Pre-training without Natural Images

Hirokatsu Kataoka

AIST

http://www.hirokatsukataoka.net/

Hirokatsu Kataoka

Chief Senior Researcher, Computer Vision Research Team, AIST

Profile：

- Ph.D. in Engineering at Keio University (Mar 2014）
- Chief Senior Researcher, AIST（Apr 2023 - Present）
- PI, cvpaper.challenge (May 2015 – Present; Research community with 1,000+ collaborators)
- Adjunct Researcher, LY Corp. (Oct 2023 - Present)
- Researcher, TICO-AIST Advanced Logistics Lab.（Oct 2016 - Present）
- Researcher, Tokyo Denki University（Apr 2016 - Present）
- Mentor, Tatsujin Program（Nov 2020 - Present）
- Editor, Computer Vision Frontier（Dec 2021 - Present）

片（かた）岡（おか）裕（ひろ）雄（かつ）

Recently Selected Projects (within 2 years)：

“Pre-training Vision Transformers with Very Limited Synthesized Images (ICCV23)”
“SegRCDB: Semantic Segmentation via Formula-Driven Supervised Learning (ICCV23)”
“Visual Atoms: Pre-training Vision Transformers with Sinusoidal Waves (CVPR23)”
”Replacing Labeled Real-Image Datasets with Auto-Generated Contours (CVPR22)”
“Point Cloud Pre-training with Natural 3D Structures (CVPR22)”
“Pre-training without Natural Images (IJCV22)”
“Can Vision Transformers Learn without Natural Images? (AAAI22)”

http:// hirokatsukataoka.net/

Hirokatsu Kataoka

Chief Senior Researcher, Computer Vision Research Team, AIST

1 Pre-training without Natural Images

Representation learning from a natural law

- ACCV 2020 Best Paper Honorable Mention Award
- Accepted to IJCV'22 CVPR'22 '23, AAAI'22, ICCV'23, BMVC'23 Oral
- MIT Technology Review (Feb. 4th, 2021)
- AIST Best Paper 2022

片（かた）岡（おか）裕（ひろ）雄（かつ）

2 Spatiotemporal 3D ResNet

Strong baseline for 3D convolution in video understanding

- Accepted to CVPR'18（1.9k+ citations; Top 0.5% in 8k+ 5-year CVPR papers)
- AIST Best Paper 2019
- GitHub 3.0k Stars (Top-1 in video recognition at the time of published)

http:// hirokatsukataoka.net/

Pre-training without Natural Images

ACCV 2020 Best Paper Honorable Mention Award
International Journal of Computer Vision (IJCV), 2022
AAAI 2022

Hirokatsu Kataoka

AIST

http://www.hirokatsukataoka.net/

What has the DNNs brought?

Benefits

- Solving various AI tasks, e.g., vision, language, audio, are widely recognized

Challenges in DNN research

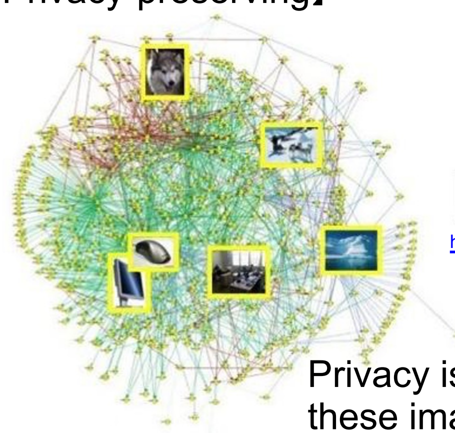
- Annotation labor
- Privacy-preserving on the Internet photos

【Large amount of annotation】

http://image-net.org/explore?wnid=n01503061

Takes 2 years, around 50k participants on AMT
14M images across 21k categories

【Privacy-preserving】



IMAGENET
http://www.image-net.org/

Privacy is a concern, limiting the use of these images to academic/educational purposes

Issues of annotation & privacy pose significant challenges for AI applications

Ethical issues in image datasets for CV

Fairness and transparency have arisen

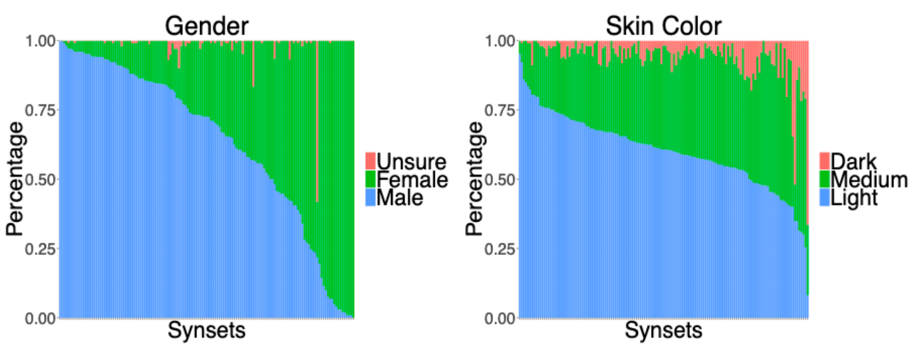
– Offensive labels, dataset bias, transparency

【Offensive labels】

- 80M Tiny Images had offensive labels
- The dataset was suspended from public access due to the difficulty of labeling and resolution https://groups.csail.mit.edu/vision/TinyImages/

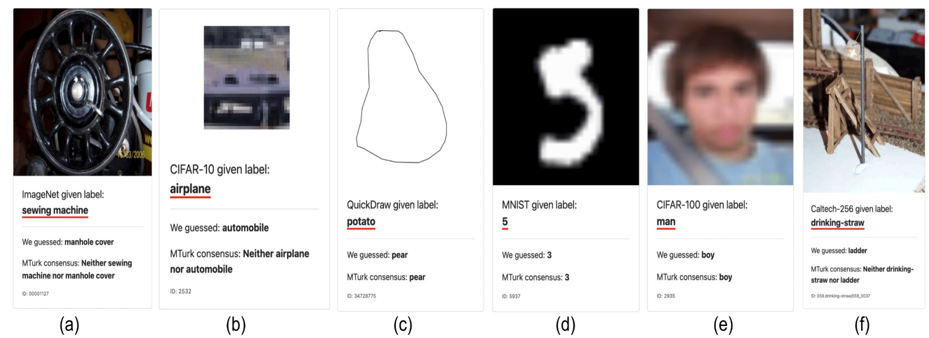
【Dataset bias】

Widely used ImageNet also faces fairness, there includes biased distributions in terms of gender/race depending on the category



https://arxiv.org/pdf/1912.07726.pdf

【Transparency】



Est. 6% label errors are included on ImageNet

C. G. Northcutt, et al. "Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks" https://arxiv.org/pdf/2103.14749.pdf

AI community recognizes ethical issues

Huge-scale datasets

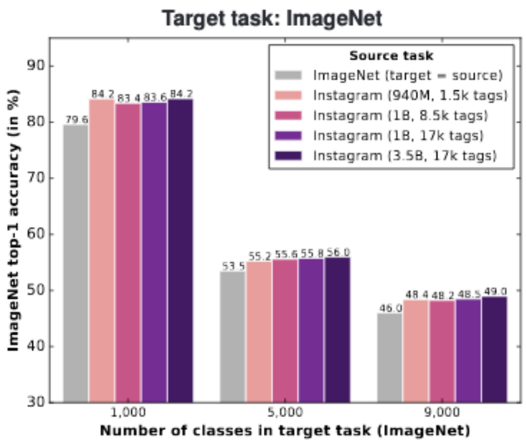
JFT-300M (Google, 2017/2021) / IG-3.5B (Meta, 2018)

300M images / 375M labels

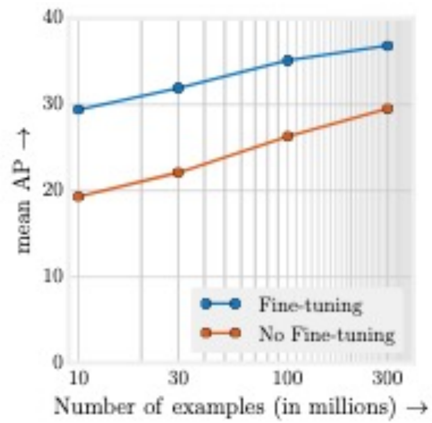
3.5B images / 3.5B weak labels

These datasets are x100 larger than ImageNet, improve image representation and recognition performance

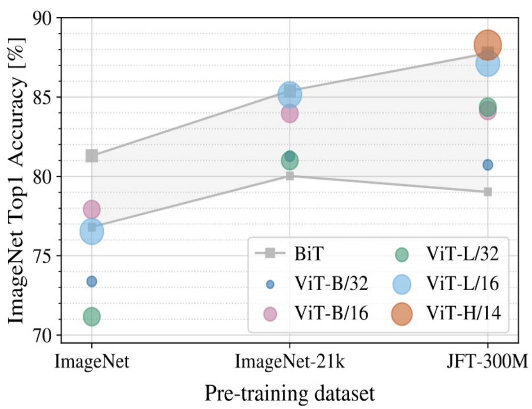
-> large-scale datasets benefits both CNN and ViT in pre-training



Meta (IG-3.5B), ECCV 2018
https://arxiv.org/pdf/1805.00932.pdf



Google (JFT-300M), ICCV 2017
http://openaccess.thecvf.com/content_ICCV_2017/papers/Sun_Revisiting_Unreasonable_Effectiveness_ICCV_2017_paper.pdf



Google (JFT-300M / ViT), ICLR 2021
https://arxiv.org/pdf/2010.11929.pdf

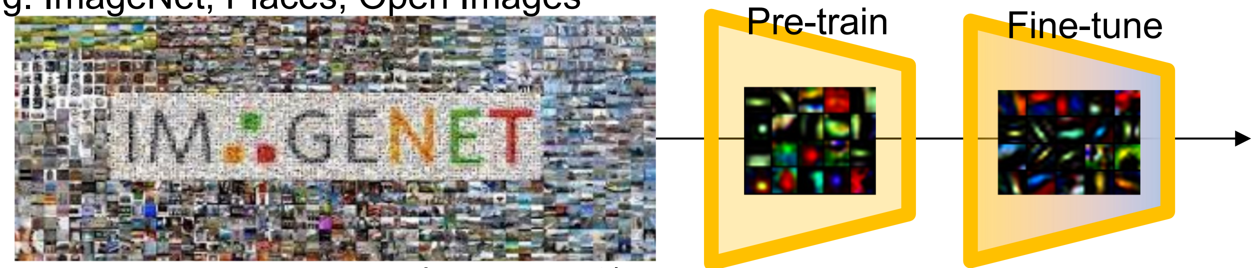
Drawback of private datasets within an organization may limit the research community

Recent vision-driven learning

Supervised Learning

remains the most promising framework, providing pre-trained models serve as good features

e.g. ImageNet, Places, Open Images



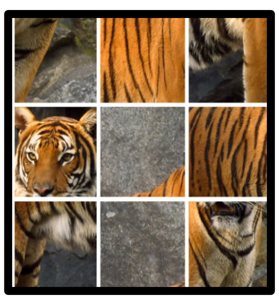
gluon-cv.mxnet.io

ImageNet + ResNet-50
76% @ImageNet val.

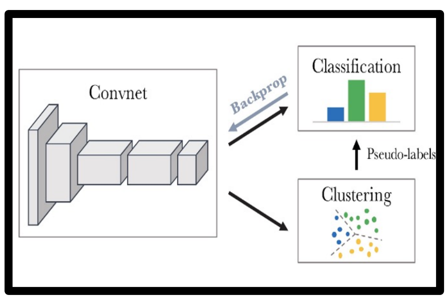
[He et al. CVPR16]

Self-supervised Learning (SSL)

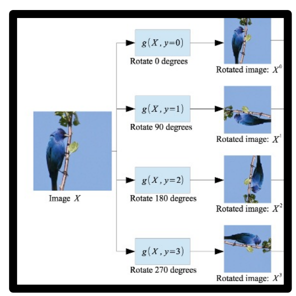
uses visual labels to create a pre-trained model in a cost-efficient way



Jigsaw Puzzle
[Noroozi al. ECCV16]



DeepCluster
[Caronet al. ECCV18]



Rotation Classify
[Gidaris et al. ICLR18]

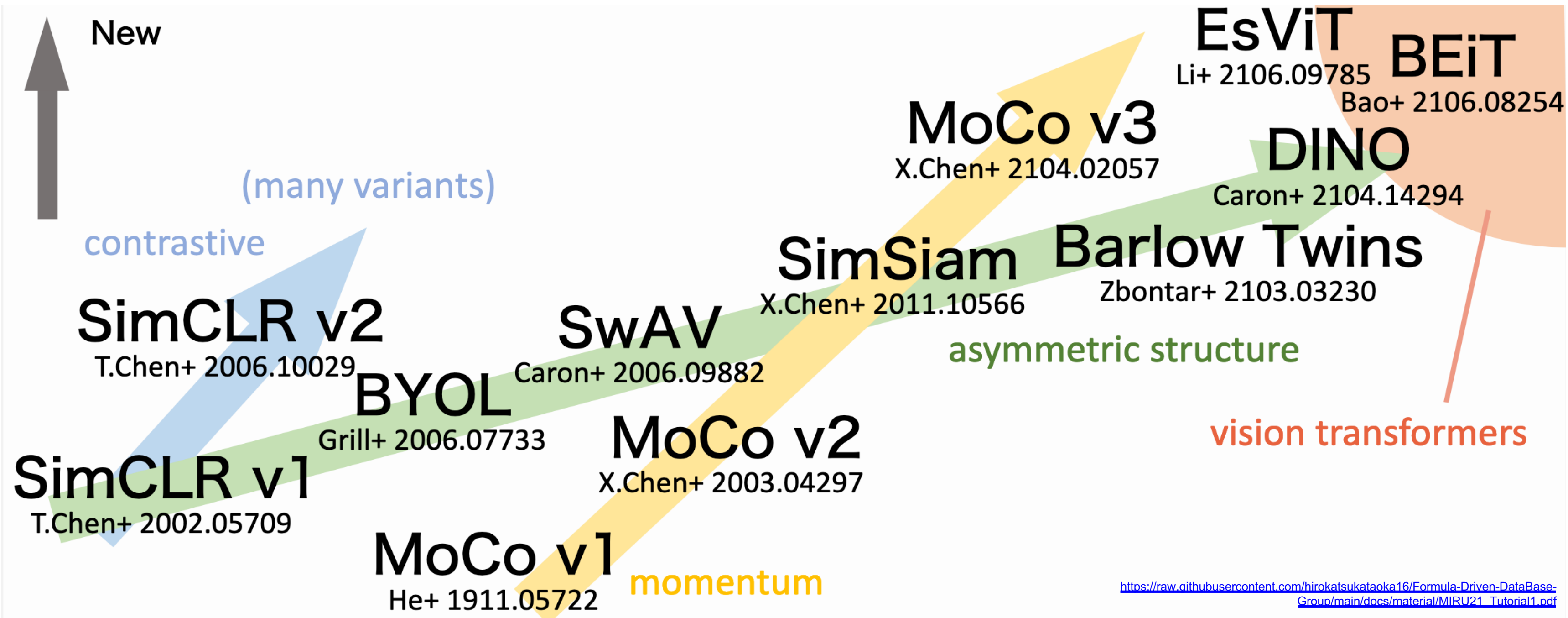
SimCLR + ResNet-50
69% @ImageNet val.

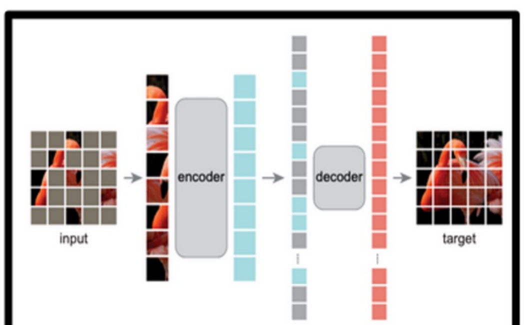
[Chen et al. ICML20]

Existing the problems of image downloading and privacy-violations

Overview of self-supervised learning

SSL is approaching the performance of SL, particularly w/ ImageNet pre-train





Masked AutoEncoder (MAE) masks parts of an image and reconstructs them to learn visual representations

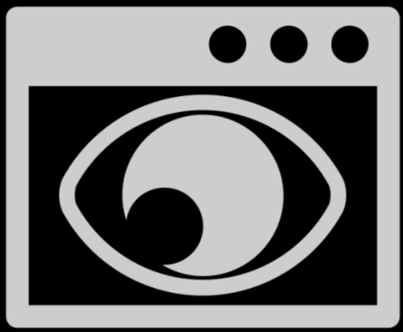
Ethical problems can occur as long as we use real images

To overcome the problems, it is better to automatically create datasets without any natural images

Annotation

FATE

Fairness, Accountability, Transparency and Ethics



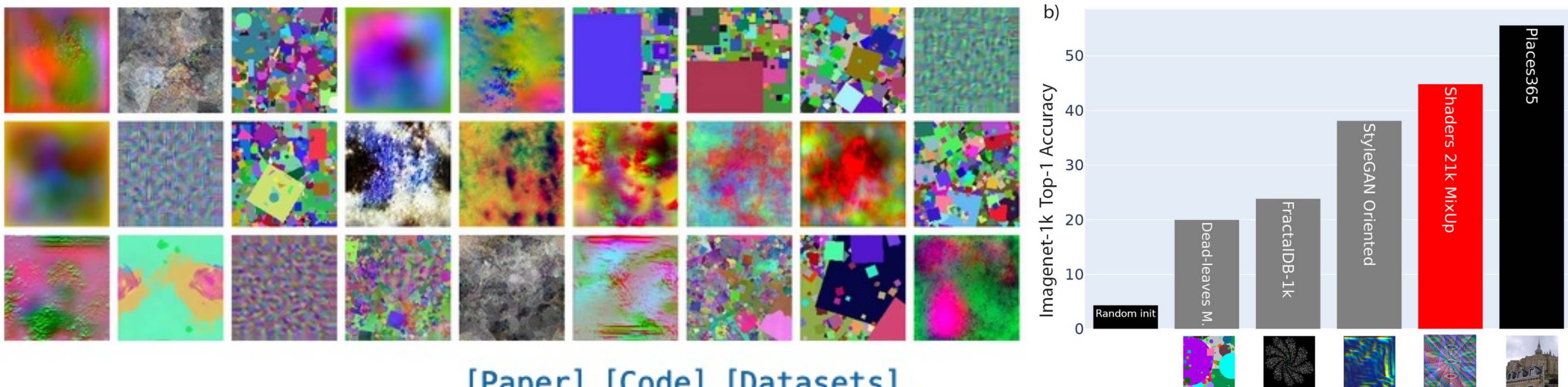
Privacy

Can we pre-train DNN without any natural images?

