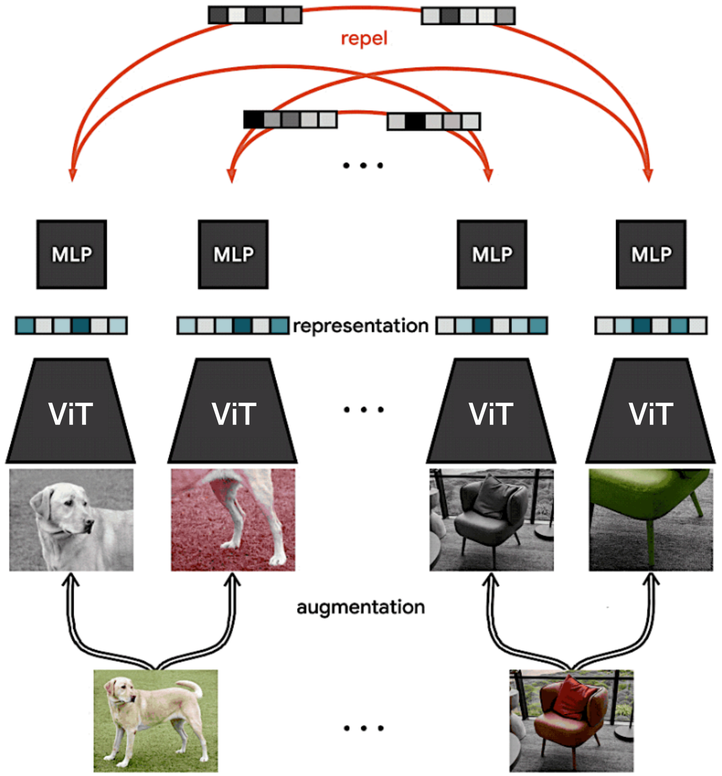
WHAT DO SELF-SUPERVISED VISION TRANSFORMERS LEARN?

NAMUK PARK, WONJAE KIM, BYEONGHO HEO,
TAEKYUNG KIM, SANGDOO YUN

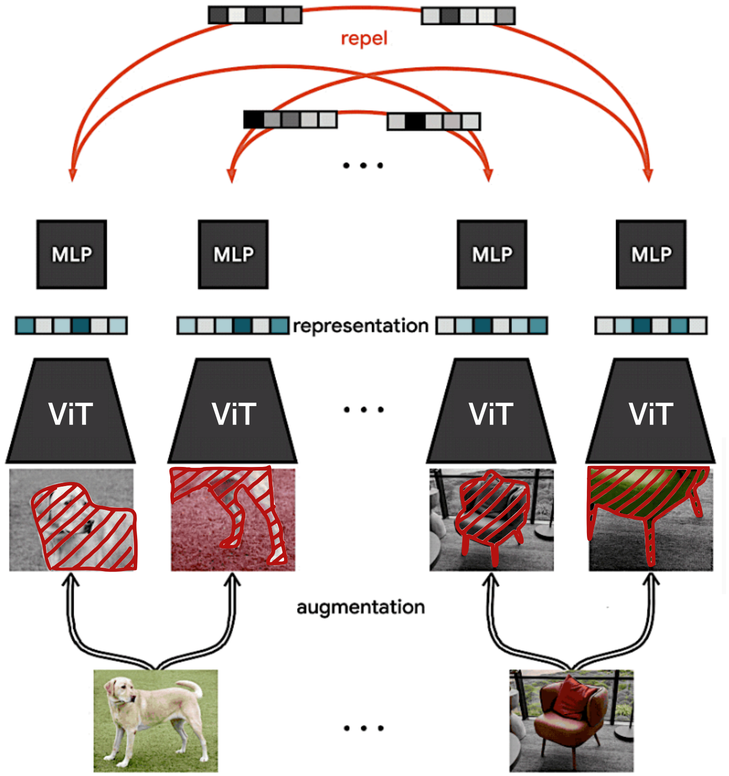
Contrastive Learning (CL) Is Image-Level Approach



CL aims to learn the invariant semantics for two random views by making global projections.
CL can be deemed as an "*image-level*" self-supervised learning approach.

Official implementation of SimCLR (https://github.com/google-research/simclr)

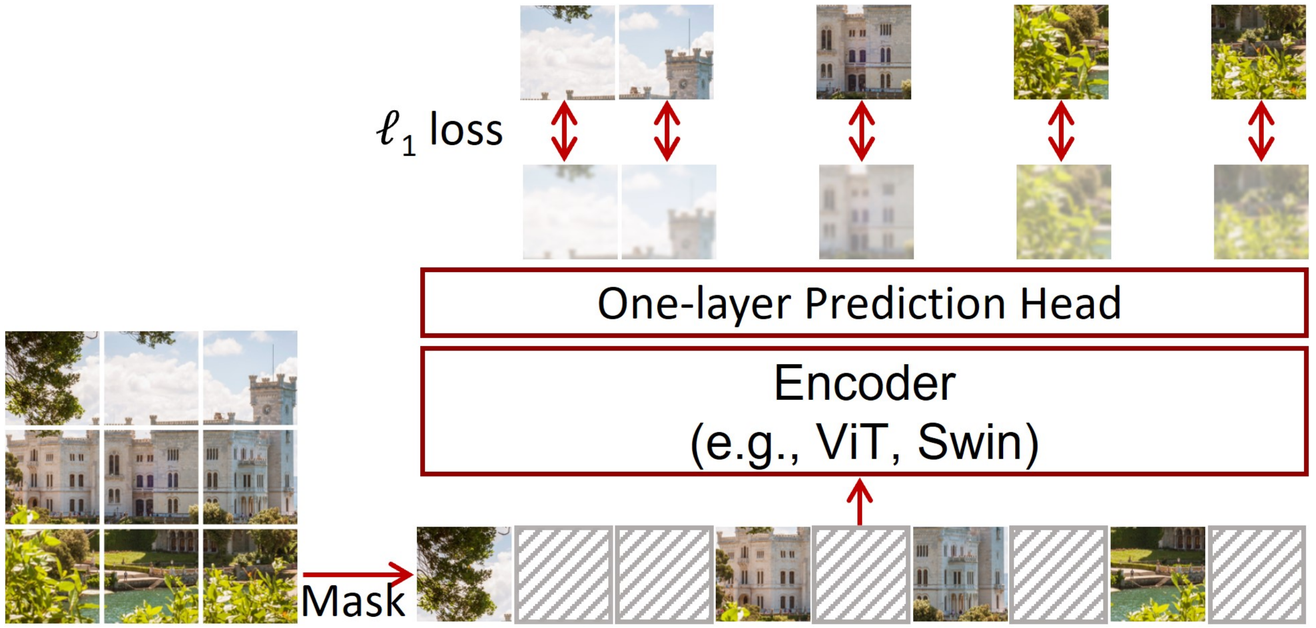
Contrastive Learning (CL) Is Image-Level Approach



CL aims to learn the invariant semantics for two random views by making global projections.
CL can be deemed as an "*image-level*" self-supervised learning approach.

