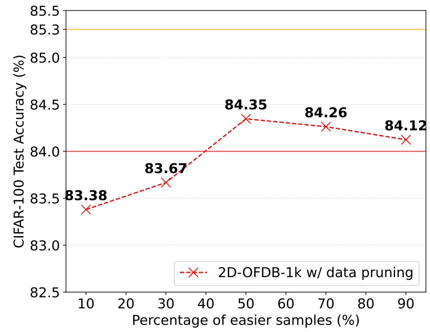


Figure 6: Data pruning for category selection on 2D-OFDB from 21k to 1k categories.

5. Discussion and Conclusion

ViT pre-training on a one-instance dataset. We validated ViT pre-training on an image dataset consisting of only one synthesized image per category. First, we succeeded in pre-training a ViT on the proposed OFDBs while still not requiring real images and human supervision. Moreover, the OFDB-1k pre-trained model performed better than the model pre-trained by FractalDB-1k (OFDB-1k 84.0 vs. FractalDB-1k 81.6; see Table 1). In further exploration, the proposed random patch augmentation with randomly selected 3×3 pixels in the fractal image rendering proved to be the most effective way to improve the OFDB pre-training effect. Consequently, we added random patch augmentation to the OFDB (w/ aug 85.3 vs. w/o 84.0; see Table 6). We also confirmed that a synthesized image dataset containing one-instance per category is sufficient to pre-train a ViT model.

Experimental comparison discussion. The proposed OFDB-1k surpassed ImageNet-1k in pre-training for all tasks, including image classification, object detection, and instance segmentation (refer to Tables 1 to 4). Notably, 2D/3D-OFDB-21k matched or exceeded ImageNet-21k's pre-training performance. With ViT-B model fine-tuning, we achieved 82.2/82.7 accuracy on ImageNet-1k (Table 2). Our pre-training utilized 21k images, compared to ImageNet-21k's 14M. The speedup between OFDB-21k and ImageNet-21k pre-training was about $\times3.3$ (2D/3D-OFDB-21k used 1,088 GPU hours vs. ImageNet-21k's 3,657 GPU hours).

In ViT pre-training with fewer images, 2D/3D-OFDB outperformed IDMM in average accuracy for benchmark fine-tuning datasets and pre-training image count (Table 3). Our method required no model modifications and fewer images, using just 1,000 synthetics versus IDMM's minimum of 2,040 reals. As shown in Figure 5, 2D-OFDB pre-training with 100 images yielded 82.7 accuracy on C100, surpassing DINO self-supervised ImageNet. This method substantially lowered image count in ViT pre-training compared to prior work.

Table 8: Datasets rendering time by full-instance (FractalDB/ExFractalDB-1k) and one-instance (2D/3D-OFDB-1k). We separately show fractal category search (Search), image rendering (Render), and total time (Total). The values are given in hours.

Dataset	Search	Render	Total
FractalDB-1k	2.37	16.86	19.23
2D-OFDB-1k	2.37	0.41	2.78
ExFractalDB-1k	0.53	5.43	5.96
3D-OFDB-1k	0.53	0.04	0.57

Limitations. By using the proposed 2D/3D-OFDB pre-trained models, the pre-training methods still have lower accuracies compared to ImageNet-1k pre-training with a full-instance scale for relatively small datasets (see Table 1). We believe that when improved image representation and teacher labels are provided, even a 1,000-image dataset will achieve a superior pre-training effect than a 1.28M-image dataset. This would be advantageous for computing resources, especially in terms of memory usage and computational time. On the other hand, the models pre-trained on 2D/3D-OFDB-21k could not achieve the performance of ExFractalDB-21k pre-trained Swin Transformer (see Table 4). Though the detection/segmentation performance rates did not decrease greatly, we intend to explore whether this is a side-effect of reducing the number of instances and whether we can overcome this problem with other better image representations.

Acknowledgement

This paper is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used. We want to thank Junichi Tsujii, Yutaka Satoh, Mariko Isoagwa, Yasufumi Kawano, Ryosuke Yamada, Kodai Nakashima for their helpful comments in research discussions.

References

[1] Connor Anderson and Ryan Farrell. Improving Fractal Pre-training. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1300–1309, 2022.

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. ViViT: A Video Vision Transformer. *CoRR*, abs/2103.15691, 2021.