Two related works:

Learning to see looking at noise / shaders (MIT Torralba Lab.)

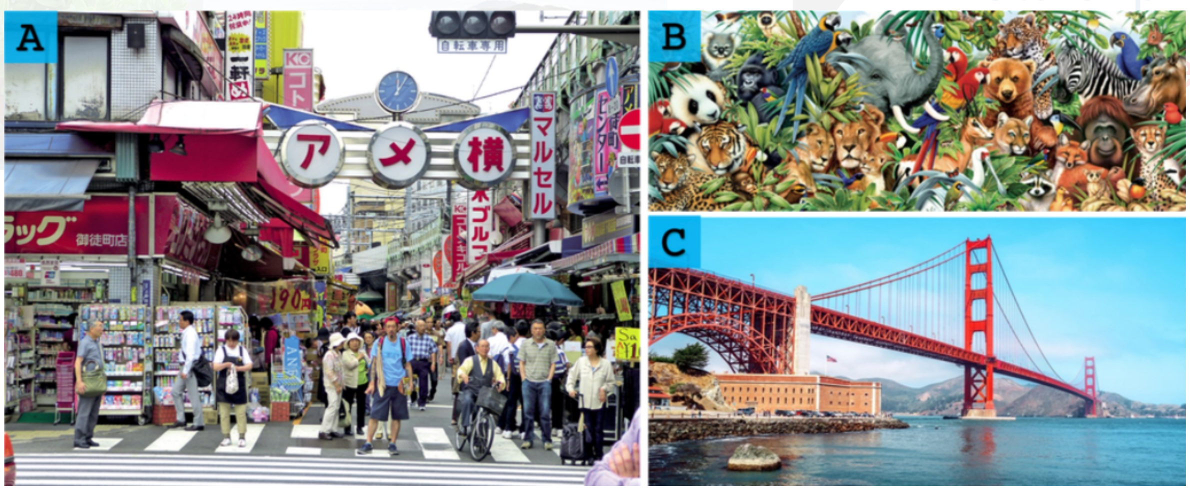
https://mbaradad.github.io/learning_with_noise/

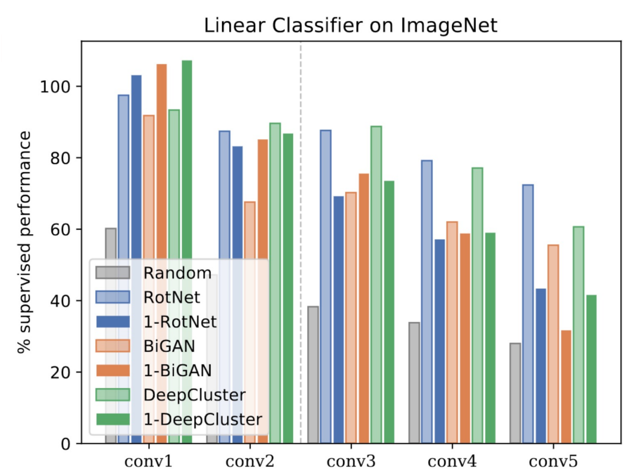


[Paper] [Code] [Datasets]

A critical analysis of self-supervision, or what we can learn from a single image (Oxford VGG)

https://arxiv.org/abs/1904.13132





Can we pre-train DNN without any natural images?

Formula-driven Supervised Learning (FDSL)

- Generate image patterns and their labels
- Using mathematical formulas and/or functions

Observed fractal geometry on ImageNet dataset

We hypothesize DNN could learn natural principles from ImageNet?

Directly render and train Fractals

Our goal is to find a way to pre-train
without any real images and human labels

Proposed method: FractalDB Pre-trained CNN

Formula-Driven Supervised Learning (FDSL)

1) to make pre-trained CNN from a mathematical formula
2) without relying on human/self-supervision & natural images

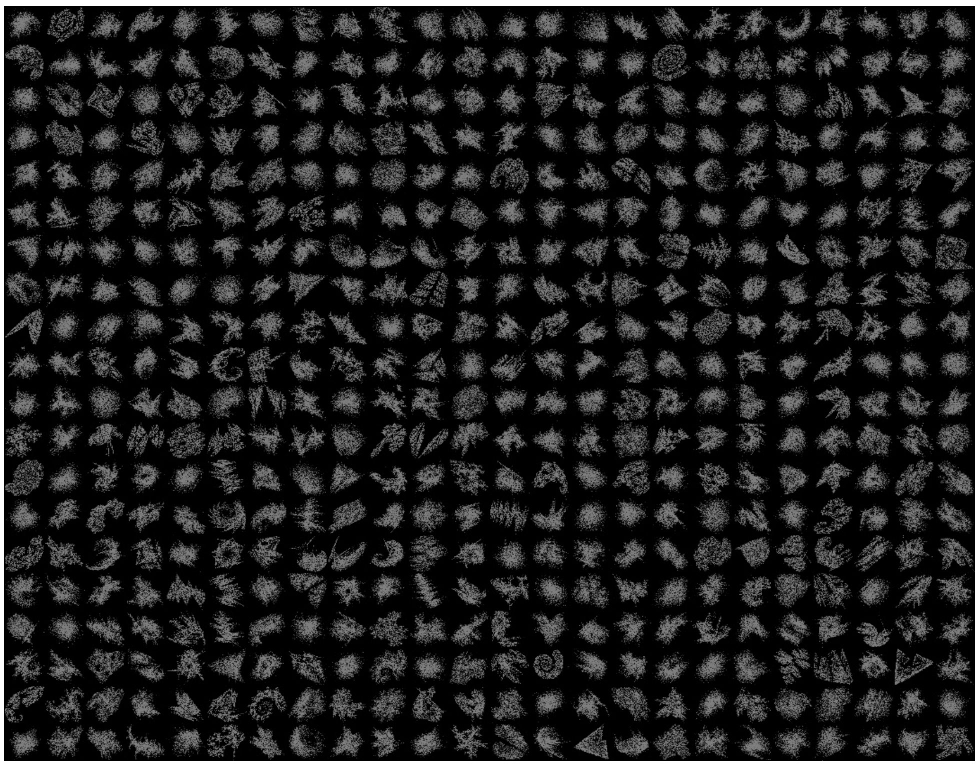
Results comparable to real images & human supervision

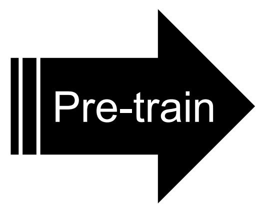
FractalDB

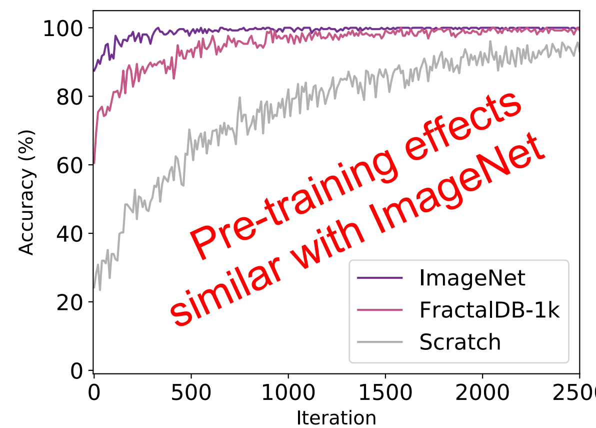
1) to make a pre-trained CNN without any natural images

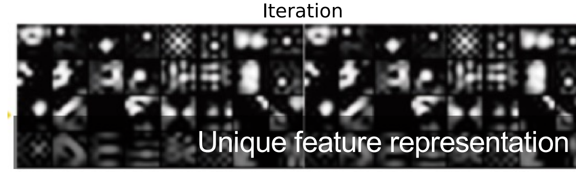
2) for a concept of Formula-driven Supervised Learning

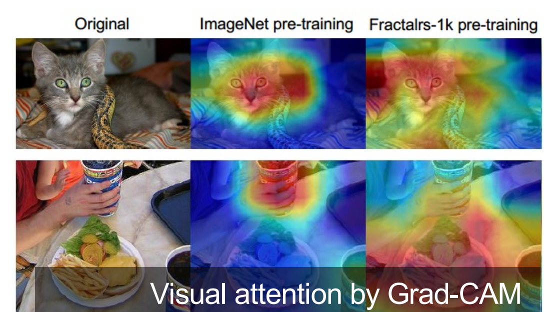
Ability to effectively train models based on natural laws

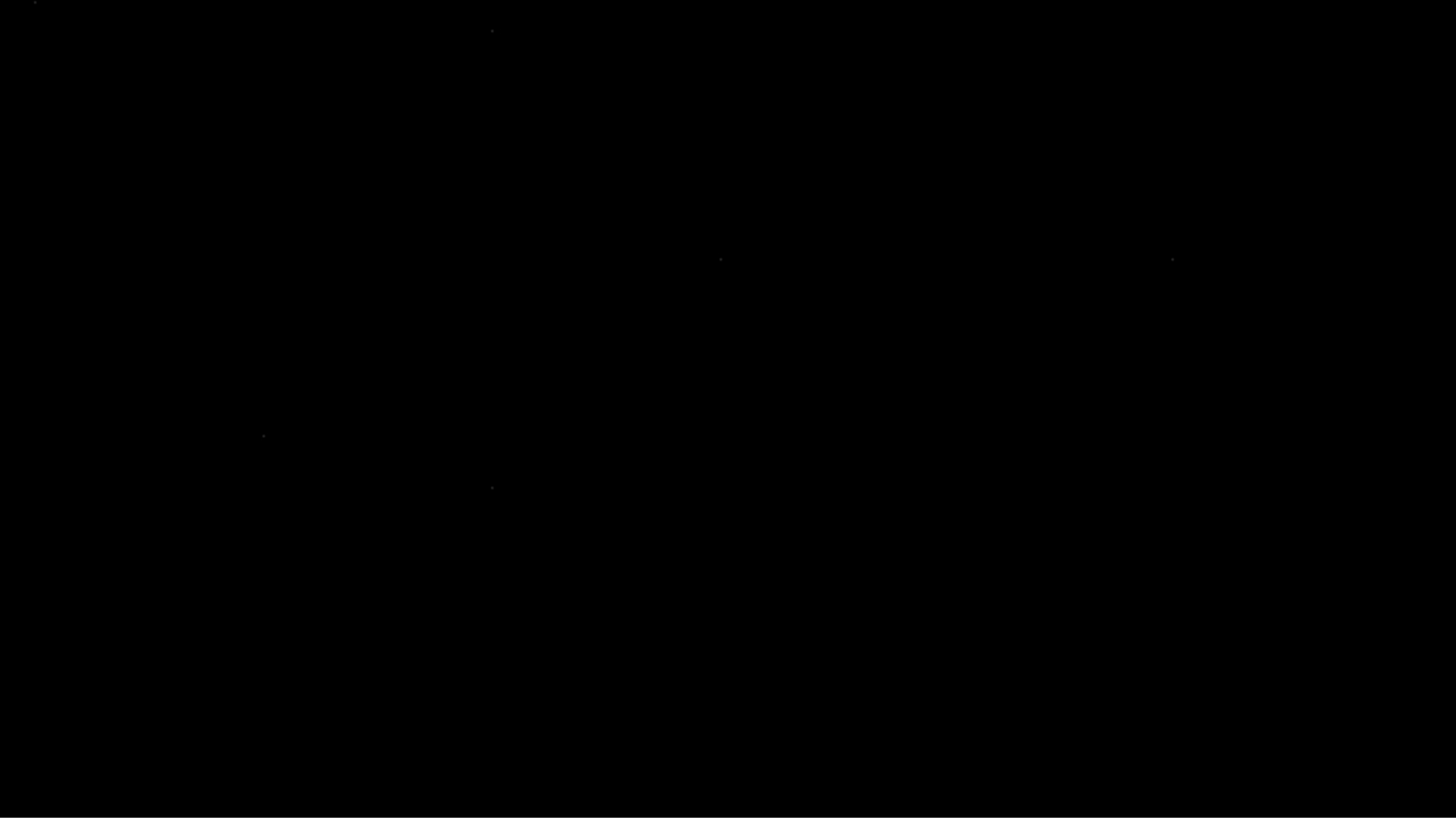












$$\mathrm{IFS} = \{\mathcal{X}; w_1, w_2, \cdots, w_N; p_1, p_2, \cdots, p_N\}$$ # Transformation probability

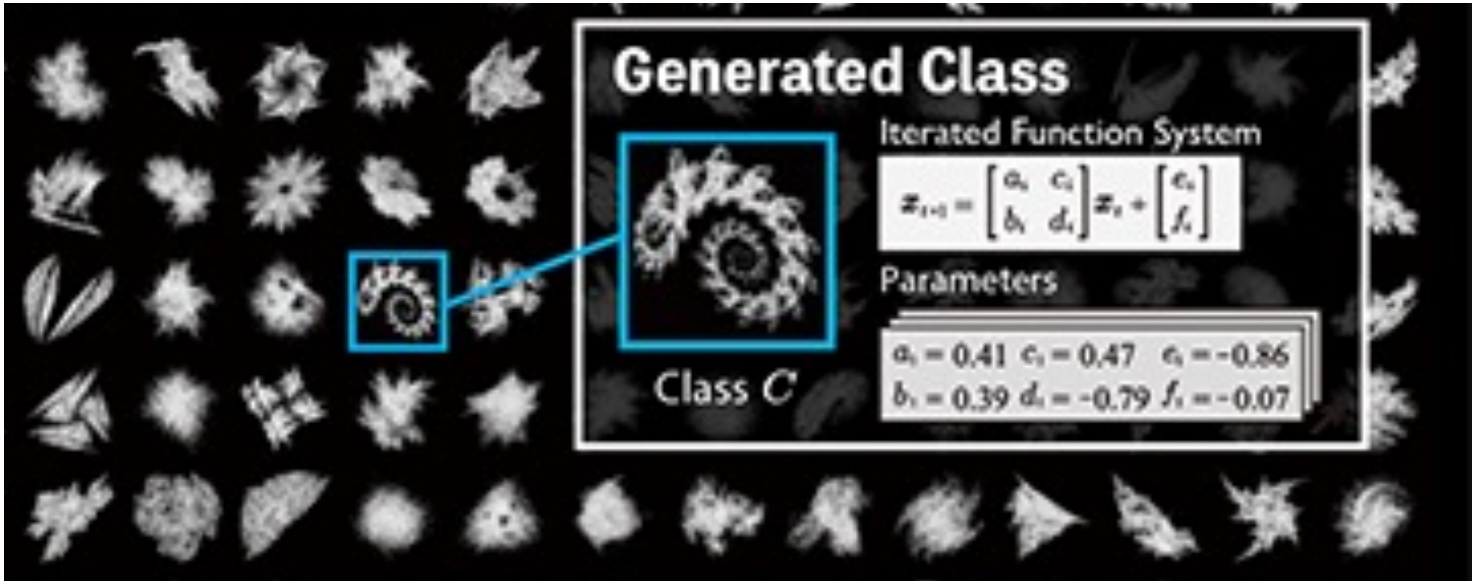
$$w_i(\boldsymbol{x}; \theta_i) = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} e_i \\ f_i \end{bmatrix}$$ # Affine transformation

Iteratively renders a large number of dots or patches in an image

Search for fractal categories

Randomly select parameters to render

1. Fractal image rendering with randomized params *a*～*f*, *w* w/ IFS
2. If the filling rate (> *r*), the fractal category is added to DB
3. Repeated up to defined #category (*C*)
   - Parameter separation makes a different fractal category

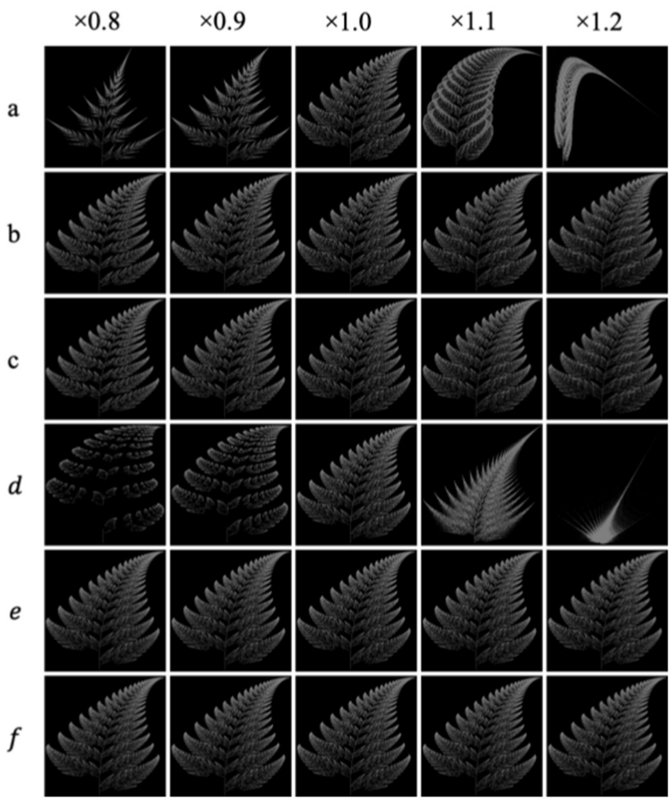


Fractal categories on FractalDB

Instance augmentation in each category

Three different augmentation methods

1. Parameter set variations (x25)
2. Image rotation (x4)
3. Patch pattern (x10)



Parameter set (x25)

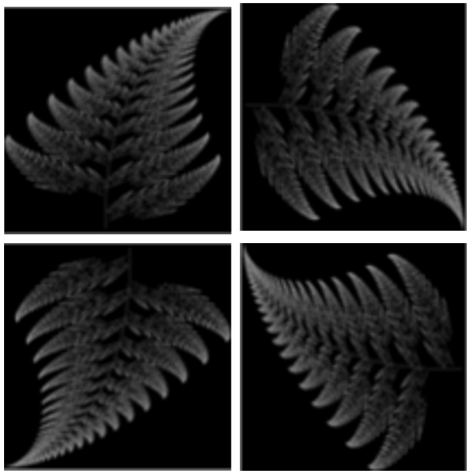
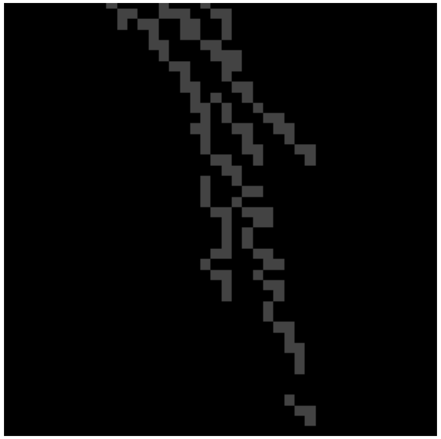