Official implementation of SimCLR (https://github.com/google-research/simclr)

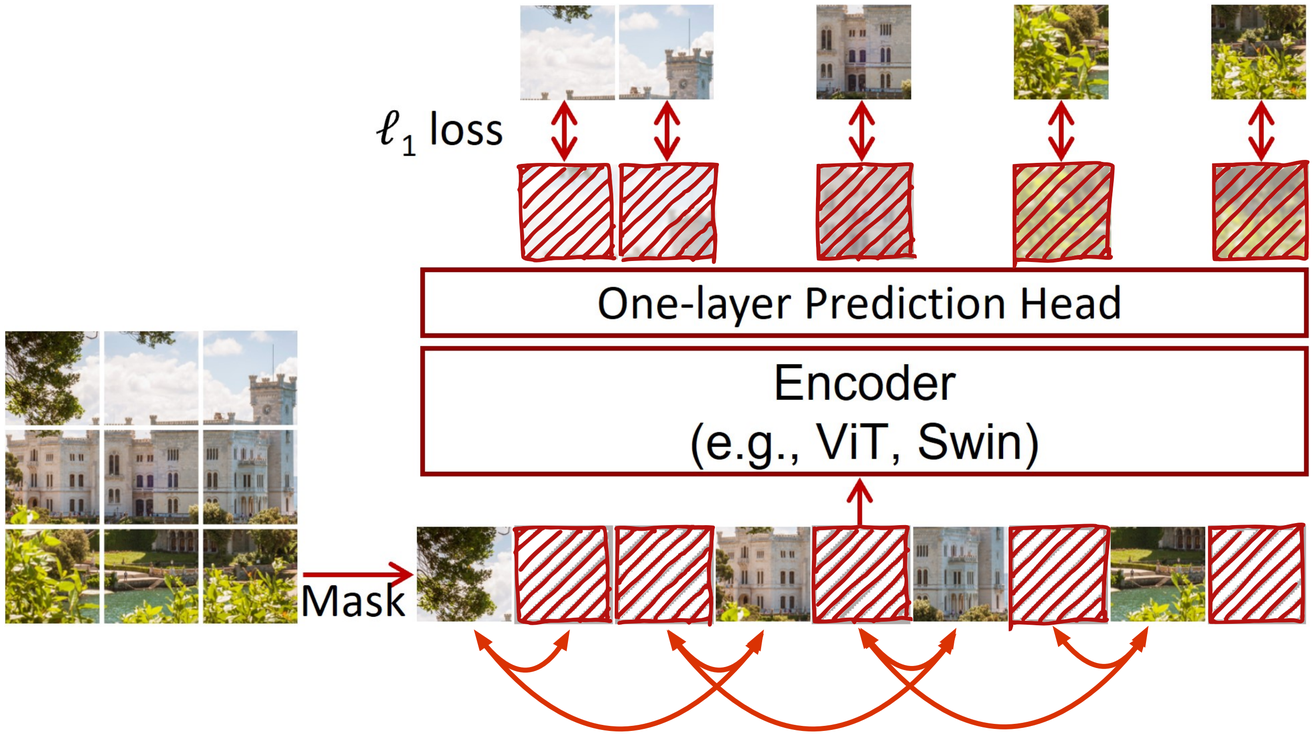
Masked Image Modeling (MIM) Is Token-Level Approach



MIM trains ViTs by reconstructing the correct semantics of masked input patches.
Since it learns semantics of patch tokens, it can be deemed as "*token-level*" self-supervised learning.

Xie, Zhenda, et al. "Simmim: A simple framework for masked image modeling." CVPR (2022).

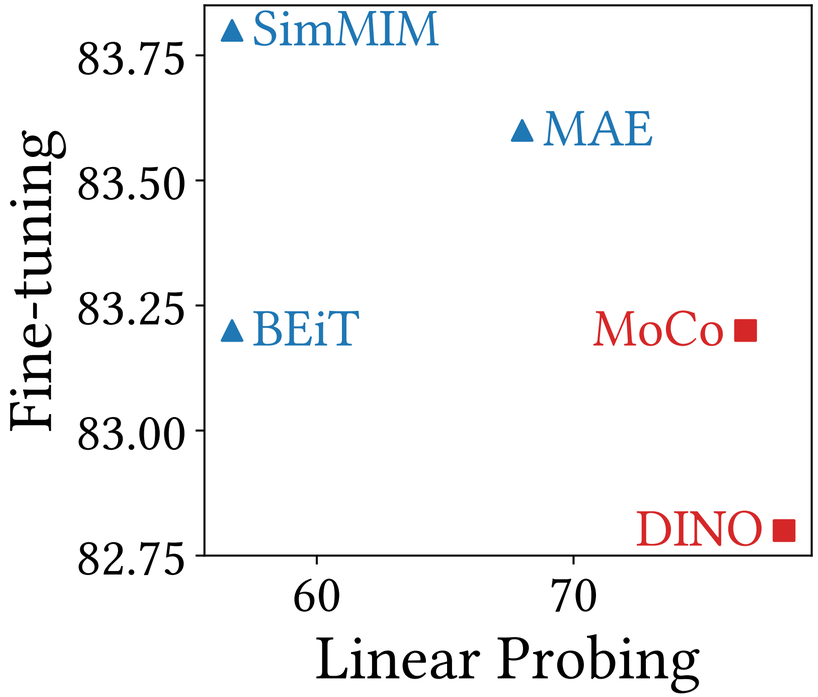
Masked Image Modeling (MIM) Is Token-Level Approach



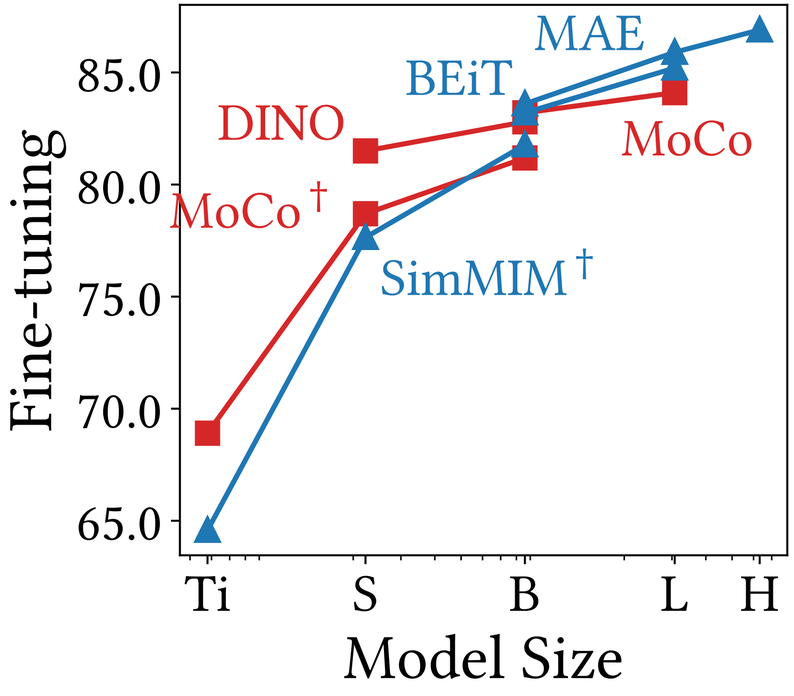
MIM trains ViTs by reconstructing the correct semantics of masked input patches.
Since it learns semantics of patch tokens, it can be deemed as "*token-level*" self-supervised learning.