[3] Yuki M. Asano, Christian Rupprecht, Andrew Zisserman, and Andrea Vedaldi. PASS: An ImageNet replacement for self-supervised pretraining without humans. In *NeurIPS Track on Datasets and Benchmarks*, 2021.

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

[5] Abeba Birhane and Vinay Uday Prabhu. Large Image Datasets: A Pyrrhic Win for Computer Vision? In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1537–1547, 2021.

[6] John Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9650–9660, October 2021.

[8] João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A Short Note on the Kinetics-700 Human Action Dataset. *CoRR*, abs/1907.06987, 2019.

[9] PeterWelinder PietroPerona CatherineWah, SteveBranson and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. (CNS-TR-2011-001), 2011.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607, 2020.

[11] Xinlei Chen, Saining Xie, and Kaiming He. An Empirical Study of Training Self-Supervised Vision Transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9640–9649, 2021.

[12] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing Textures in the Wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3606–3613, 2014.

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representation (ICLR)*, 2021.

[15] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes Challenge: A Retrospective. *International Journal of Computer Vision (IJCV)*, 111(1):98–136, 2015.

[16] 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 (CVPR)*, pages 16000–16009, June 2022.

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017.

[18] Christopher Kanan and Garrison Cottrell. Color-to-Grayscale: Does the Method Matter in Image Recognition? *PloS one*, 7:e29740, 01 2012.

[19] Hirokatsu Kataoka, Ryo Hayamizu, Ryosuke Yamada, Kodai Nakashima, Sora Takashima, Xinyu Zhang, Edgar Josafat Martinez-Noriega, Nakamasa Inoue, and Rio Yokota. Replacing Labeled Real-Image Datasets With Auto-Generated Contours. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21232–21241, June 2022.

[20] 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.

[21] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without Natural Images. volume 130, 2022.

[22] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3DRR-13)*, pages 554–561, 2013.

[23] Alex Krizhevsky and Geoffrey Hinton. Learning Multiple Layers of Features from Tiny Images. *Technical report, University of Toronto*.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25, pages 1097–1105. Curran Associates, Inc., 2012.

[25] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551, 1989.

[26] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 248–255, 2021.

[27] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. Fine-Grained Visual Classification of Aircraft. *CoRR*, abs/1306.5151, 2013.

[28] Kodai Nakashima, Hirokatsu Kataoka, Asato Matsumoto, Kenji Iwata, and Nakamasa Inoue. Can Vision Transformers Learn without Natural Images? *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2):1990–1998, 2022.

[29] Maria-Elena Nilsback and Andrew Zisserman. Automated Flower Classification over a Large Number of Classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, pages 722–729, 2008.

[30] Nobuyuki Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[31] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and Dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505, 2012.

[32] K. Perlin. Improving noise. *ACM Transactions on Graphics (TOG)*, 21(3):681–682, 2002.

[33] Ariadna Quattoni and Antonio Torralba. Recognizing Indoor Scenes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 413–420, 2009.

[34] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision Transformers for Dense Prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12179–12188, October 2021.

[35] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond Neural Scaling Laws: Beating Power Law Scaling via Data Pruning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[36] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 843–852, 2017.

[37] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning (ICML)*, pages 10347–10357, 2021.

[38] Serge Belongie Lubomir Bourdev Ross Girshick James Hays Pietro Perona Deva Ramanan C. Lawrence Zitnick Piotr Dollar Tsung-Yi Lin, Michael Maire. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

[39] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. PVTv2: Improved Baselines with Pyramid Vision Transformer. *CoRR*, abs/2106.13797, 2021.

[40] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. *CoRR*, abs/2105.15203, 2021.

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

[42] Ryosuke Yamada, Ryo Takahashi, Ryota Suzuki, Akio Nakamura, Yusuke Yoshiyasu, Ryusuke Sagawa, and Hirokatsu Kataoka. MV-FractalDB: Formula-driven Supervised Learning for Multi-view Image Recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[43] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards Fairer Datasets: Filtering and Balancing the Distribution of the People Subtree in the ImageNet Hierarchy. In *Conference on Fairness, Accountability and Transparency (FAT)*, pages 547–558, 2020.

[44] Hao Yu Yun-Hao Cao and Jianxin Wu. Training Vision Transformers with Only 2040 Images. *The European Conference on Computer Vision (ECCV)*, pages 220–237, 2022.

[45] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(6):1452–1464, 2017.