Image rotation (x4)



Patch pattern (x10)

Select 10 rando 3x3 patch patterns out of 256 (2^8)

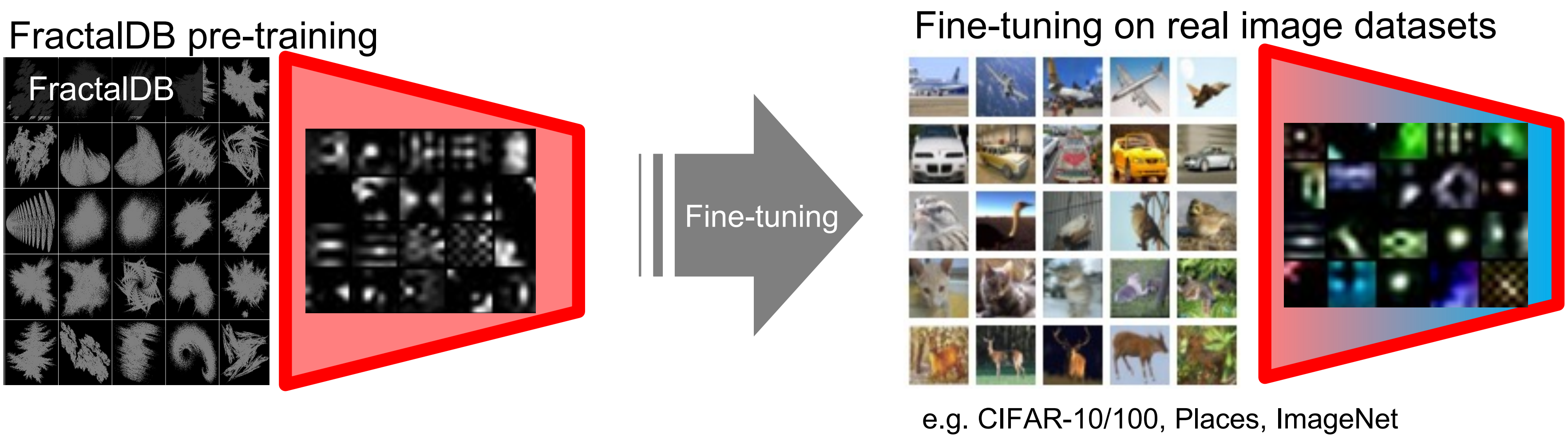
Up to x1000 instances per category

Experimental setting

Pre-training & Fine-tuning

– Pre-training done without using any real images

– Fine-tuning in a traditional manner

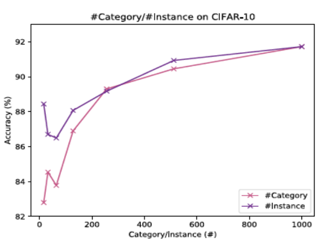


Parameter tunings on FractalDB pre-trained CNN

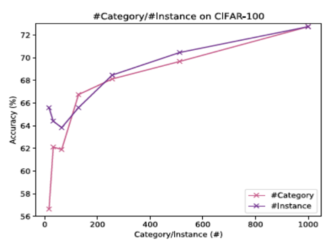
Through the exploration study, our findings that:

– #Category, #instance, and patch-rendering are the most effective parameters on the pre-training phase

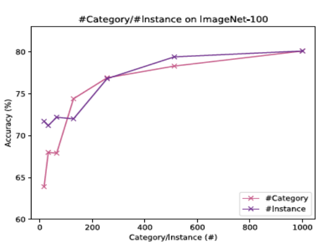
– A more difficult pre-train is slightly better in weights



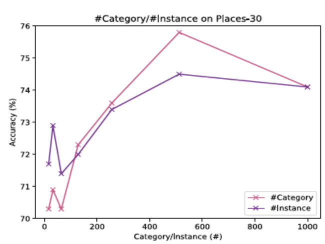
(a) CIFAR10



(b) CIFAR100



(c) ImageNet100



(d) Places30

Table 1. Patch vs. point.

	C10	C100	IN100	P30
Point	87.4	66.1	73.9	73.0
Patch (random)	**92.1**	**72.0**	**78.9**	**73.2**
Patch (fix)	**92.9**	**73.6**	**80.0**	**75.0**

Table 2. Filling rate.

	C10	C100	IN100	P30
.05	91.8	**72.4**	80.2	74.6
.10	**92.0**	72.3	**80.5**	**75.5**
.15	91.7	71.6	80.2	74.3
.20	91.3	70.8	78.8	74.7
.25	91.1	63.2	72.4	74.1

Table 3. Weights.

	C10	C100	IN100	P30
.1	92.1	72.0	78.9	73.2
.2	92.4	72.7	79.2	73.9
.3	92.4	72.6	79.2	74.3
.4	**92.7**	**73.1**	**79.6**	**74.9**
.5	91.8	72.1	78.9	73.5

Table 4. #Dot.

	C10	C100	IN100	P30
100k	**91.3**	70.8	78.8	74.7
200k	90.9	**71.0**	79.2	**74.8**
400k	90.4	70.3	**80.0**	74.5

Table 5. Image size.

	C10	C100	IN100	P30
256	**92.9**	**73.6**	80.0	75.0
362	92.2	73.2	**80.5**	**75.1**
512	90.9	71.0	79.2	73.0
724	90.8	71.0	79.2	73.0
1024	89.6	68.6	77.5	71.9

Please refer to our main paper for more details

Results (1/5)

Experimental comparisons on SL, SSL, and FDSL

Method	Pre-train Img	Type	C10	C100	IN1k	P365	VOC12	OG
Scratch	–	–	87.6	62.7	<u>**76.1**</u>	49.9	58.9	1.1
DC-10k	Natural	Self-supervision	89.9	66.9	66.2	<u>**51.5**</u>	67.5	15.2
Places-30	Natural	Supervision	90.1	67.8	69.1	–	69.5	6.4
Places-365	Natural	Supervision	**94.2**	76.9	71.4	–	**78.6**	10.5
ImageNet-100	Natural	Supervision	91.3	70.6	–	49.7	72.0	12.3
ImageNet-1k	Natural	Supervision	<u>**96.8**</u>	<u>**84.6**</u>	–	50.3	<u>**85.8**</u>	17.5
FractalDB-1k	Formula	Formula-supervision	93.4	75.7	70.3	49.5	58.9	**20.9**
FractalDB-10k	Formula	Formula-supervision	94.1	**77.3**	**71.5**	**50.8**	73.6	<u>**29.2**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

Results (1/5)

Comparison between training from scratch and proposed methods

Method	Pre-train Img	Type	C10	C100	IN1k	P365	VOC12	OG
Scratch	–	–	87.6	62.7	<u>**76.1**</u>	49.9	58.9	1.1
DC-10k	Natural	Self-supervision	89.9	66.9	66.2	<u>**51.5**</u>	67.5	15.2
Places-30	Natural	Supervision	90.1	67.8	69.1	–	69.5	6.4
Places-365	Natural	Supervision	**94.2**	76.9	71.4	–	**78.6**	10.5
ImageNet-100	Natural	Supervision	91.3	70.6	–	49.7	72.0	12.3
ImageNet-1k	Natural	Supervision	<u>**96.8**</u>	<u>**84.6**</u>	–	50.3	<u>**85.8**</u>	17.5
FractalDB-1k	Formula	Formula-supervision	93.4	75.7	70.3	49.5	58.9	**20.9**
FractalDB-10k	Formula	Formula-supervision	94.1	**77.3**	**71.5**	**50.8**	73.6	<u>**29.2**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

FractalDB pre-trained model achieved much higher rates than training from scratch

Results (1/5)

Comparison between SSL and proposed methods

Method	Pre-train Img	Type	C10	C100	IN1k	P365	VOC12	OG
Scratch	–	–	87.6	62.7	**76.1**	49.9	58.9	1.1
DC-10k	Natural	Self-supervision	89.9	66.9	66.2	**51.5**	67.5	15.2
Places-30	Natural	Supervision	90.1	67.8	69.1	–	69.5	6.4
Places-365	Natural	Supervision	**94.2**	76.9	71.4	–	**78.6**	10.5
ImageNet-100	Natural	Supervision	91.3	70.6	–	49.7	72.0	12.3
ImageNet-1k	Natural	Supervision	**96.8**	**84.6**	–	50.3	**85.8**	17.5
FractalDB-1k	Formula	Formula-supervision	93.4	75.7	70.3	49.5	58.9	**20.9**
FractalDB-10k	Formula	Formula-supervision	94.1	**77.3**	**71.5**	**50.8**	73.6	**29.2**

Underlined bold: best score, **Bold**: second best score

In the most cases, our method surpasses DeepCluster with 10k categories

Results (1/5)

Comparison between SL with 100k-order datasets and proposed methods

Method	Pre-train Img	Type	C10	C100	IN1k	P365	VOC12	OG
Scratch	–	–	87.6	62.7	<u>**76.1**</u>	49.9	58.9	1.1
DC-10k	Natural	Self-supervision	89.9	66.9	66.2	<u>**51.5**</u>	67.5	15.2
Places-30	Natural	Supervision	90.1	67.8	69.1	–	69.5	6.4
Places-365	Natural	Supervision	**94.2**	76.9	71.4	–	**78.6**	10.5
ImageNet-100	Natural	Supervision	91.3	70.6	–	49.7	72.0	12.3
ImageNet-1k	Natural	Supervision	<u>**96.8**</u>	<u>**84.6**</u>	–	50.3	<u>**85.8**</u>	17.5
FractalDB-1k	Formula	Formula-supervision	93.4	75.7	70.3	49.5	58.9	**20.9**
FractalDB-10k	Formula	Formula-supervision	94.1	**77.3**	**71.5**	**50.8**	73.6	<u>**29.2**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

The FractalDB pre-trained model is still better than 100k-order supervised datasets

Results (1/5)

Comparison between SL with 1M-order datasets and proposed methods

Method	Pre-train Img	Type	C10	C100	IN1k	P365	VOC12	OG
Scratch	–	–	87.6	62.7	<u>**76.1**</u>	49.9	58.9	1.1
DC-10k	Natural	Self-supervision	89.9	66.9	66.2	<u>**51.5**</u>	67.5	15.2
Places-30	Natural	Supervision	90.1	67.8	69.1	–	69.5	6.4
Places-365	Natural	Supervision	**94.2**	76.9	71.4	–	**78.6**	10.5
ImageNet-100	Natural	Supervision	91.3	70.6	–	49.7	72.0	12.3
ImageNet-1k	Natural	Supervision	<u>**96.8**</u>	<u>**84.6**</u>	–	50.3	<u>**85.8**</u>	17.5
FractalDB-1k	Formula	Formula-supervision	93.4	75.7	70.3	49.5	58.9	**20.9**
FractalDB-10k	Formula	Formula-supervision	94.1	**77.3**	**71.5**	**50.8**	73.6	<u>**29.2**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

Our method partially surpasses the ImageNet/Places pre-trained models

Results (2/5)

Auto-generated label and use of real images in DeepCluster and Fractal images

Mtd	PT Img	C10	C100	IN1k	P365	VOC12	OG
DC-10k	Natural	89.9	66.9	66.2	51.2	67.5	15.2
DC-10k	Formula	83.1	57.0	65.3	**53.4**	60.4	15.3
F1k	Formula	93.4	75.7	70.3	49.5	58.9	20.9
F10k	Formula	**94.1**	**77.3**	**71.5**	50.8	**73.6**	**29.2**

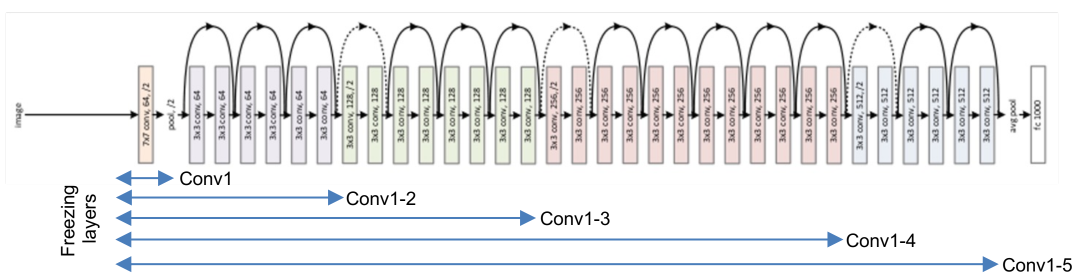
Bold: best score

Our results suggest that self-supervision alone is not enough to effectively pre-train for recognizing real images, this shows our method assigns an appropriate image pattern and the category

Results (3/5)

Evaluation of frozen conv layers

Freezing layer(s)	C10	C100	IN100	P30
Fine-tuning	93.4	75.7	82.7	75.9
Conv1	92.3	72.2	77.9	74.3
Conv1–2	92.0	72.0	77.5	72.9
Conv1–3	89.3	68.0	71.0	68.5
Conv1–4	82.7	56.2	55.0	58.3
Conv1–5	49.4	24.7	21.2	31.4



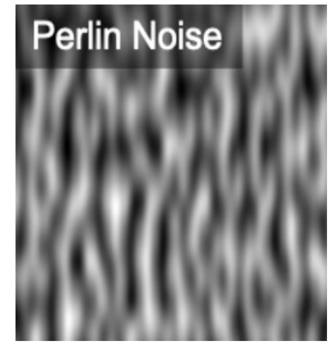
Full fine-tuning resulted the best score

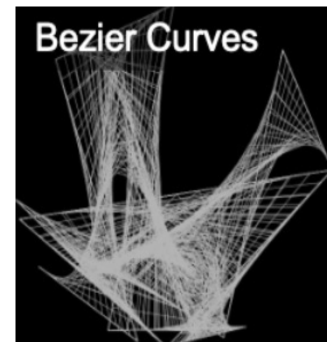
Moreover, earlier layers tend to be good feature representations

Results (4/5)

Compared to Perlin noise and Bezier curves

Pre-training	C10	C100	IN100	P30
Scratch	87.6	60.6	75.3	70.3
Bezier-144	87.6	62.5	72.7	73.5
Bezier-1024	89.7	68.1	73.0	73.6
Perlin-100	90.9	70.2	73.0	73.3
Perlin-1296	90.4	71.1	79.7	74.2
FractalDB-1k	**93.4**	**75.7**	**82.7**	**75.9**



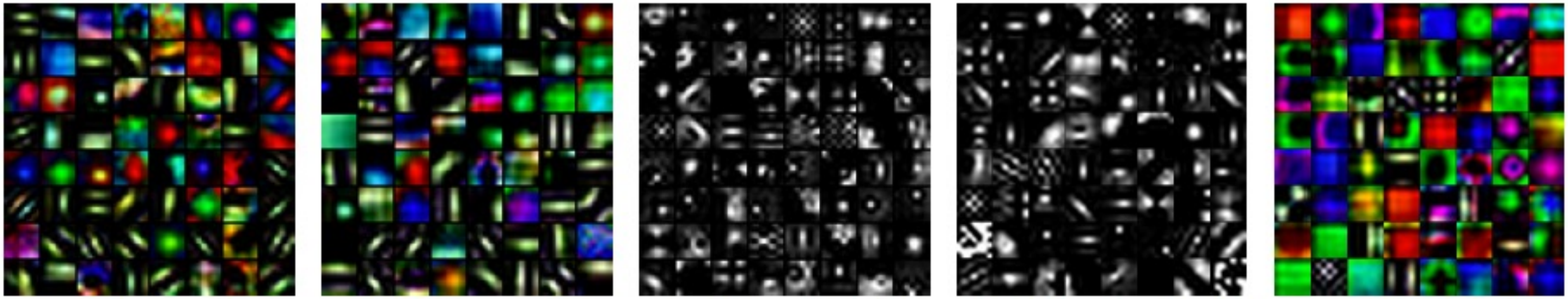


We compare Formula-driven Supervised Learning with other principles

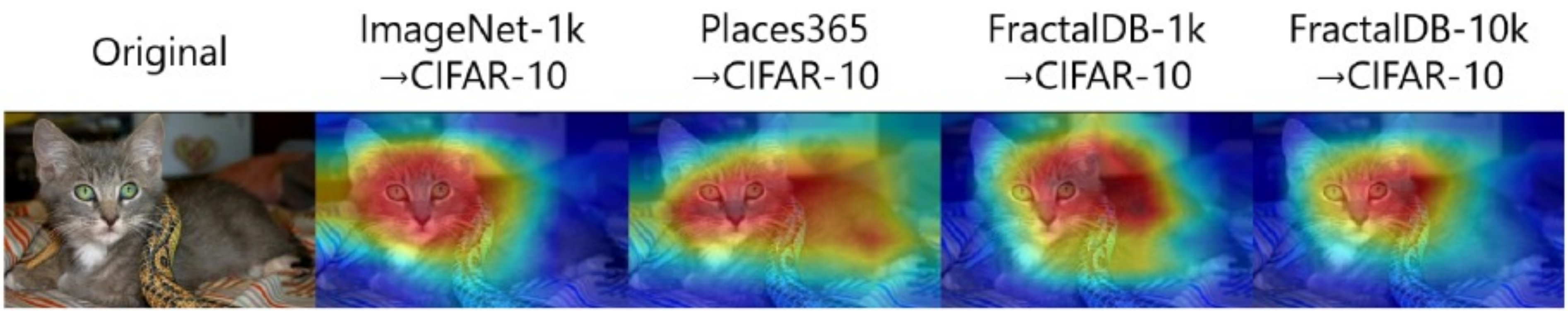
The FractalDB pre-training expected to improve from other methods

Results (5/5)

Visualization of Conv1



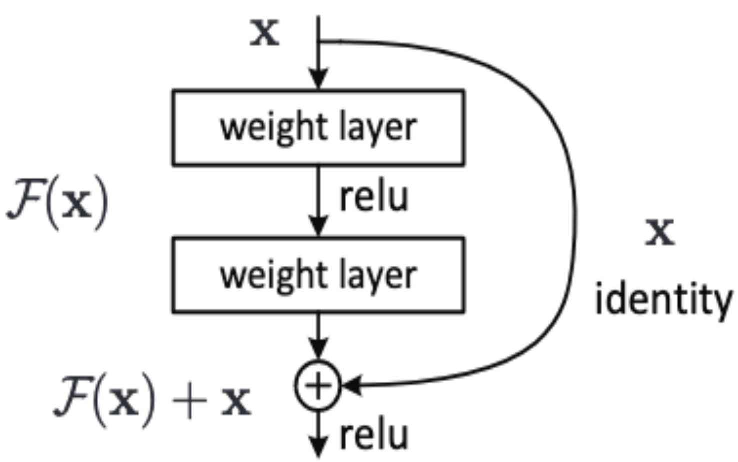
(a) ImageNet (b) Places365 (c) Fractal-1K (d) Fractal-10K (e) DC-10k



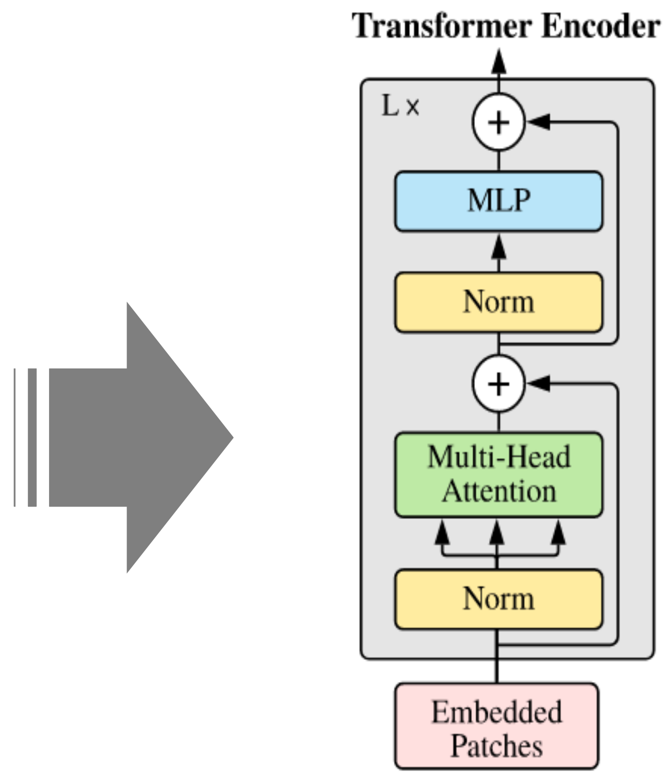
FractalDB pre-training acquires different representations, yet focuses on similar areas

Paradigm Shift in Computer Vision

‘Convolution’ to ‘Self-attention’



[He al. CVPR16]



[Vaswani al. NIPS17]
Figure from [Dosovitskiy al. ICLR21]

Computer vision researchers are now exploring ways to replace convolutional layers with Transformer encoders

Can Vision Transformers Learn without Natural Images?