Xie, Zhenda, et al. "Simmim: A simple framework for masked image modeling." CVPR (2022).

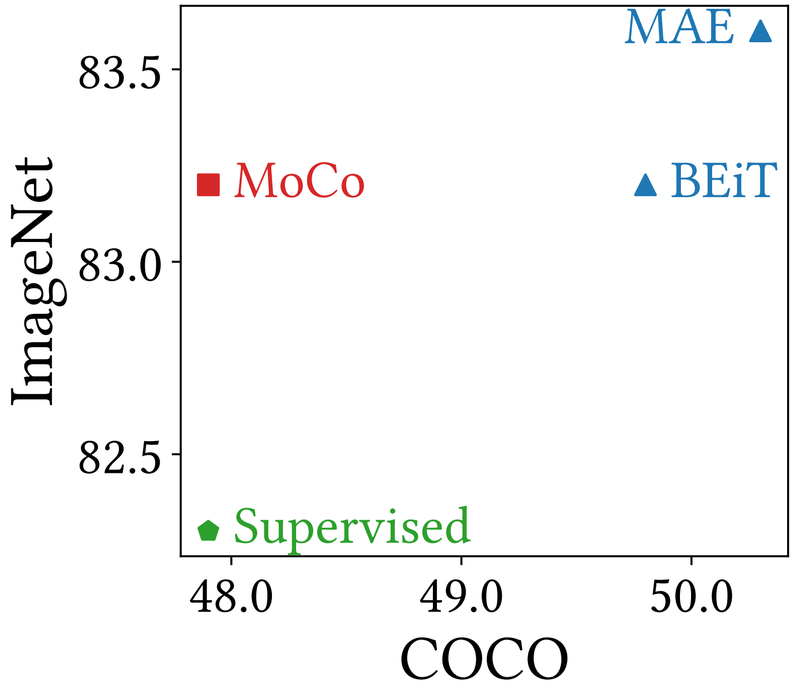
CL and MIM May Not Be a Silver Bullet for All Tasks



In **linear probing accuracy**, contrast learning (CL) outperforms masked image modeling (MIM), but underperforms in fine-tuning accuracy (w/ ViT-B on IN1K).



In **small model regimes**, CL outperforms MIM. For large models, MIMs outperform CLs (on IN1K).



In **classification tasks**, CL works well. In dense prediction tasks, MIM outperforms CL (w/ ViT-B).

MoCo means MoCo-v3.
† Dagger mean the reproduced results w/ 100 epoch pre-training.

Summary: MIM and CL Are Complementary

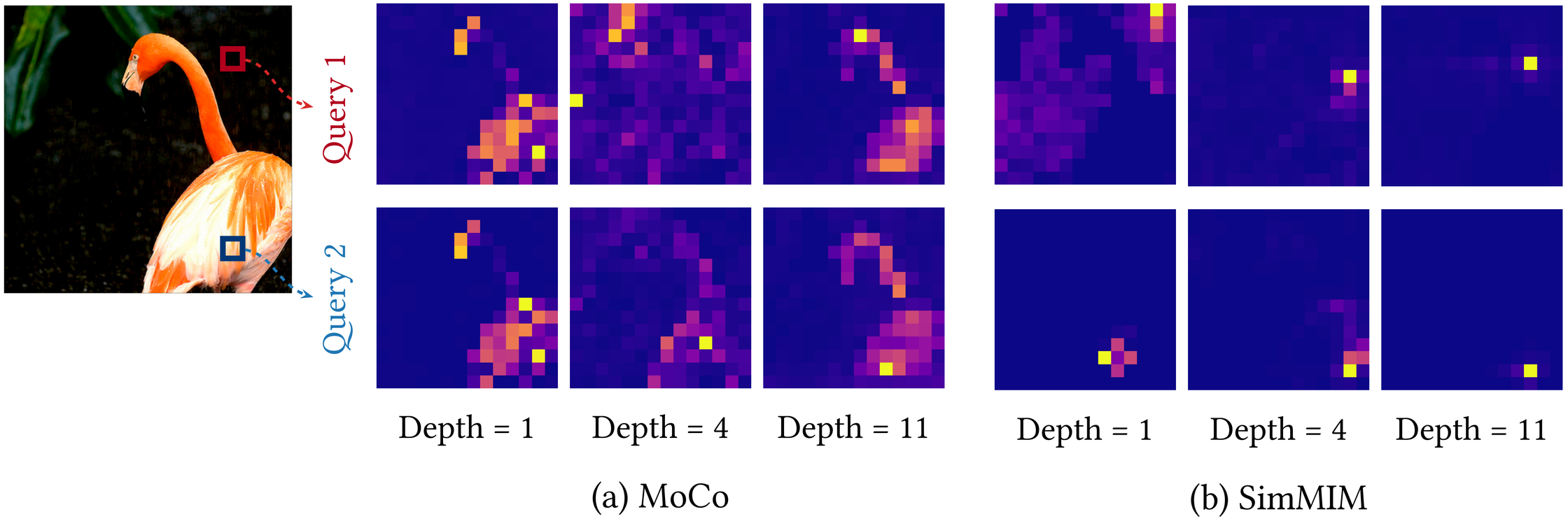
- **How do *self-attentions* behave?** The self-attentions of CL capture *global* relationships and that of MIM capture *local* relationships. Therefore, CL can recognize objects well, but it struggle with preserving local information.

- **How are *representations* transformed?** CL makes images linearly separable, but it has difficulty in distinguishing tokens, compared with MIM. Moreover, ViT learns *low-frequency information* from *CL*, and *high-frequency information* from *MIM*. Therefore, CL is texture-biased whereas MIM is texture-biased.

- **Which *components* play an important role?** Self-attentions in CL and MLPs in MIM are important. Moreover, CL plays a crucial role in the *later layers* of ViT architecture, while MIM mainly focuses on the *early layers*.

Q1. How Do Self-Attentions Behave?