AAAI 2022

Hirokatsu Kataoka

AIST

http://www.hirokatsukataoka.net/

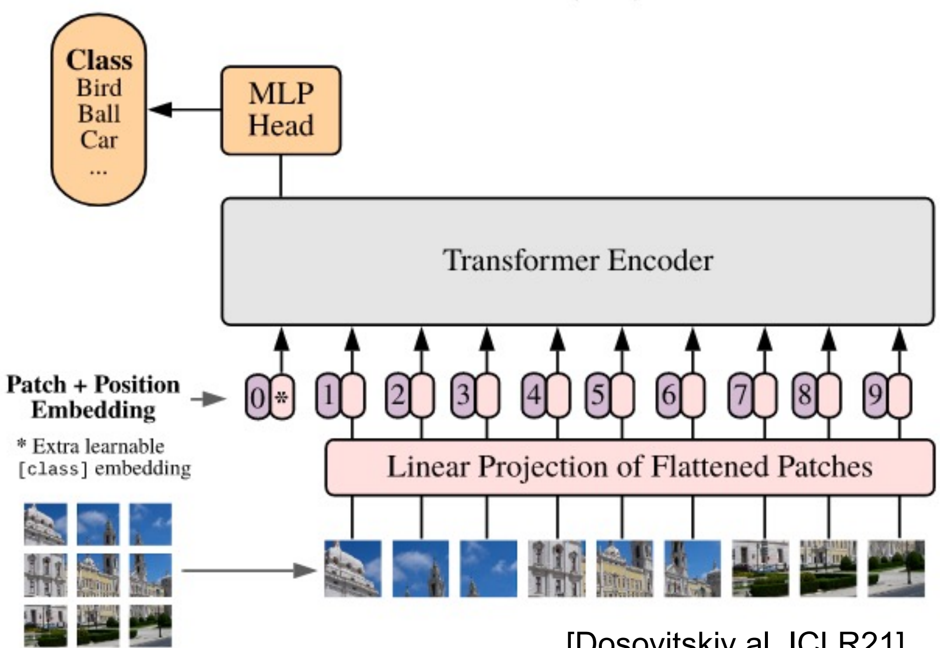
Vision Transformer (ViT), so far

One more shift in Transformer

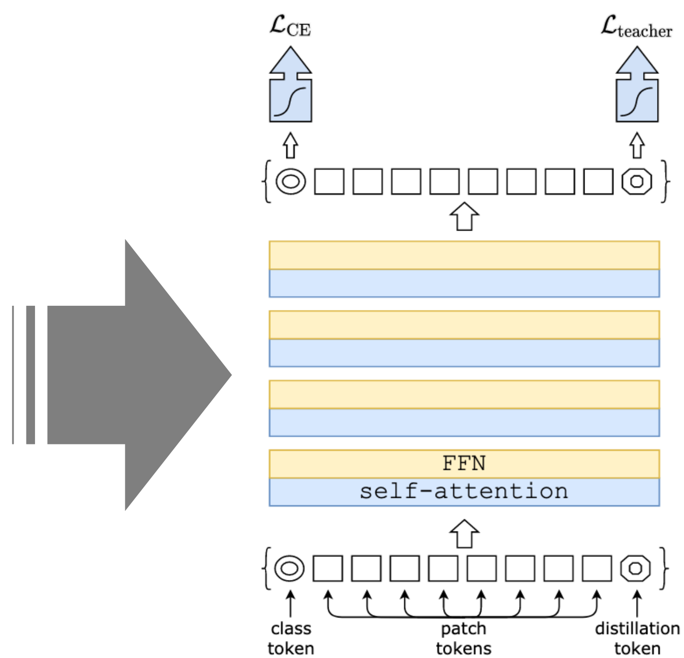
– ViT to DeiT (Data-efficient image Transformer)

– JFT-300M to ImageNet-1k in pre-training

Can ViT learn without real images?



[Dosovitskiy al. ICLR21]



[Touvron al. arXiv20]

Experimental setting

Architecture

– ViT

- No difference from the original vision transformer
- We assign richer data augmentation proposed in DeiT

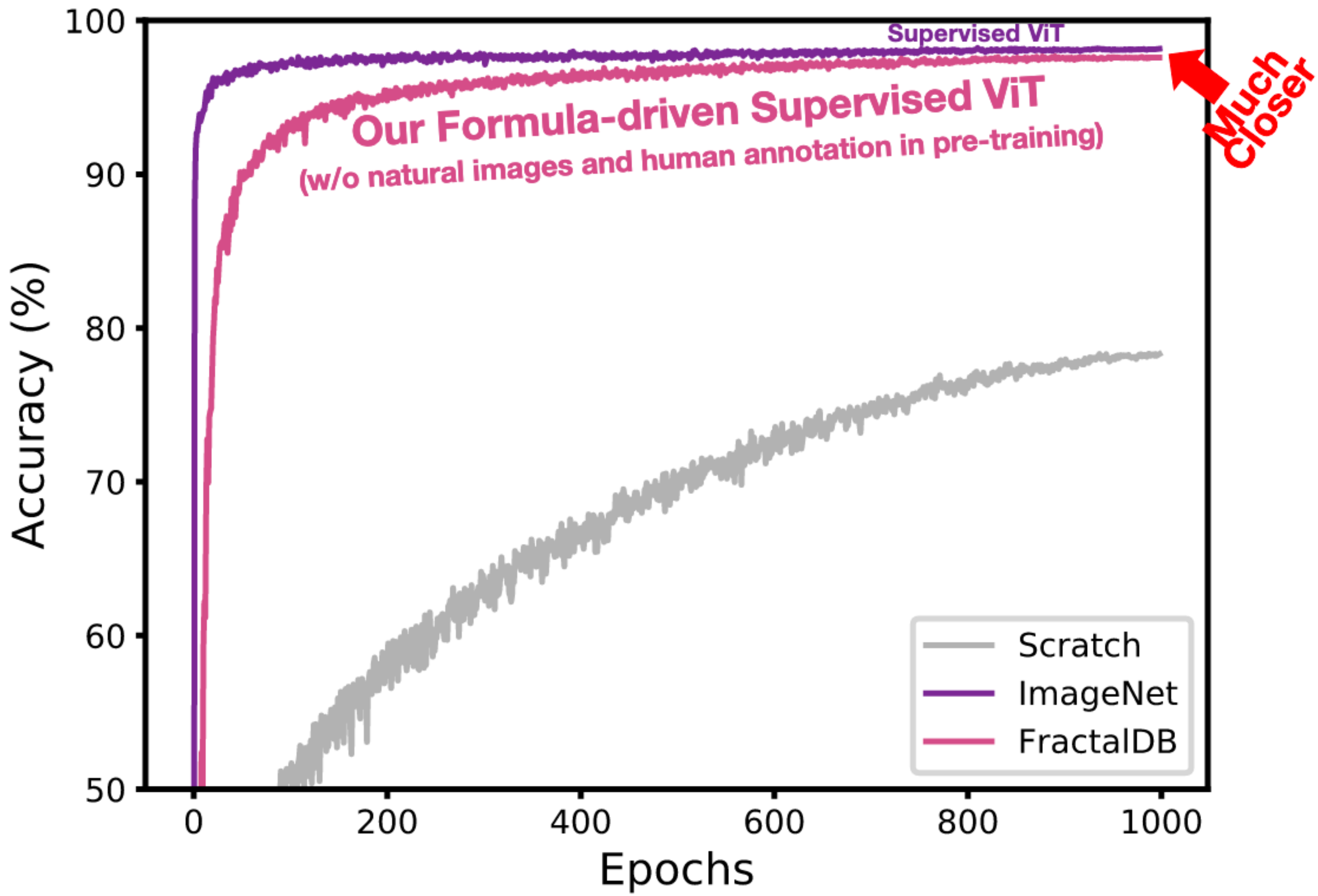
Dataset

– FractalDB

- Grayscale is better than colored FractalDB
  - ResNet: colored FractalDB is slightly better
  - DeiT: grayscale FractalDB is better
- Longer pre-training is better
  - 300 epochs in ViT

FractalDB pre-trained Vision Transformer

– We succeeded a ViT pre-training without real images



Results (1/2)

vs. Supervised Learning

PT	PT Img	PT Type	C10	C100	Cars	Flowers	VOC12	P30	IN100
Scratch	–	–	78.3	57.7	11.6	77.1	64.8	75.7	73.2
Places-30	Natural	Supervision	95.2	78.5	69.4	96.7	77.6	–	86.5
Places-365	Natural	Supervision	**97.6**	**83.9**	**89.2**	**99.3**	84.6	–	<u>**89.4**</u>
ImageNet-100	Natural	Supervision	94.7	77.8	67.4	97.2	78.8	78.1	–
ImageNet-1k	Natural	Supervision	<u>**98.0**</u>	<u>**85.5**</u>	<u>**89.9**</u>	<u>**99.4**</u>	<u>**88.7**</u>	<u>**80.0**</u>	–
FractalDB-1k	Formula	Formula-supervision	96.8	81.6	86.0	98.3	84.5	78.0	87.3
FractalDB-10k	Formula	Formula-supervision	**97.6**	83.5	87.7	98.8	**86.9**	**78.5**	**88.1**

<u>**Underlined bold**</u>: best score, **Bold**: second best score

FractalDB pre-trained model showed significantly improved performance compared to training from scratch

Results (1/2)

vs. Supervised Learning

PT	PT Img	PT Type	C10	C100	Cars	Flowers	VOC12	P30	IN100
Scratch	–	–	78.3	57.7	11.6	77.1	64.8	75.7	73.2
Places-30	Natural	Supervision	95.2	78.5	69.4	96.7	77.6	–	86.5
Places-365	Natural	Supervision	**97.6**	**83.9**	**89.2**	**99.3**	84.6	–	<u>**89.4**</u>
ImageNet-100	Natural	Supervision	94.7	77.8	67.4	97.2	78.8	78.1	–
ImageNet-1k	Natural	Supervision	<u>**98.0**</u>	<u>**85.5**</u>	<u>**89.9**</u>	<u>**99.4**</u>	<u>**88.7**</u>	<u>**80.0**</u>	–
FractalDB-1k	Formula	Formula-supervision	96.8	81.6	86.0	98.3	84.5	78.0	87.3
FractalDB-10k	Formula	Formula-supervision	**97.6**	83.5	87.7	98.8	**86.9**	**78.5**	**88.1**

<u>**Underlined bold**</u>: best score, **Bold**: second best score

Though our method was not able to beat the ImageNet pre-trained model, the FractalDB pre-trained model partially surpassed the Places

Results (2/2)

vs. Self-supervised Learning

Method	Use Natural Images?	C10	C100	Cars	Flowers	VOC12	P30	Average
Jigsaw	YES	96.4	82.3	55.7	98.2	82.1	**80.6**	82.5
Rotation	YES	95.8	81.2	70.0	96.8	81.1	79.8	84.1
MoCov2	YES	96.9	83.2	78.0	98.5	85.3	<u>**80.8**</u>	87.1
SimCLRv2	YES	**97.4**	<u>**84.1**</u>	**84.9**	<u>**98.9**</u>	**86.2**	80.0	**88.5**
FractalDB-10k	NO	<u>**97.6**</u>	**83.5**	<u>**87.7**</u>	**98.8**	<u>**86.9**</u>	78.5	<u>**88.8**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

The proposed method recorded higher scores compared to SSL methods such as MoCoV2, rotation, and jigsaw puzzle

Results (2/2)

vs. Self-supervised Learning

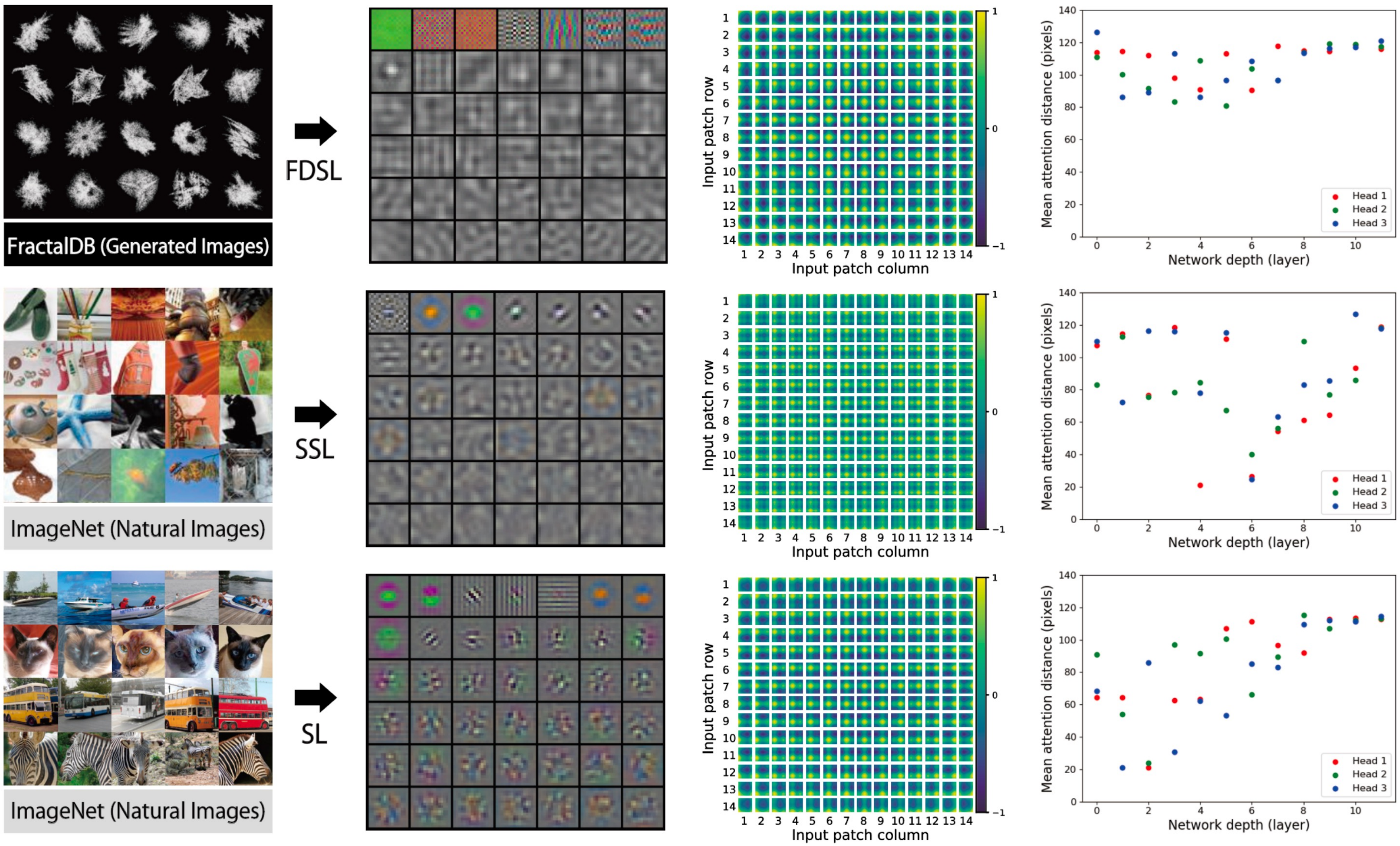
Method	Use Natural Images?	C10	C100	Cars	Flowers	VOC12	P30	Average
Jigsaw	YES	96.4	82.3	55.7	98.2	82.1	**80.6**	82.5
Rotation	YES	95.8	81.2	70.0	96.8	81.1	79.8	84.1
MoCov2	YES	96.9	83.2	78.0	98.5	85.3	<u>**80.8**</u>	87.1
SimCLRv2	YES	**97.4**	<u>**84.1**</u>	**84.9**	<u>**98.9**</u>	**86.2**	80.0	**88.5**
FractalDB-10k	NO	<u>**97.6**</u>	**83.5**	<u>**87.7**</u>	**98.8**	<u>**86.9**</u>	78.5	<u>**88.8**</u>

<u>**Underlined bold**</u>: best score, **Bold**: second best score

FractalDB-10k pre-trained ViT recorded a slightly higher in average accuracy on various benchmarks (88.8 vs. 88.5)