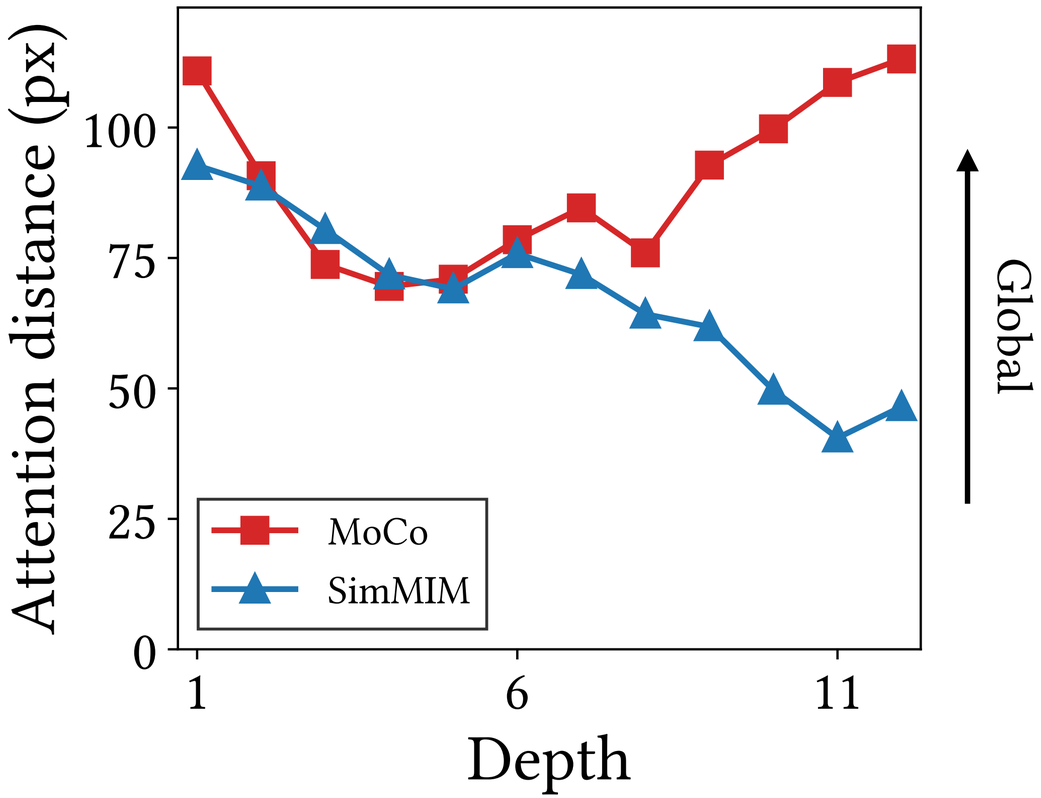
Intriguing Properties of CL's Self-Attentions



(a) MoCo (b) SimMIM

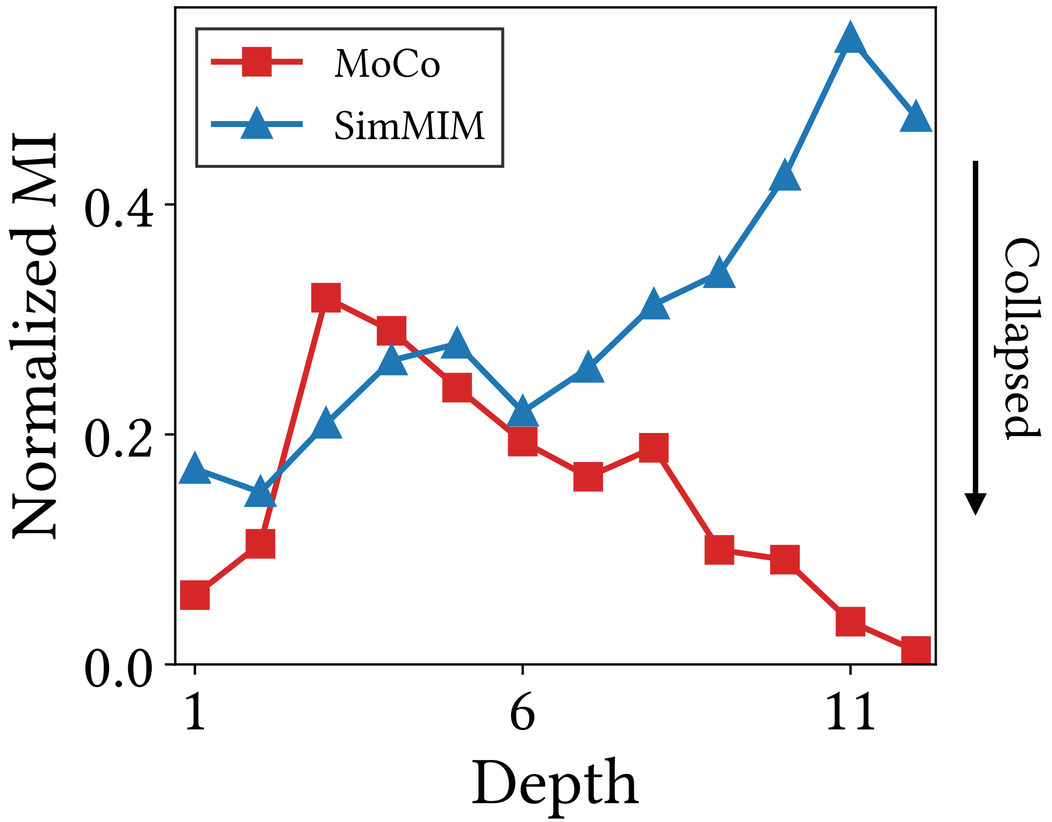
Self-attentions of CL (MoCo) capture **global relationships**, but they **collapse into homogeneous attention maps** for all heads, depths, and query tokens. Self-attentions of MIM (SimMIM) focus on local areas.

CL Mainly Captures Global Relations, but MIM Does Not



The effective receptive fields of CL (MoCo) are global, but those of MIM (SimMIM) are local.
Data points represent the attention distance of heads as analogous to the size of the receptive field.

Self-Attentions of CL Collapse Into Homogeneity

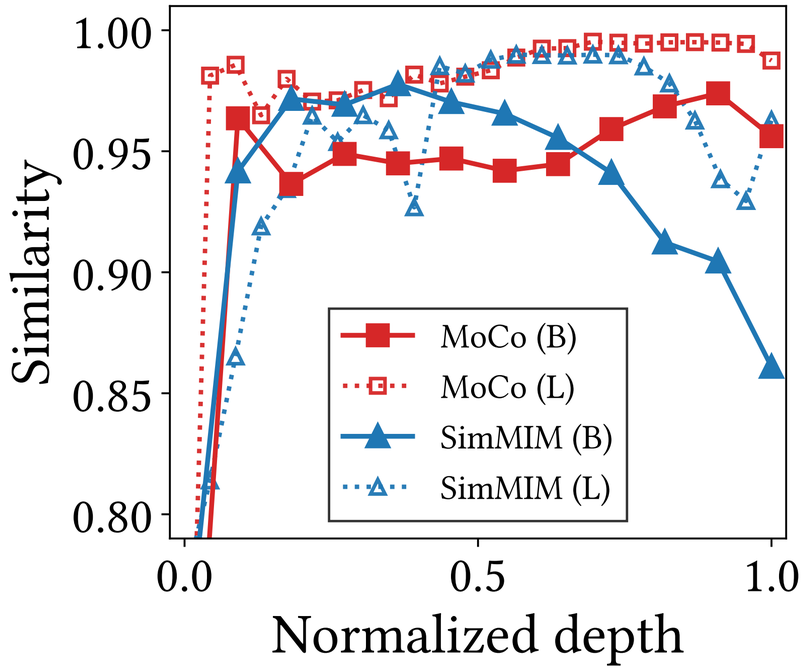


Self-attentions of CL have little to do with query tokens. We use normalized mutual information to measure the *correlation between self-attention maps and queries*. The low mutual information value represents the attention map is less dependent on the query tokens implying the attention collapse into homogeneity.

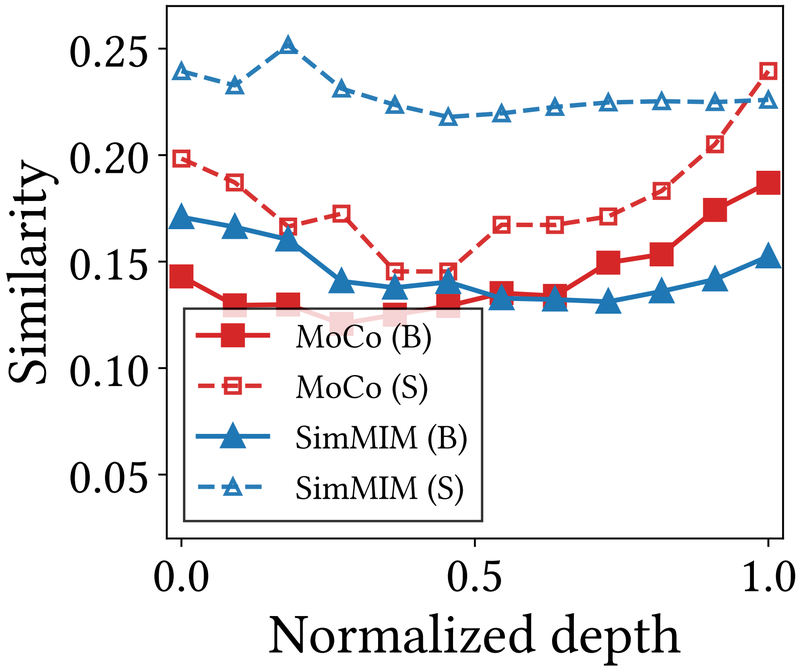
$$\frac{I(q,k)}{\sqrt{H(q)\,H(k)}}$$

I (q,k): mutual information
H(q): entropy

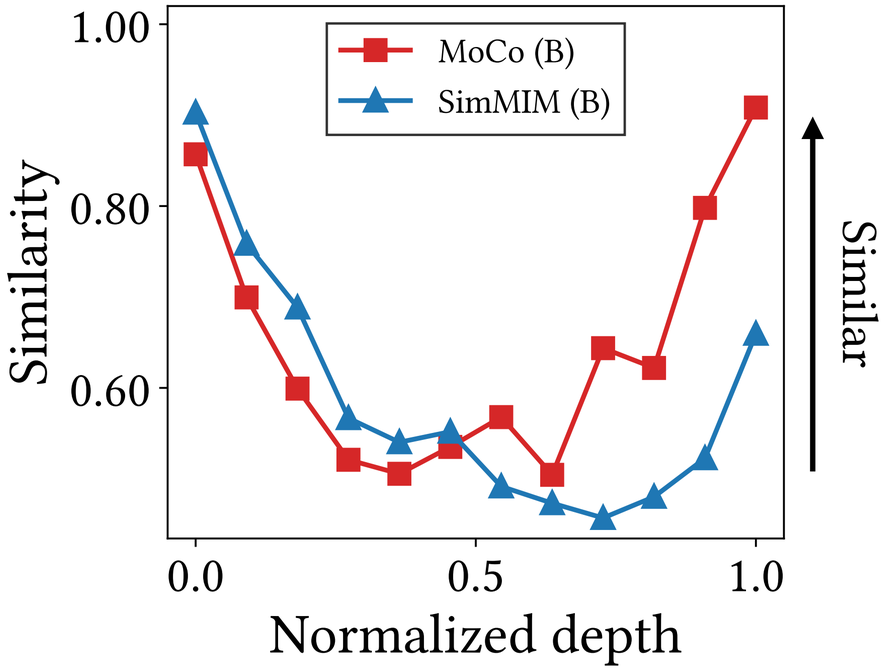
The Attention Collapse Reduces the Diversity