Visualization

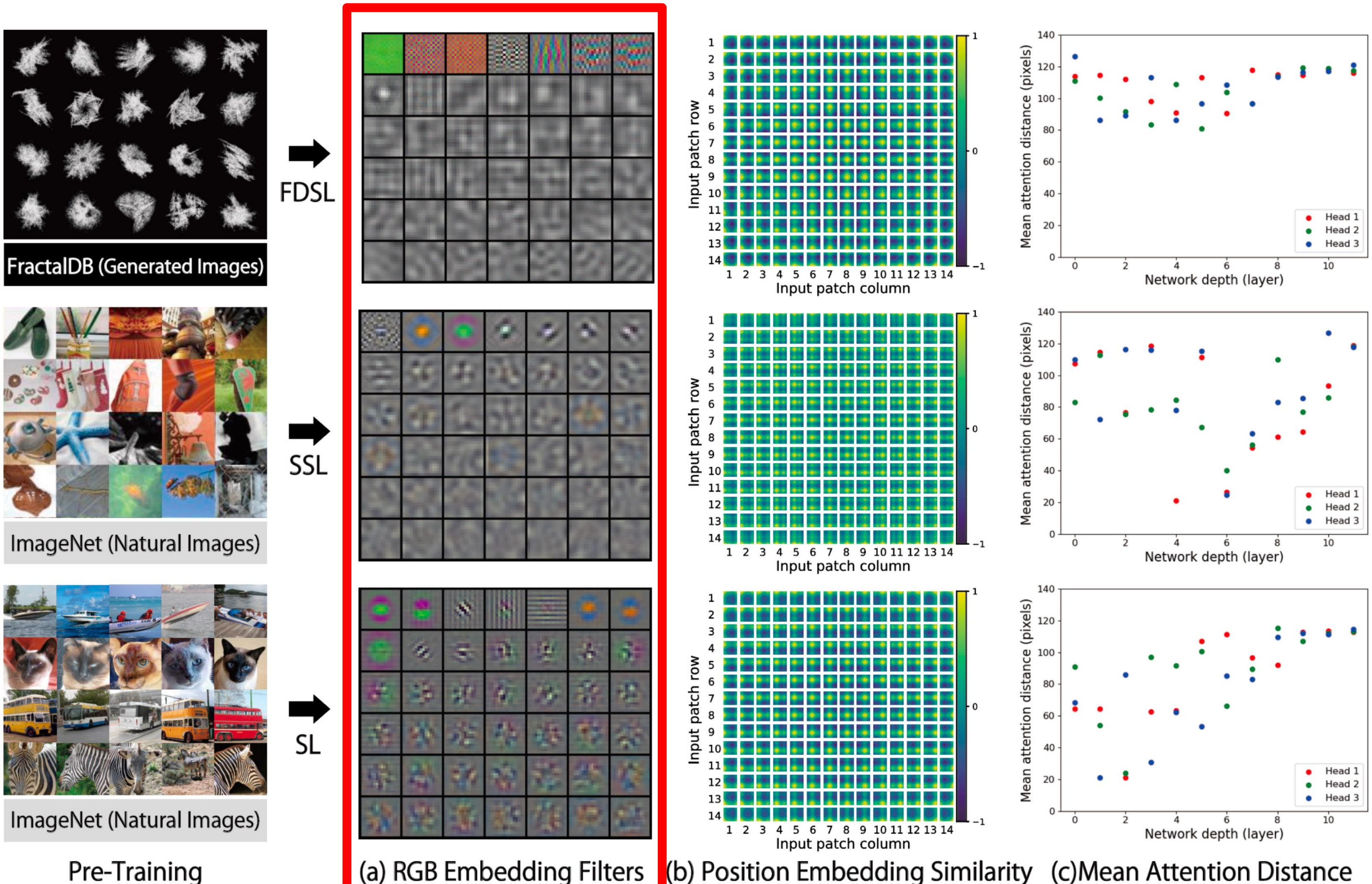
Characteristics of FDSL, SSL, and SL



Pre-Training | (a) RGB Embedding Filters | (b) Position Embedding Similarity | (c)Mean Attention Distance

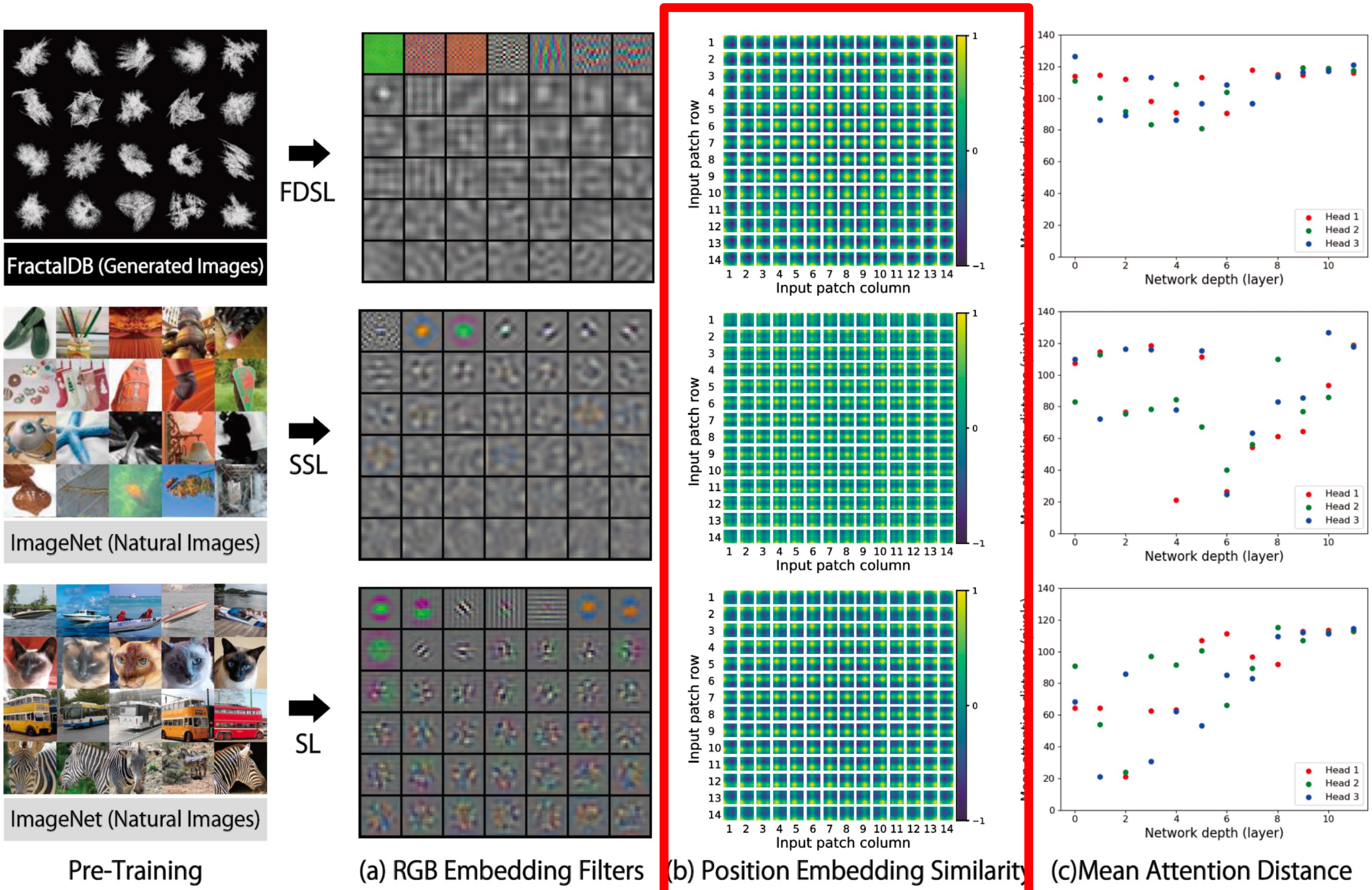
Initial filter representation

Ours is similar with SL and SSL representations



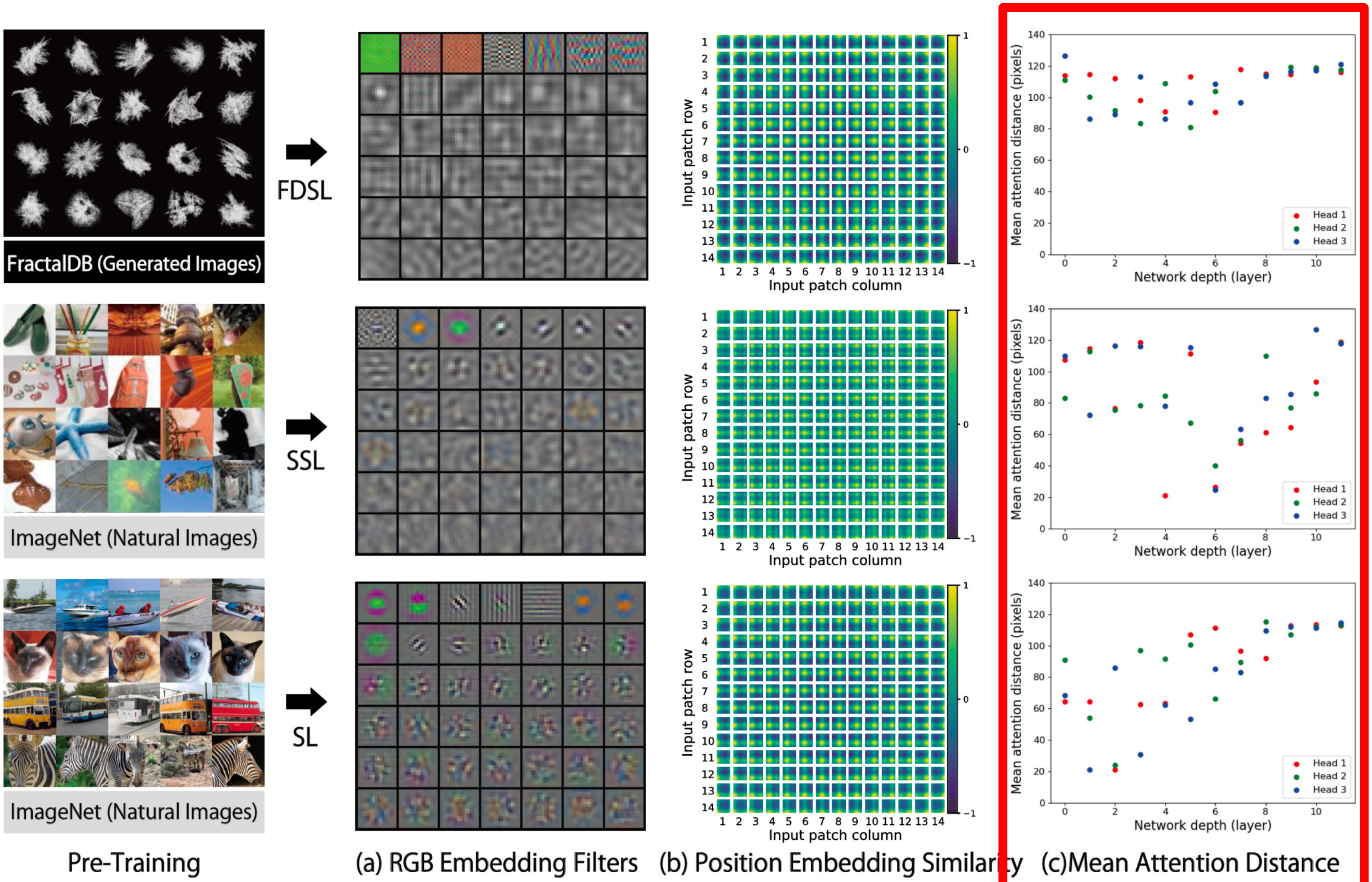
Cosine similarity of positional embeddings

Similar positional embedding to SL



Attention distance visualization

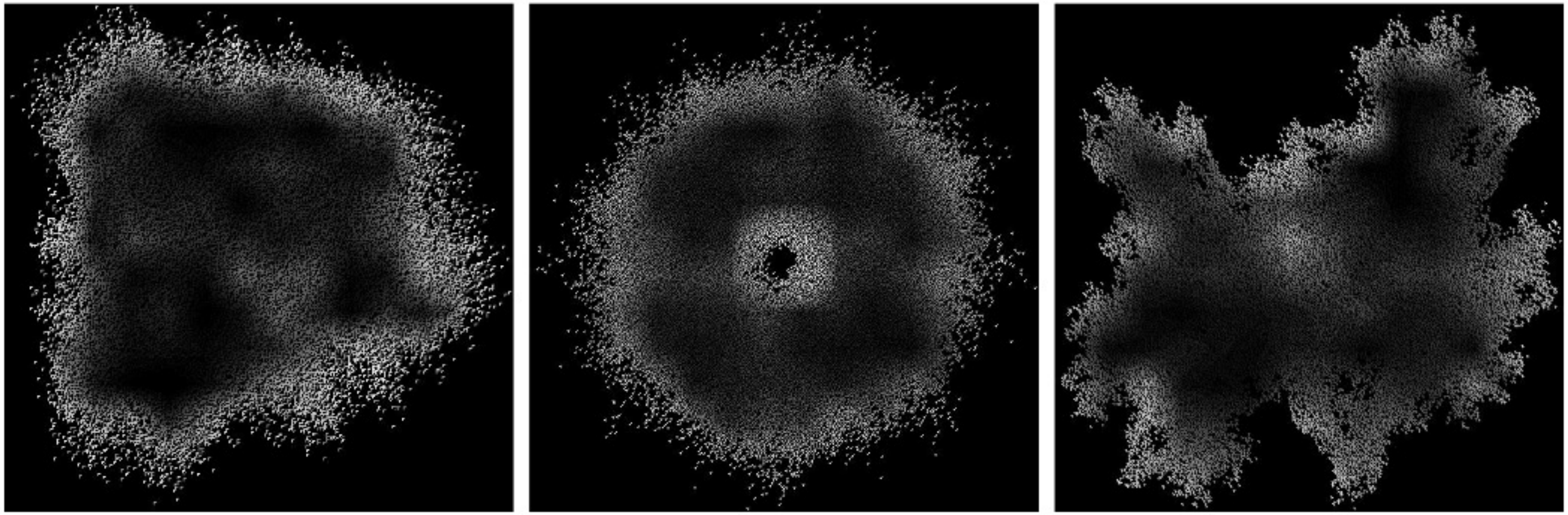
Looks at wider areas within an image



Visualization of attention maps

FractalDB pre-trained model focuses on contours

- The figures show attention on fractal images



(d) Attention maps in fractal images with FractalDB-1k pre-trained DeiT. The brighter areas show more attentive areas.

Can vision transformers learn without natural images?

→ Answer is “Yes”. The FractalDB pre-training achieved comparable performance to ImageNet-1k pre-training

Replacing Labeled Real-image Datasets with Auto-generated Contours

CVPR 2022

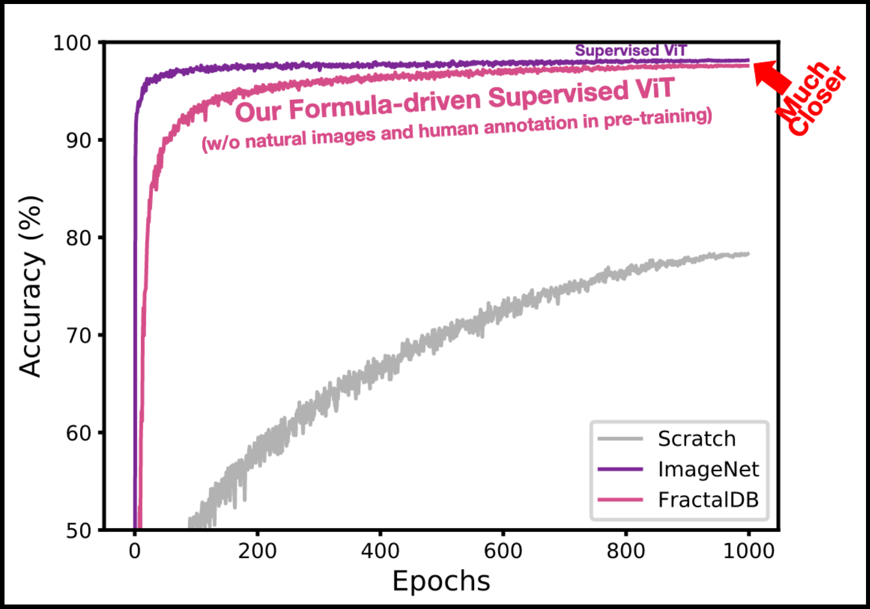
Hirokatsu Kataoka[*], Ryo Hayamizu[*], Ryosuke Yamada[*], Kodai Nakashima[*], Sora Takashima[*,**], Xinyu Zhang[*,**], Edgar Josafat MARTINEZ-NORIEGA[*,**], Nakamasa Inoue[*,**], Rio Yokota[*,**]

* National Institute of Advanced Industrial Science and Technology (AIST)
**Tokyo Institute of Technology

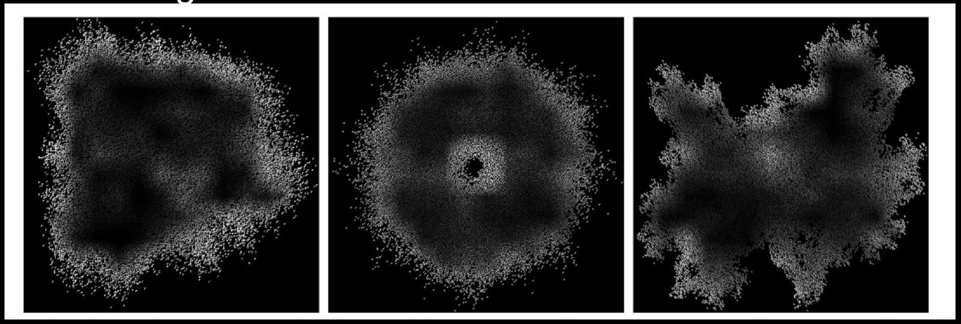
Can Vision Transformers Learn without Natural Images? (AAAI22)

Successfully trained a FractalDB pre-trained ViT

- Reducing the use of real images 14M to 0
- Exploring the reason behind the success