(a) head similarity



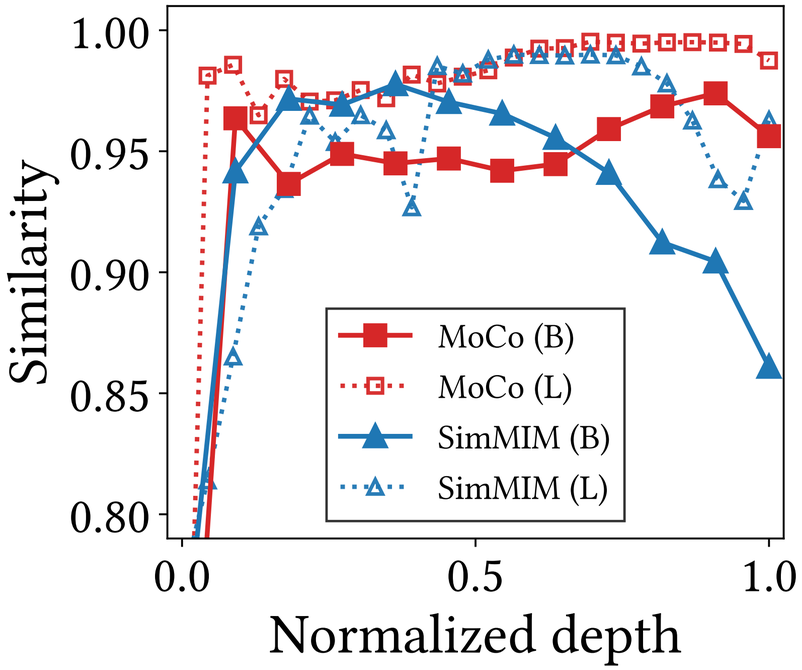
(b) layer similarity



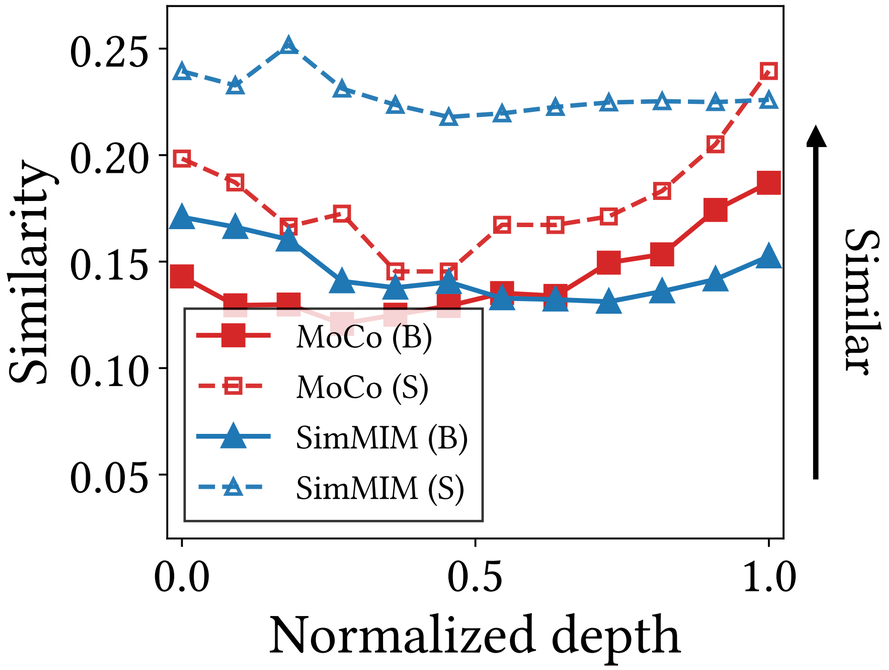
(c) spatial similarity

CL lacks representational diversities, especially at the end of the models**.** We measure similarities of representations in self- attentions by using mini-batch CKA between the heads (left), depths (middle), and spatial coordinates (right).

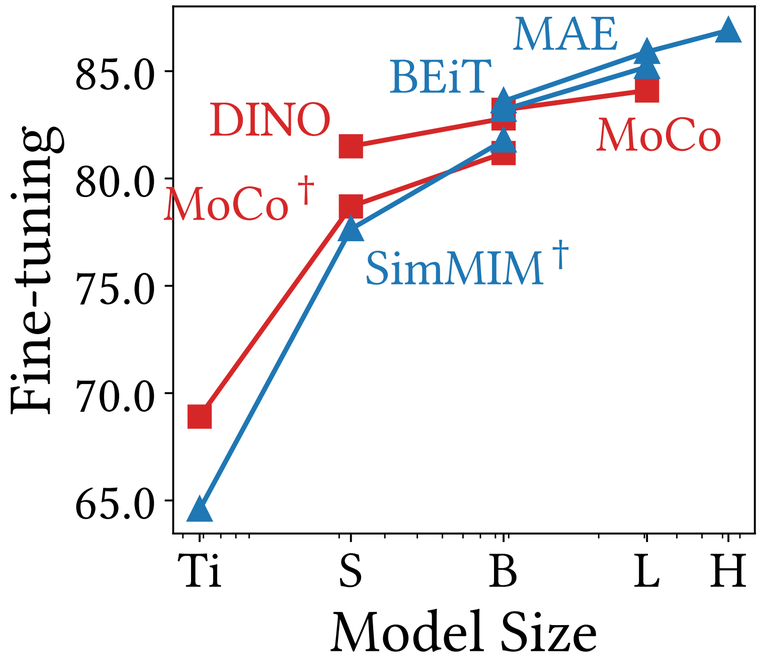
The Attention Collapse Reduces the Diversity



(a) head similarity

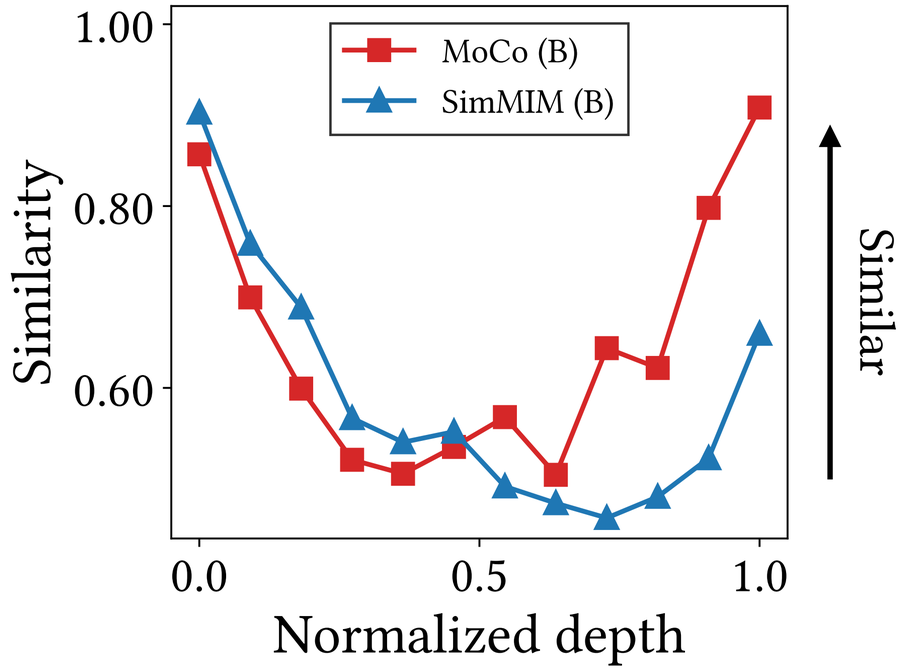


(b) layer similarity

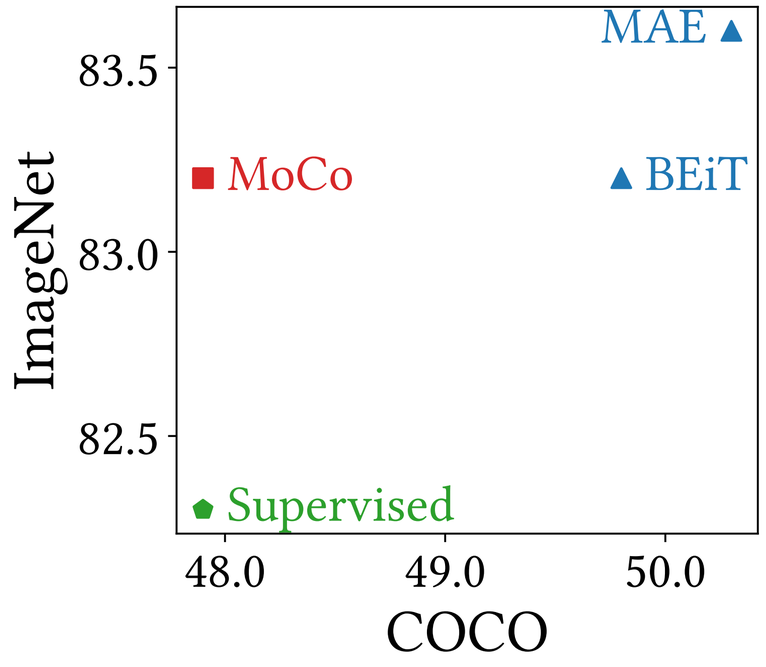


CL lacks representational diversities, especially at the end of the models. We measure similarities of representations in self- attentions by using mini-batch CKA between the heads (left), depths (middle), and spatial coordinates (right).