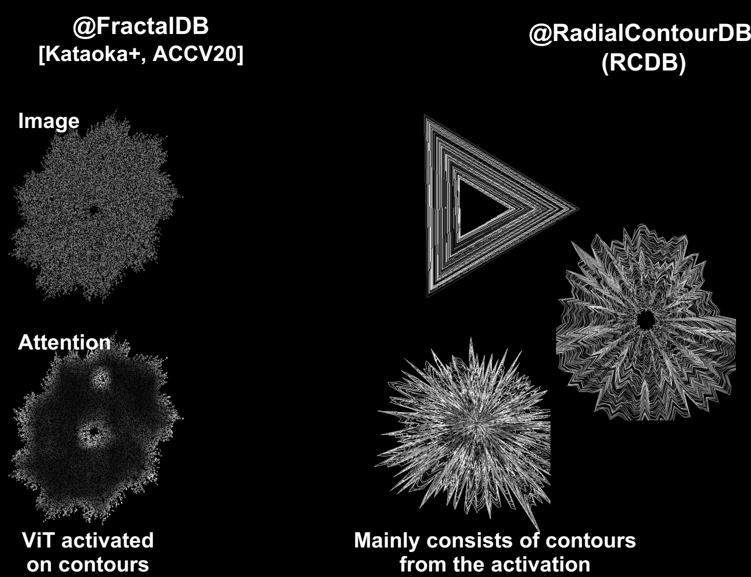
Visualizing self-attention in ViT



→ The fact describes that it focuses on object contours, rather than use of fractals

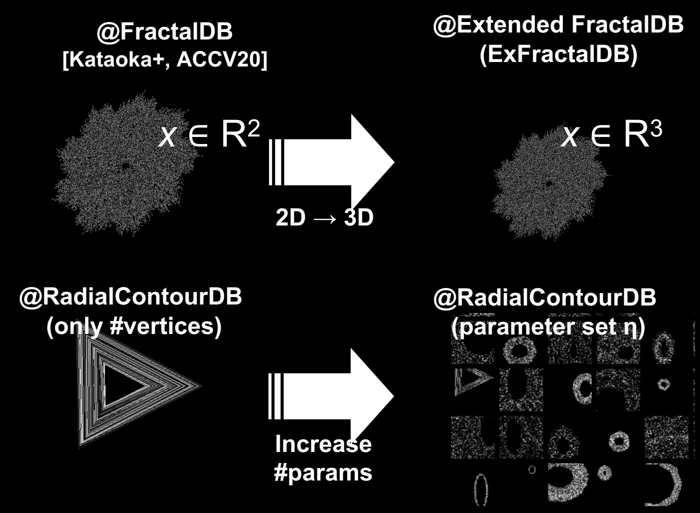
Two hypotheses regarding FDSL pre-training

Hypothesis 1:
Object contours are what matter



As the extreme case of contour classification, we implemented RCDB mainly consists of contours in an image

Hypothesis 2:
Task difficulty matters



Our finding showed that #parameters are linked to task difficulty

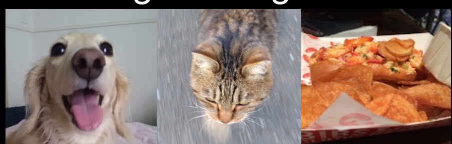
Validation on classification, detection, and segmentation

ImageNet-1k / MS COCO dataset

Image Classification / Object Detection, Instance Segmentation

Accuracy on ImageNet-1k

Real images: ImageNet-21k



81.8%

3D fractal images: ExFractalDB-21k

82.7%

Contour images: RCDB-21k

82.4%

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

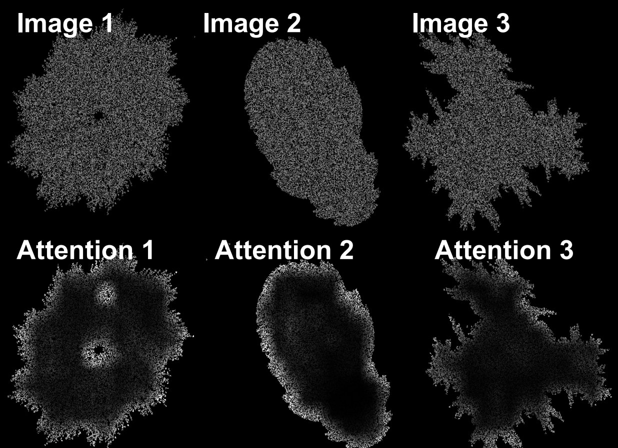
Exceeded ImageNet-21k pre-training
Radial contours also surpassed the accuracy with ImageNet pre-training in addition to Fractal pre-training

Our pre-trained models perform good fine-tuning results on COCO with a pre-training from only contour classification

Hypothesis 1

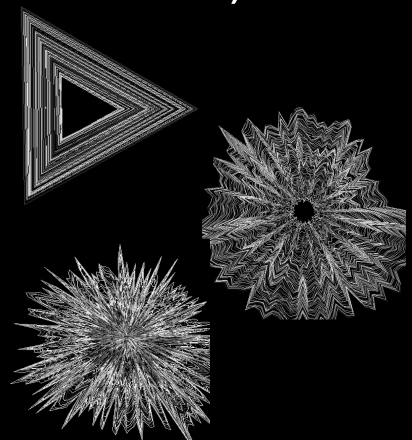
Object contours are what matter in FDSL datasets

@FractalDB [Kataoka+, ACCV20]



ViT activated on contours of fractal images

@RadialContourDB (RCDB)



RCDB mainly consists of contours

Pre-training	C10	C100	Cars	Flowers
Scratch	78.3	57.7	11.6	77.1
Perlin Noise [21]	95.0	78.4	70.6	96.1
Dead Leaves [3]	95.9	79.6	72.8	96.9
Bezier Curves [21]	96.7	80.3	82.8	98.5
RCDB	**96.8**	**81.6**	84.2	**98.7**
FractalDB [27]	**96.8**	**81.6**	**86.0**	98.3

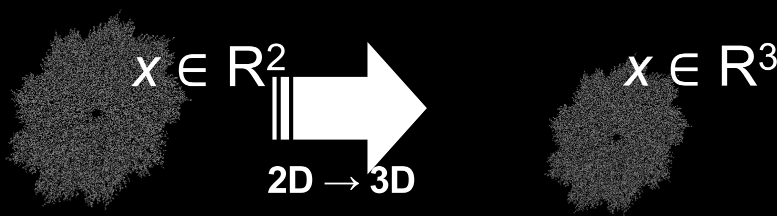
Radial contour pre-training achieved similar results as FractalDB without extensive parameter tuning

Hypothesis 2

Task difficulty matters in FDSL pre-training

@FractalDB
[Kataoka+, ACCV20]

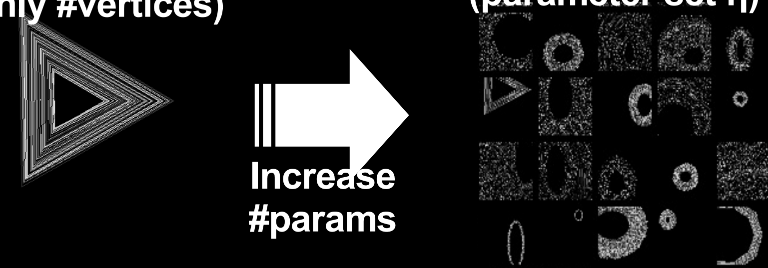
@Extended FractalDB
(ExFractalDB)



- 3D Fractal rendering
- Projecting onto 2D image plane from a random viewpoint

@RadialContourDB
(only #vertices)

@RadialContourDB
(parameter set η)



- We mainly adjust #vertices
- Additional parameters, e.g., #polygons, smoothness for category generation

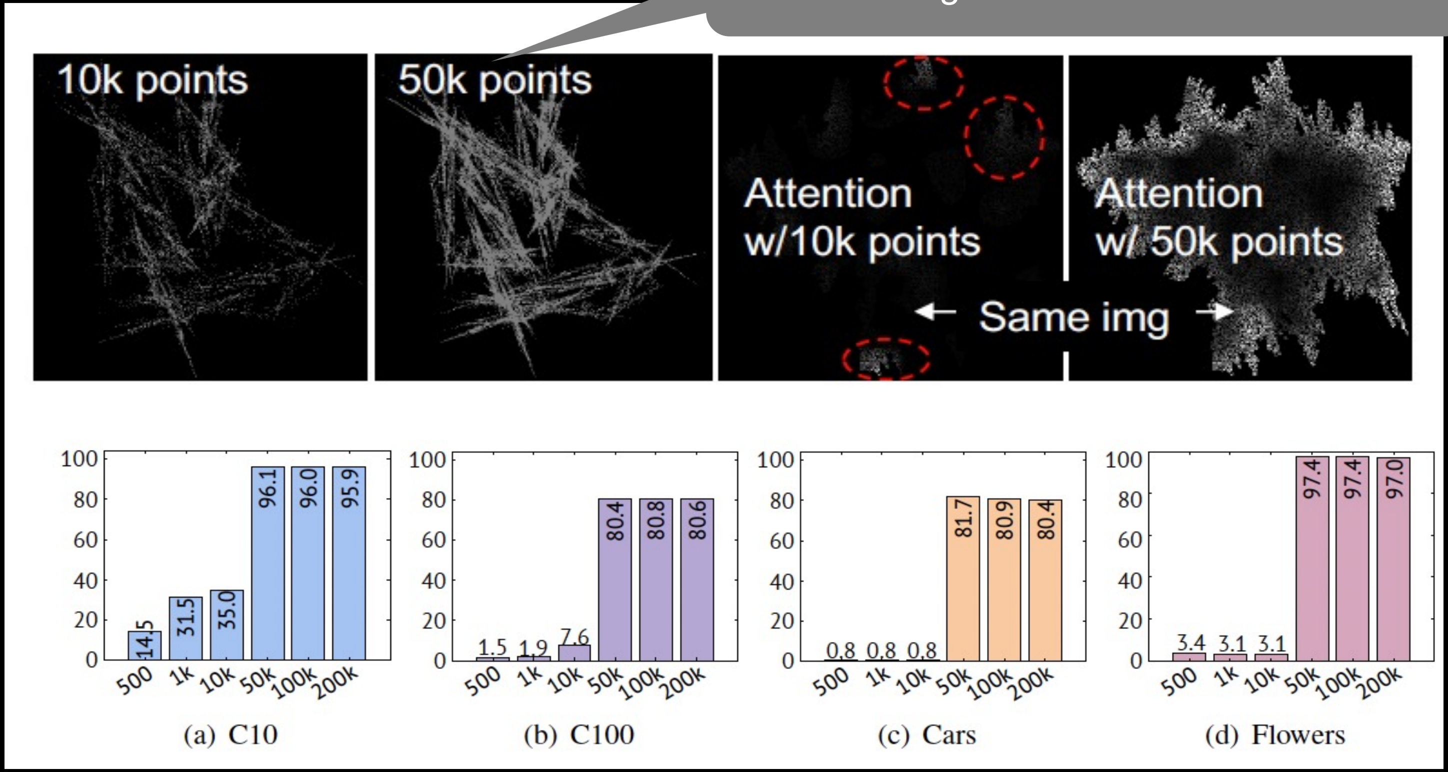
Pre-training	C10	C100	Cars	Flowers
BC	96.9 (0.2)	81.4 (1.1)	85.9 (3.1)	97.9 (-0.6)
RCDB	97.0 (0.2)	**82.2** (0.6)	86.5 (2.4)	**98.9** (0.2)
ExFractalDB	**97.2** (0.4)	81.8 (0.2)	**87.0** (1.0)	**98.9** (0.6)

In relation to #formula-parameters, the image variation contributes to the pre-training effect

Failure modes in FractalDB

Investigate when and how FDSL can fail

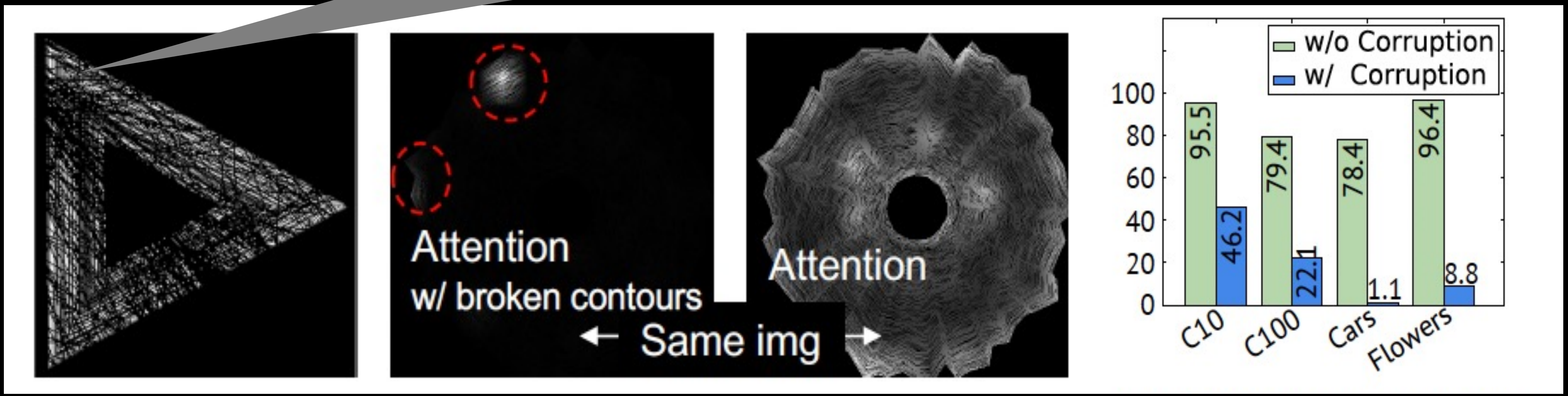
Fractal images start to form a contour in 50k or higher



In point-rendered FractalDB, although the fractal images with 50k points trained the visual representations, the fractal images with 10k points failed

Failure modes in RCDB

We deliberately draw lines with the same color as the background



At the same time, the RCDB with broken contours failed to acquire a visual representation. The attention and accuracy were also broken from the visualization and result

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

Visual Atoms: Pre-training Vision Transformers with Sinusoidal Waves

CVPR 2023 [new!]

Sora Takashima[*,], Ryo Hayamizu[*],**
Nakamasa Inoue[*,], Hirokatsu Kataoka[*], Rio Yokota[*,**]**

*** National Institute of Advanced Industrial Science and Technology (AIST)**
****Tokyo Institute of Technology**

How contours important in pre-training?

Throughout many experiments, the diversity of contours

- Frequency, orbits, vertices, quantization…

Table 3. Accuracy when varying the range of frequency parameters n_1, n_2.

Range of n, m min	max	C10	C100	IN100
0	20	**97.6**	**84.9**	**90.3**
0	40	97.1	84.3	89.5
0	60	97.3	84.1	89.1
2	20	**97.6**	84.8	**90.3**
10	20	97.5	84.4	89.8
20	20	97.3	83.6	89.6

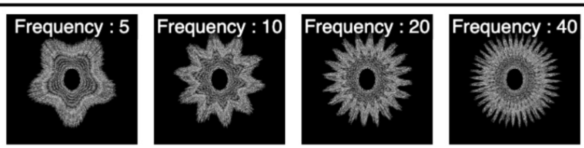


Table 4. Accuracy when varying the range of number of orbits K.

Range of K min	max	C10	C100	IN100
1	200	**97.6**	**84.9**	**90.3**
20	200	97.5	84.7	89.9
100	200	97.5	84.5	89.8
200	200	97.5	84.3	89.4

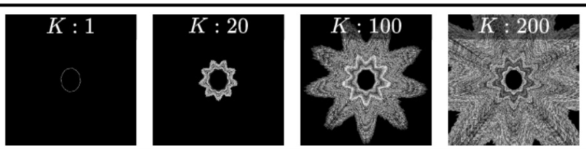
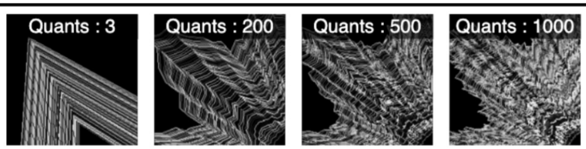


Table 5. Accuracy when varying the range of quantization parameter q.

Range of q min	max	C10	C100	IN100
200	1,000	**97.6**	84.9	**90.3**
800	1,000	97.4	**85.1**	89.9
3	200	97.3	84.6	89.7
3	500	97.3	84.9	90.1
3	1,000	97.4	85.0	90.1



Param in contours of “frequency”

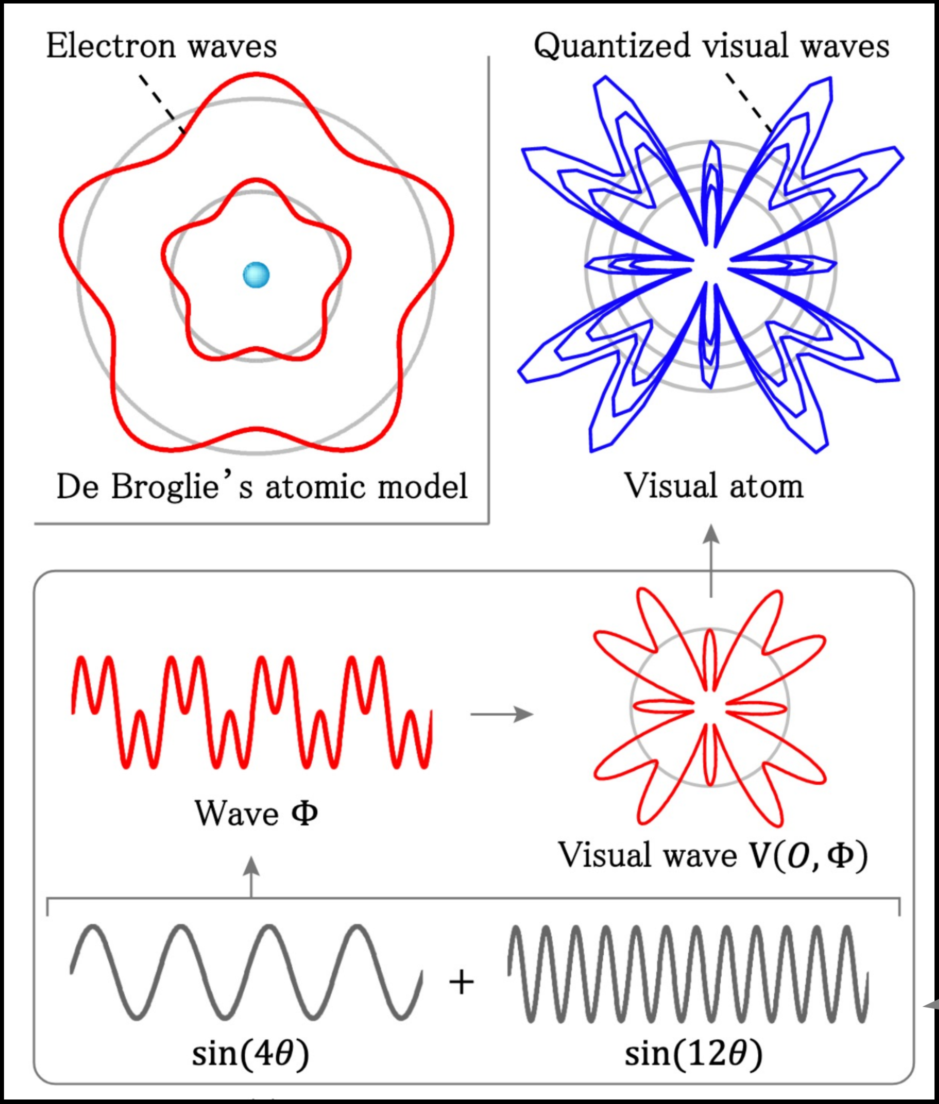
Param in contours of “orbits”

Param in contours of “vertices”

…We’ve carried out the experiments with over 1 million GPU hours

How contours important in pre-training?