The Attention Collapse Reduces the Diversity



(c) spatial similarity



CL lacks representational diversities, especially at the end of the models**.** We measure similarities of representations in self- attentions by using mini-batch CKA between the heads (left), depths (middle), and spatial coordinates (right).

Why Does Contrastive Learning Work That Way?

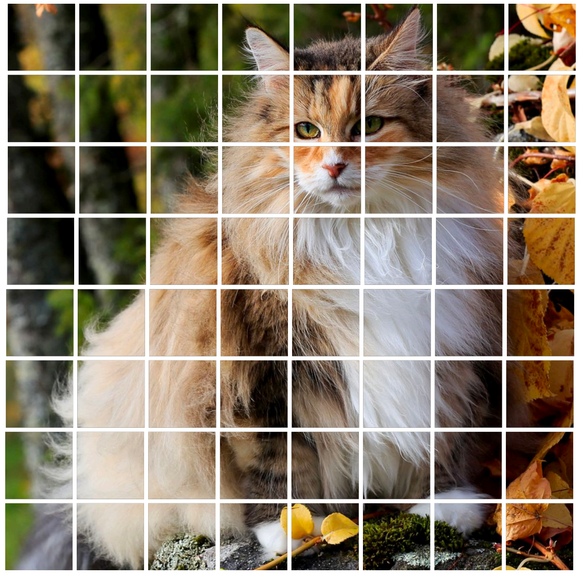
- **Linear probing**: Capturing shapes by CL helps recognize objects and distinguish images, so CL outperforms MIM in linear probing tasks. Although MIM preserves texture and the diversity of representation, these are not strongly correlated with objects or content, compared to shapes.

- **Scalability**: The attention collapse prohibits CL from fully exploiting heads, depths, and tokens of ViTs. ViTs trained with CL wastes a large part of network capability due to the attention collapse.

- **Dense prediction**: CL is not suitable for dense prediction since the token features are homogeneous with respect to their spatial coordinates.

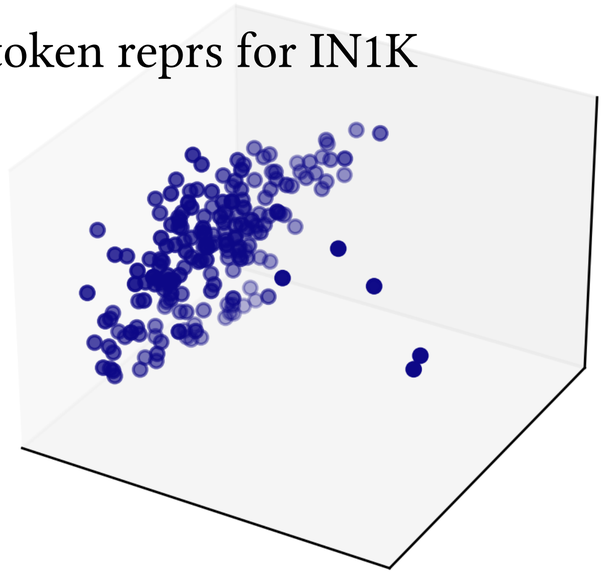
Q2. How Are Representations Transformed?

Two Perspectives on Interpreting Representations



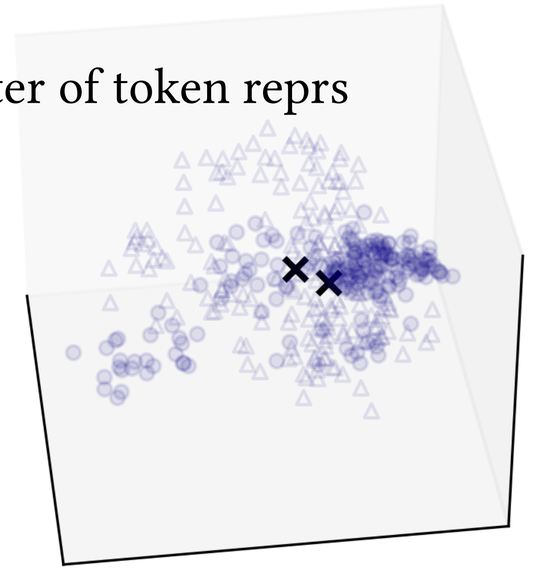
We analyze the representation from two perspectives: one for tokens and one for images.

196 token reprs for IN1K



The ***"token-level"* perspective** focuses on how tokens from "*a single image*" interact with each other.

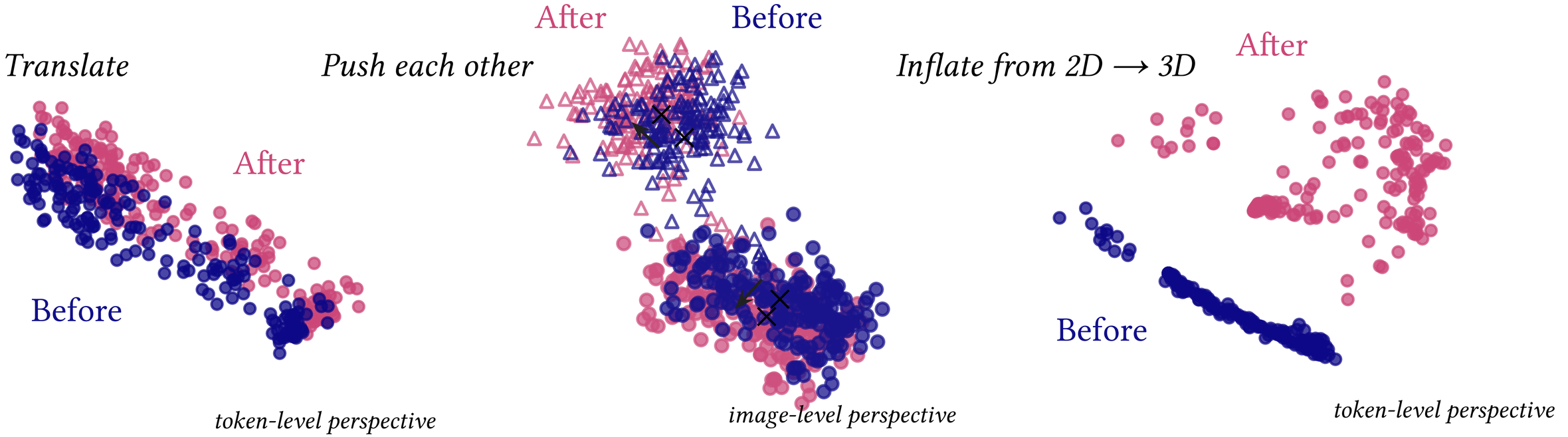
center of token reprs



The ***"image-level"* perspective** focuses on how tokens from "*multiple images*" interact with each other. To this end, we use GAP to derive the center of representations from a single image.