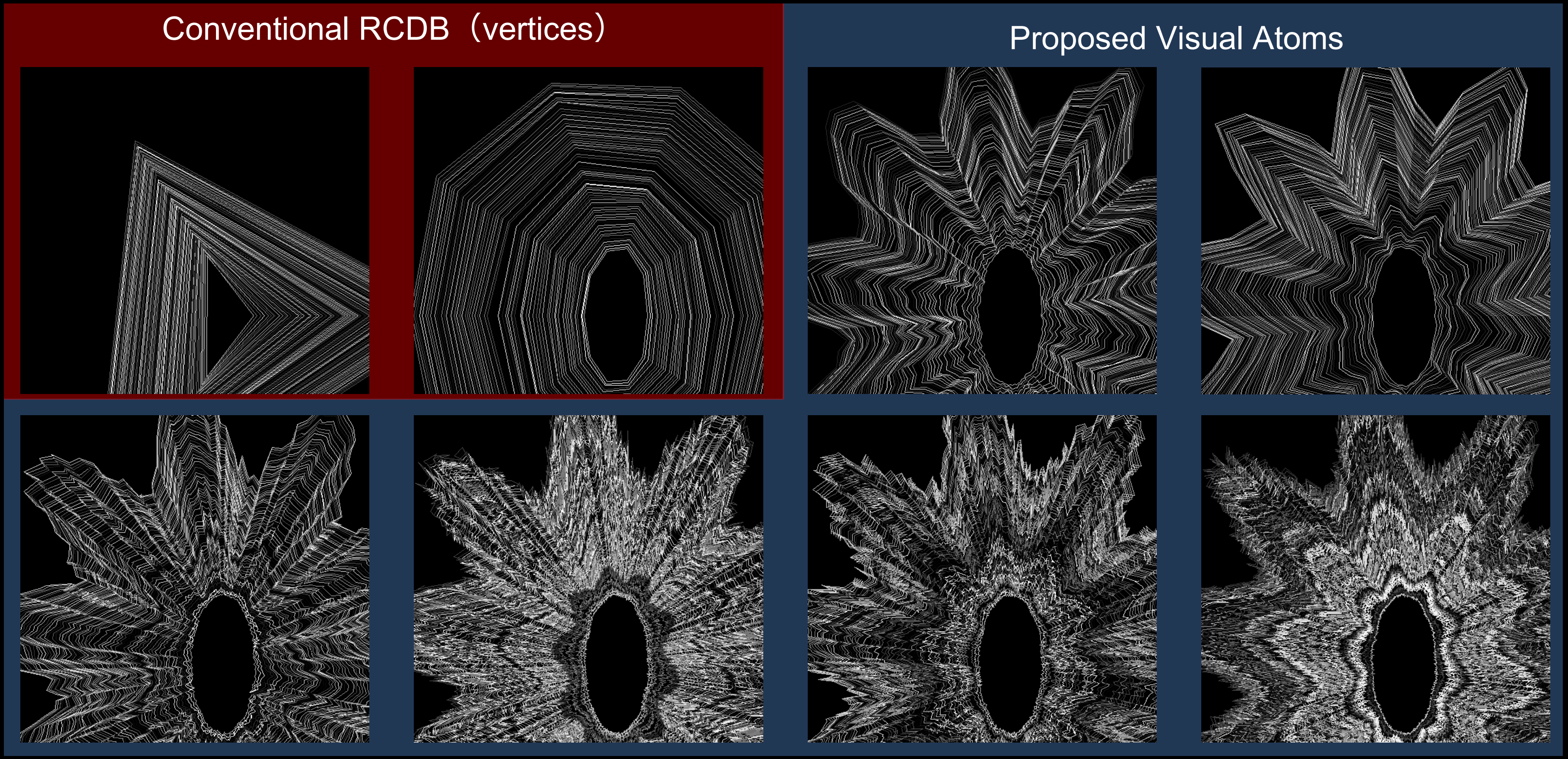
Throughout many experiments, the diversity of contours



Combined two different sine curves (sinusoidal waves)

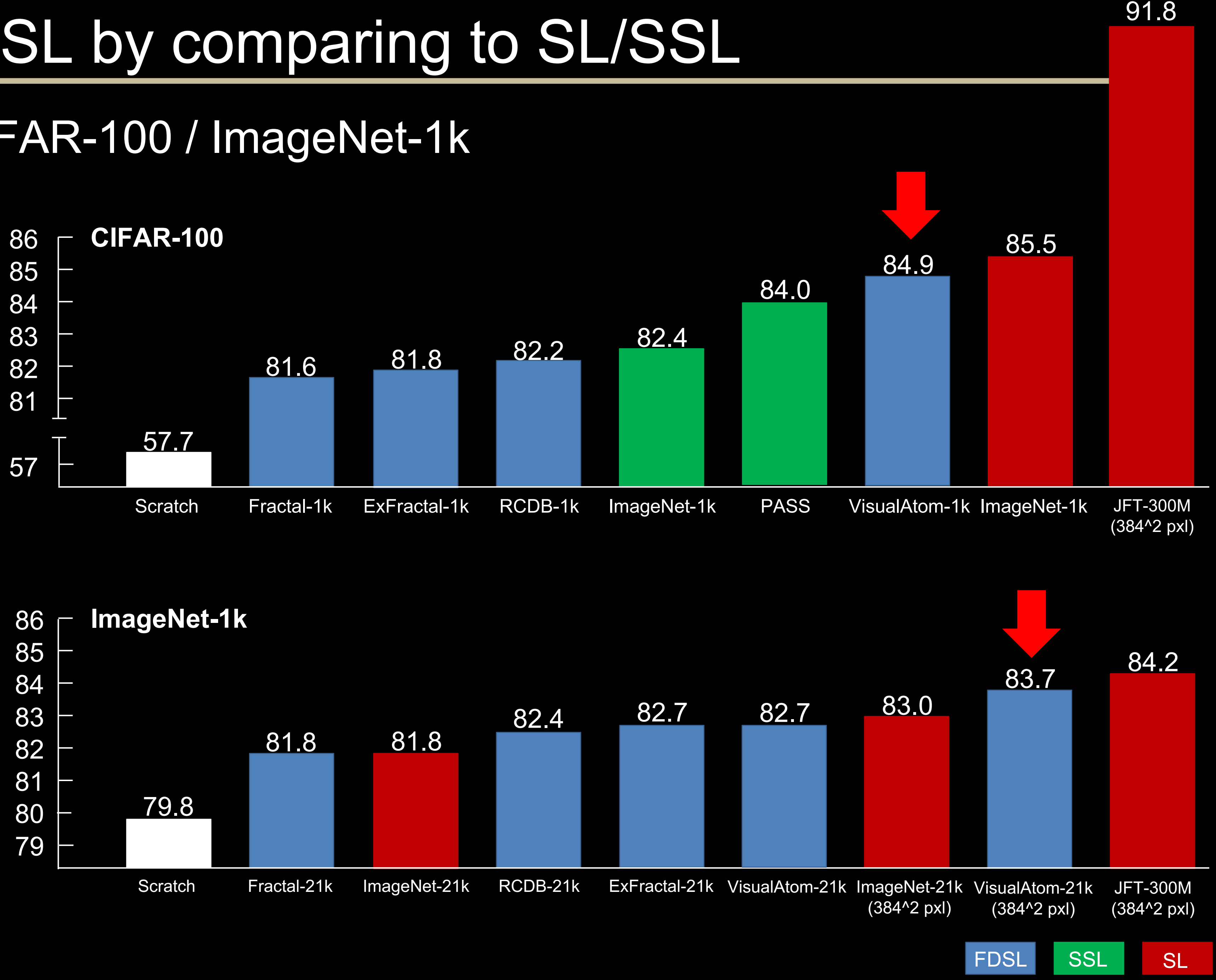
RCDB (vertices) vs. VisualAtom (2 sine curves)

Vertices vs. 2 sine curves



FDSL by comparing to SL/SSL

CIFAR-100 / ImageNet-1k



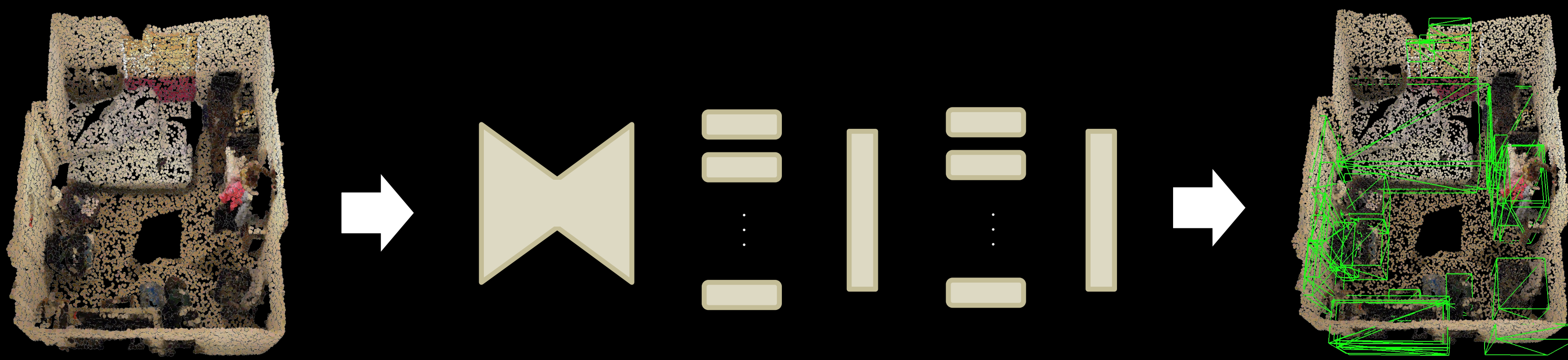
Point Cloud Pre-training with Natural 3D Structures

CVPR 2022

Ryosuke Yamada[*], Hirokatsu Kataoka[*], Naoya Chiba[**], Yukiyasu Domae[*], Testuya Ogata[*, **]

*** National Institute of Advanced Industrial Science and Technology (AIST)**
****Waseda University**

Construction of a pre-training 3D dataset is challenging, as there is no equivalent to ImageNet in the 2D image domain

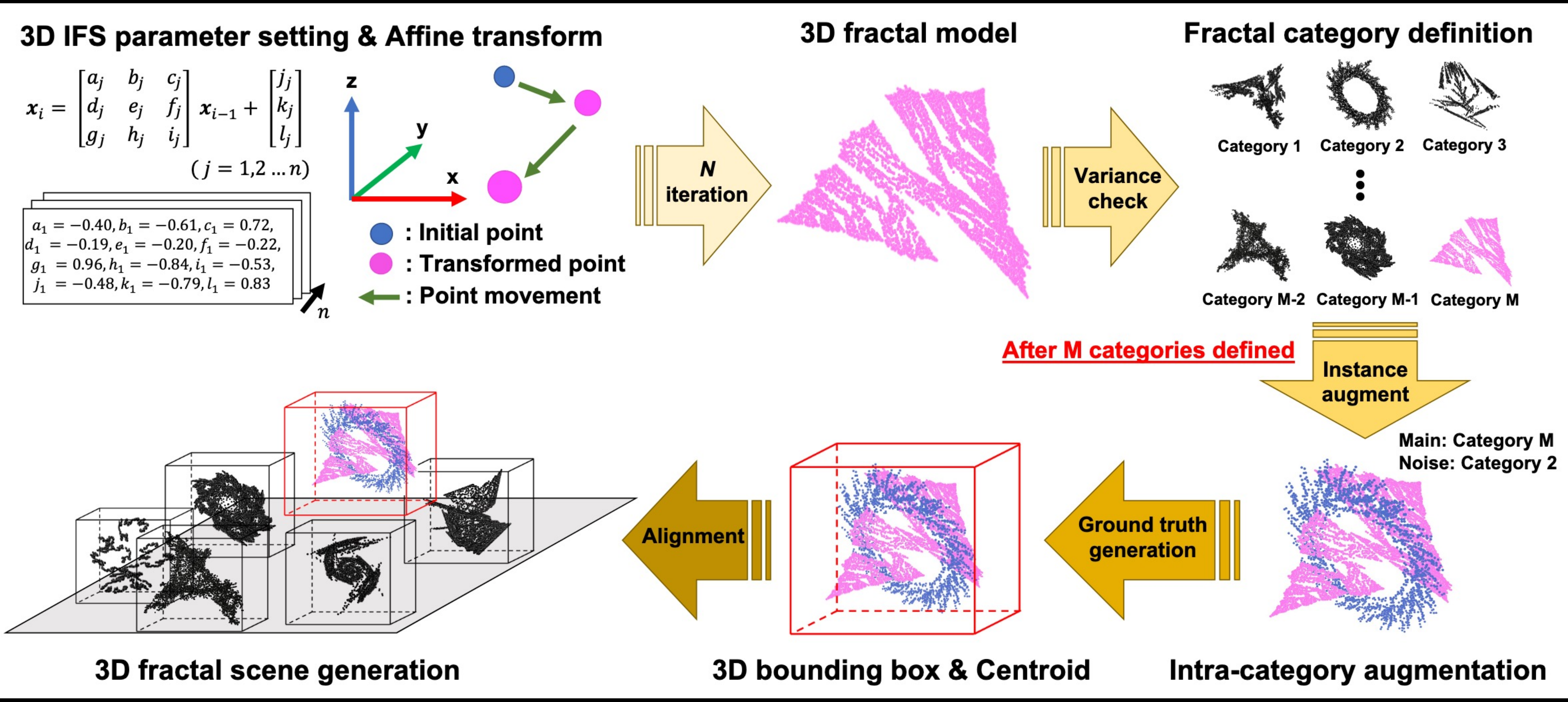


Input **3D Object Detection Network** **Output**

Can we acquire a general 3D representation from a principle in our real world?

Formula-driven 3D Point Cloud Pre-training

Overview of formula-driven 3D point cloud pre-training



Point Cloud Fractal Database: 3D fractal generation

How could we render 3D Fractal model
→ Extend the transformation matrix from 2D to 3D

3D IFS $= \{(\boldsymbol{w}_j, p_j)\}_{j=1}^{N}$

$\boldsymbol{w}_j$: Affine Transformation
p_j: Selection probability

1. 3D-IFS parameters setting

$$\mathcal{W}_1 = \begin{bmatrix} 0.57 & -0.68 & 0.40 \\ -0.55 & -0.61 & -0.16 \\ -0.59 & 0.63 & 0.08 \end{bmatrix} + \begin{bmatrix} 0.18 \\ -0.22 \\ 0.50 \end{bmatrix}$$

N

3. Variance check & category definition

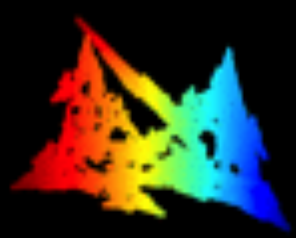
$$min(Var[x], Var[y], Var[z]) = \mathbf{0.17}\ldots > 0.15$$

2. Affine transformation

$$\boldsymbol{x}_i = w_j\ \boldsymbol{x}_{i-1}$$

$(i = 1, 2, 3, .., n)$

$\boldsymbol{x} = [x, y, z]^T$

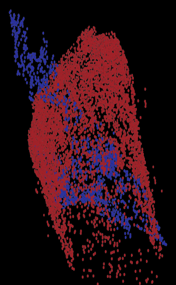


3D fractal model: $P = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$

Point Cloud Fractal Database: Instance aug.

Instance augmentation / 3D scene generation

Mixed instance from 2 models



FractalNoiseMix

Main Category (80%)

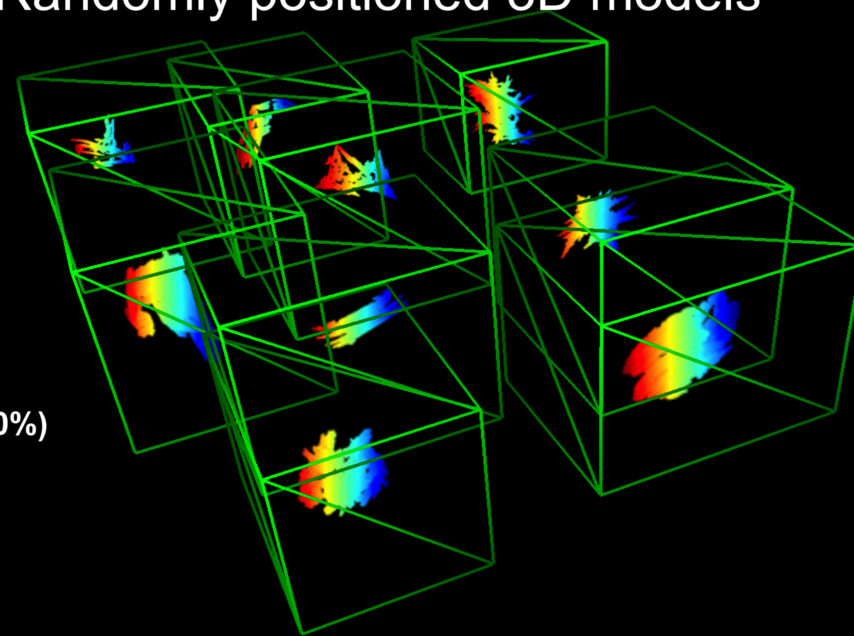
#Points: 3,200

#Points: 4,000

Noise Category(20%)

#Points: 800

Randomly positioned 3D models



Important to construct a 3D scene from 3D fractal models

Experimental results: 3D object detection in point clouds

Comparisons on ScanNetV2 / SUN RGB-D

Pre-training	Backbone	Parameter	Input	ScanNetV2 mAP@0.25	ScanNetV2 mAP@0.50	SUN RGB-D mAP@0.25	SUN RGB-D mAP@0.50
Scratch	PointNet++	0.95M	Geo + Height	57.9	32.1	57.4	32.8
Scratch	SR-UNet	38.2M	Geo	57.0	35.8	56.1	34.2
RandomRooms [51]	PointNet++	0.95M	Geo + Height	61.3	36.2	59.2	35.4
PointContrast [67]	SR-UNet	38.2M	Geo	59.2	38.0	57.5	34.8
CSC [26]	SR-UNet	38.2M	Geo	-	**39.3**	-	<u>**36.4**</u>
PC-FractalDB	PointNet++	0.95M	Geo + Height	**61.9**	38.3	**59.4**	33.9
PC-FractalDB	PointNet++ ×2	38.2M	Geo + Height	<u>**63.4**</u>	<u>**39.9**</u>	<u>**60.2**</u>	35.2
PC-FractalDB	SR-UNet	38.2M	Geo	59.4	37.0	57.1	**35.9**

<u>**Underlined bold**</u>: best score **Baseline** **Ours**

PC-FractalDB 61.9 vs 59.2 (PointContrast; ECCV 2020)
vs 61.3 (RandomRoom; ICCV 2021)

ScanNetV2 / mAP @ 0.25

Classification PT vs. detection PT in 3D point cloud

Pre-training comparison between classification and detection

- We only add detection head in VoteNet, with PointNet++ backbone

	ScanNetV2 mAP@0.25	SUN RGB-D mAP@0.25
PointNet++	48.8	49.8
VoteNet	**61.1**	**57.6**

Detection pre-training performs much higher scores

Self-supervision vs. formula-supervision in synthetic 3D models

Self-supervised label and formula-supervised label on PC-FractalDB

- Self-supervised label: PointContrast (ECCV 2020)
- Formula-supervised label: Fractal category (ours)

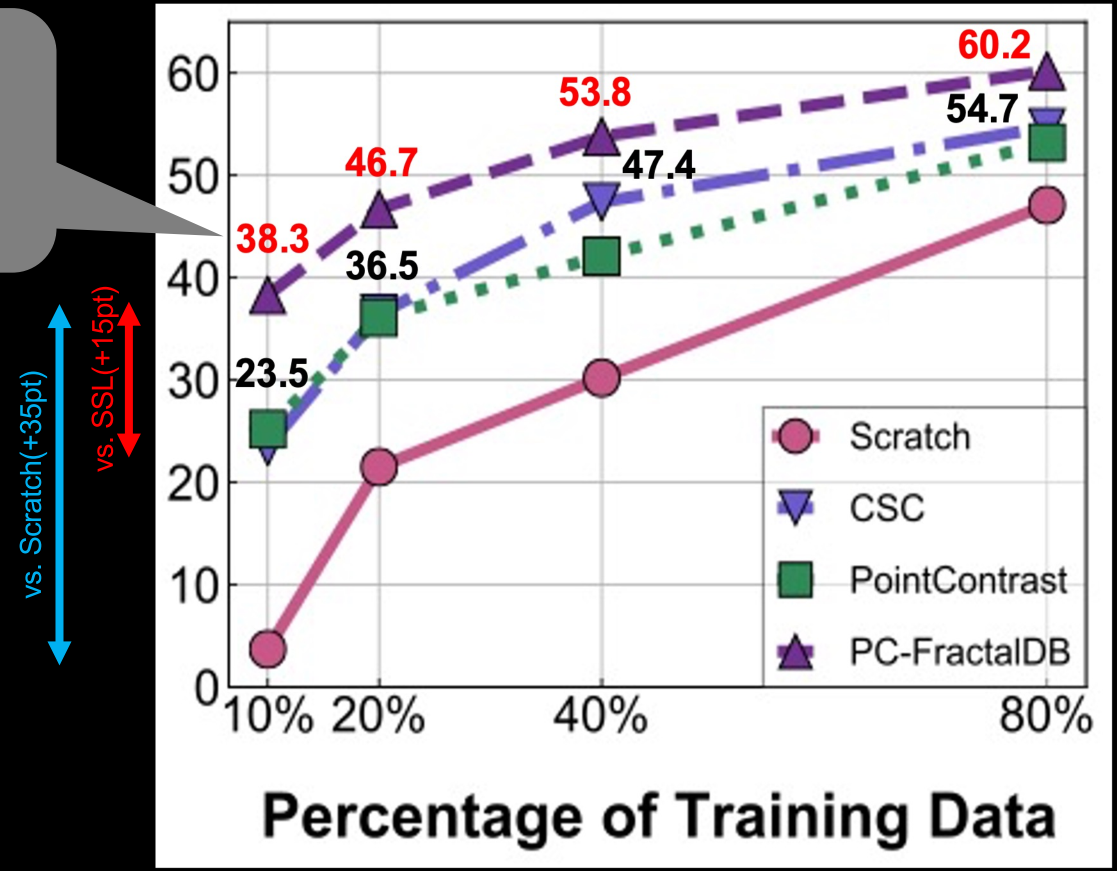
Supervisor label	ScanNetV2 mAP@0.25	SUN RGB-D mAP@0.25
PointContrast (SSL)	57.6	54.3
3D IFS (FDSL)	**59.4**	**57.1**

It is better to assign data and label from a single equation

Experimental results: Limited data/annotation

Higher accuracy on a dataset with limited data

10% amount :
+15% vs. SSL
+35% vs. from scratch



mAP@0.25

SegRCDB: Formula-driven Supervised Learning for Semantic Segmentation

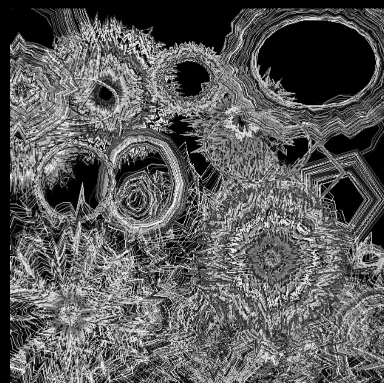
ICCV 2023

Risa Shinoda [*], Ryo Hayamizu [*], Kodai Nakashima [*], Nakamasa Inoue [*, **], Rio Yokota [*, **], Hirokatsu Kataoka [*]

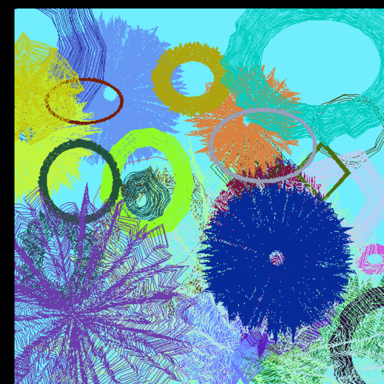
*National Institute of Advanced Industrial Science and Technology (AIST)
**Tokyo Institute of Technology

SegRCDB is used to accelerate a semantic segmentation pre-training without any human supervision and real images

Pre-training images



Semantic labels



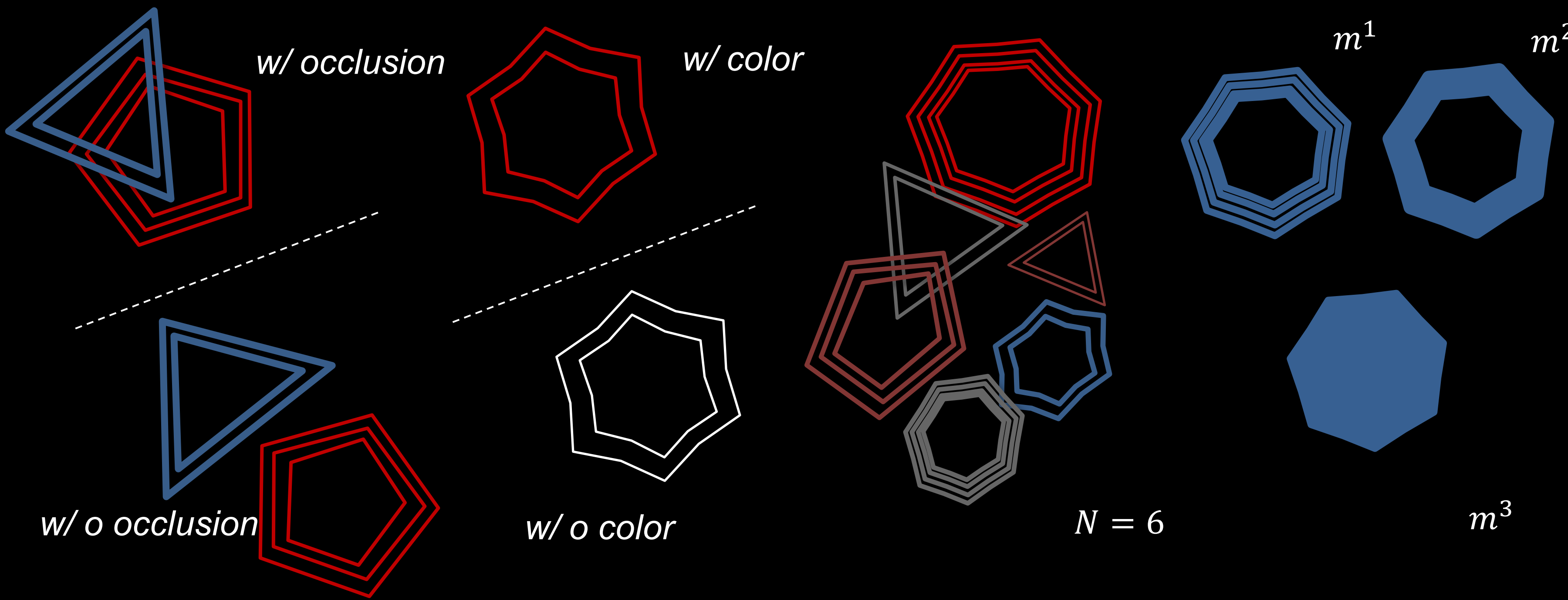
Pre-training	Fine-tuning @ADE20k
COCO Stuff-164k	43.39
RCDB	41.07
SegRCDB (Ours)	**43.85**

Pre-training	Fine-tuning @Cityscapes
GTA5	71.00
RCDB	69.66
SegRCDB (Ours)	**73.06**

SegRCDB enables to improve segmentation pre-training and surpass a real-image pre-training

What matters in semantic segmentation pre-training?

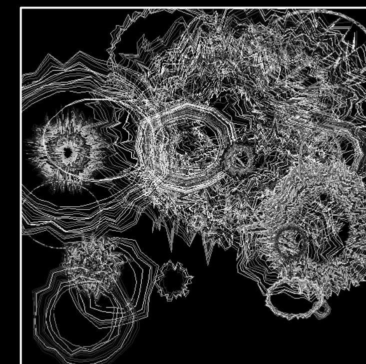
We have investigated …

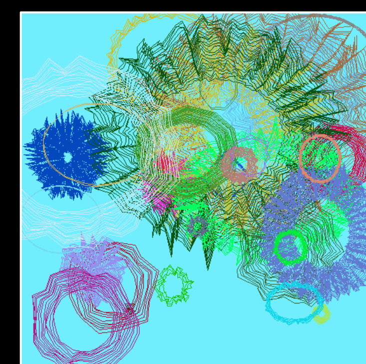


Degree of occlusion *Colorization* *# of objects per image* *Mask patterns*

What matters in semantic segmentation pre-training?

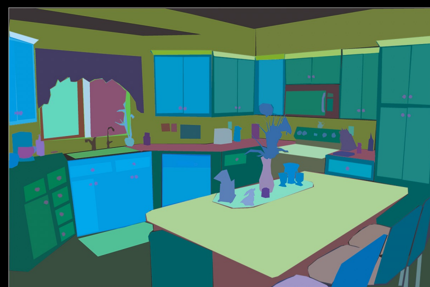
	Best parameters
(F1) # of instances	32
(F2) Mask patterns	M^1
(F3) Colorization	Grayscale
(F4) Degree of occlusion	400
(F5) Instance shape	1pix, 1-25 polygons
(F6) # of categories	255
(F7) # of images per dataset	118k





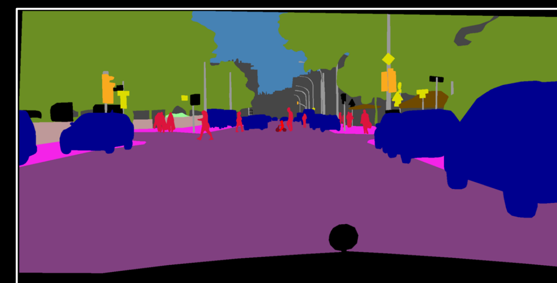
Fine-tuning for semantic segmentation datasets

Indoor Scenes



ADE-20k

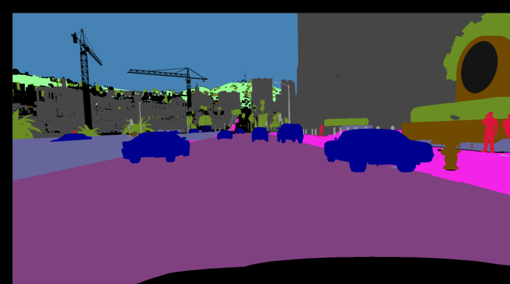
Urban Scenes



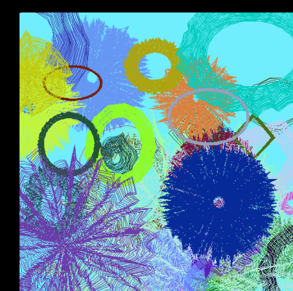
Cityscapes

Fine-tuning

COCO Stuff-164k



GTA5



SegRCDB

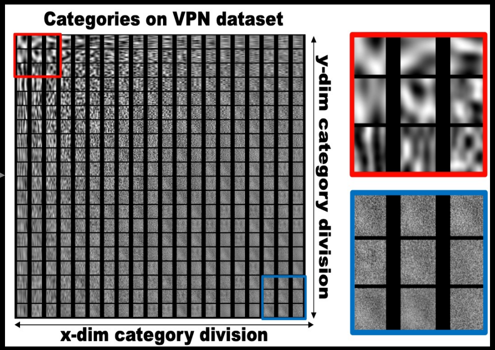
Fine-tuning for semantic segmentation datasets

		ADE-20k		Cityscapes	
Pre-training	#Img	mIoU	mAcc	mIoU	mAcc
Scratch	-	31.40	41.02	54.65	62.89
ADE-20k	20k	-	-	68.46	77.13
GTA5	25k	39.31	49.79	71.00	79.31
COCO-Stuff	118k	**43.39**	**54.41**	**72.21**	**80.62**
SegRCDB	118k	**<u>43.85</u>**	**<u>54.98</u>**	**<u>73.06</u>**	**<u>81.59</u>**

- SegRCDB pre-training surpassed the fine-tuning performance from the other synthetic and real-image pre-training
- Semantic labels in addition to category labels are beneficial for segmentation pre-training

【FDSL Family】

Spatiotemporal Domain

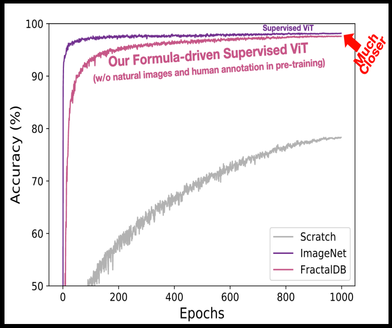


動画ドメインにも
適用できる

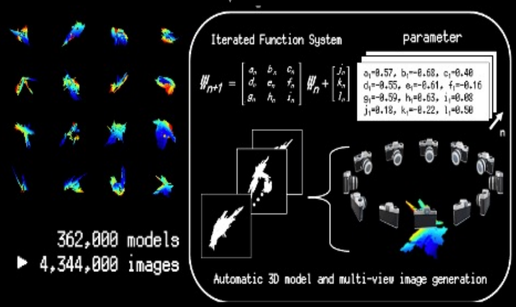
Video Perlin Noise
[Kataoka+, WACV22]

3D Domain

Vision Transformers



FractalDB Pre-trained ViT
[Nakashima+, AAAI22]



3Dドメインにも
適用できる

Multi-viewpoint [Yamada+, IROS22]
Point Cloud [Yamada+, CVPR22]

Enhanced by Hypotheses

Replacing Labeled Real-image Datasets [Kataoka+, CVPR22]
Visual Atoms [Takashima+, CVPR23]

輪郭形状の識別で
ViTを事前学習する

Future direction (1/4)

Aim to explore better pre-trained models

- FDSL pre-training partially outperformed supervised pre-training with real images, e.g., ImageNet-1k/Places-365
- 80M Tiny Images/ ImageNet (human-related categories) withdrew the public access
- FDSL achieved impressive results without relying on real images

Future direction (2/4)

FDSL exhibits a unique capability to understand natural images without any natural images

- FDSL allows for steerable pre-training adapts to the fine-tuning task at hand
- Free to create a diverse labeled dataset: Geometric model, object detection, semantic segmentation…
- FDSL has the potential to be a flexible pre-training dataset for a broad range of tasks

Future direction (3/4)

Are fractals a good rendering formula?

- We are continuously exploring better principles for FDSL
- The framework is not limited to fractal geometry, and can employ any principles to generate labeled images

Future direction (4/4)

Constructing foundation models with generative pre-training

Main focuses — Collaborative

Generative Models

- Model
- Data
- Label

Data & Labels from generative models / mathematical formula

Pretrain → Pre-trained Models

Resource: Super Computers

{Img,Video,3D,Audio,Text}

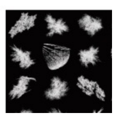
Adaptation

Additional training via fine-tuning while inserting domain knowledge

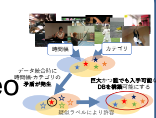
- Various fields
- FT with limited data
- Improved performance

Modality

Images
Images/labels are generated from diffusion models or formulas



Videos
Videos/labels are generated with video generative models



Audio
1D signal/labels are generated. The 1D signal is like a noise generation

Texts
Language models are constructed from a word probability / language models

The concept relates to…

Data → Generative Modeling → Data++ → Learner → f

Phillip Isola (MIT)
https://www.youtube.com/watch?v=YuRAeQsTSo8

Three general approaches to employ generative models.

1. To solve the task directly
2. As priors
3. To generate training data

Christian Rupprecht (Univ. of Oxford)
https://www.youtube.com/watch?v=HUyP2C2rYto

Our goal is to improve FDSL to potentially replace the pre-trained model done with real images and human annotations, addressing concerns around ethical and annotation issues

Thank you.