○, △: tokens from a single image (they are *not* images)
×: center of a image reprs (i.e., GAP of tokens)

CL and MIM Transform Representations Differently



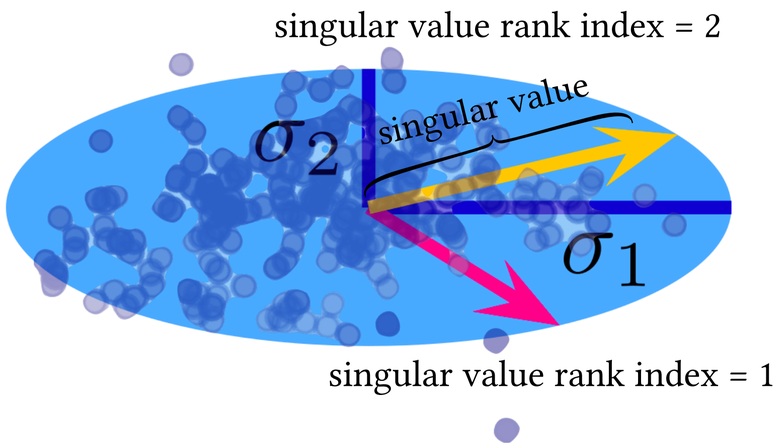
In **MoCo**, self-attentions simply translate ($z \to z + c$) all tokens equally (due to attention collapse). That is, they do *not* encode an image.

But **MoCo** makes "the center of different image tokens" move away from each other $\Longrightarrow$ *linearly separable*.

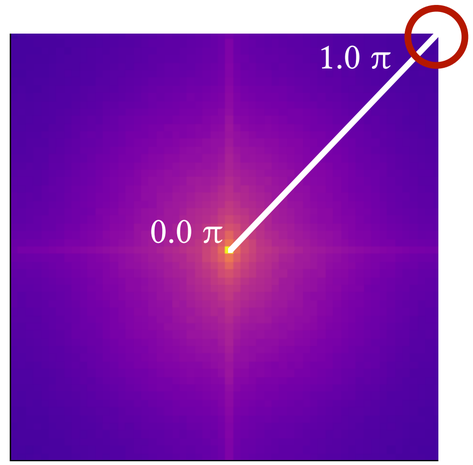
In **SimMIM**, self-attentions transform the token depending on the token, .i.e., they *encode* an image and inflate the token volume (2D ⟶ 3D in this case).

○, △: tokens from a single image (they are *not* images)
×: center of a image reprs (i.e., GAP of tokens)

Two Quantitative Methods for Investigation



Singular Value Decomposition (SVD) measures the *effective volume* occupied by tokens/images. If NN increases the volume (singular value), it implies that NN processes the tokens and transform the geometries.

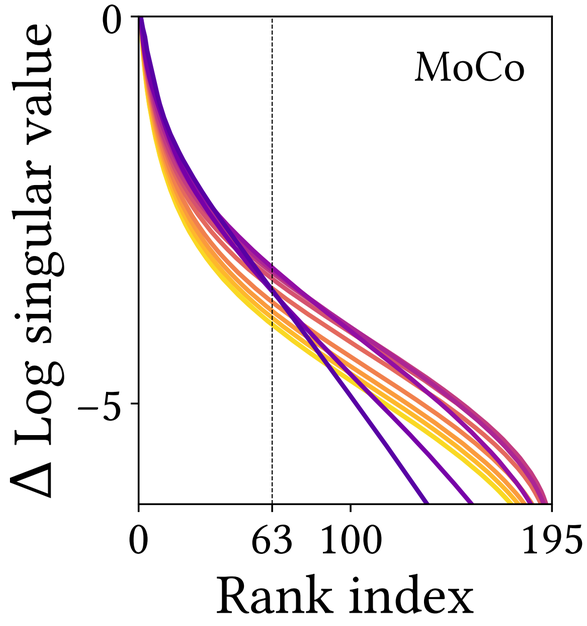


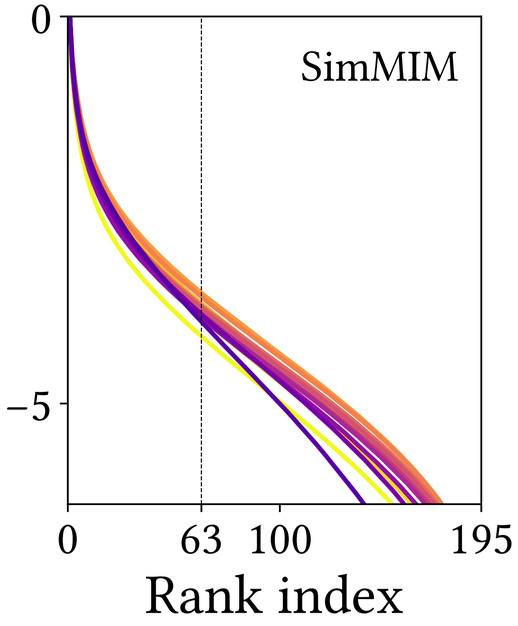
high-frequency information

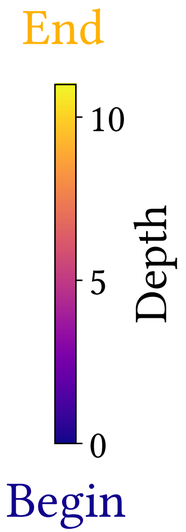
Fourier analysis of representation shows whether NN captures high frequency information (texture) or low frequency information (shape). To do so, we only report the high frequency amplitude.

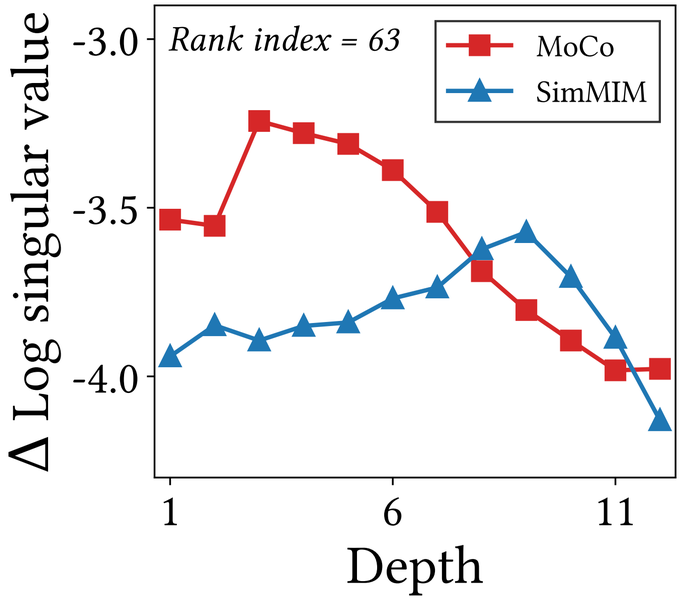
CL and MIM Transform Representations Differently

SVD analysis: token-level





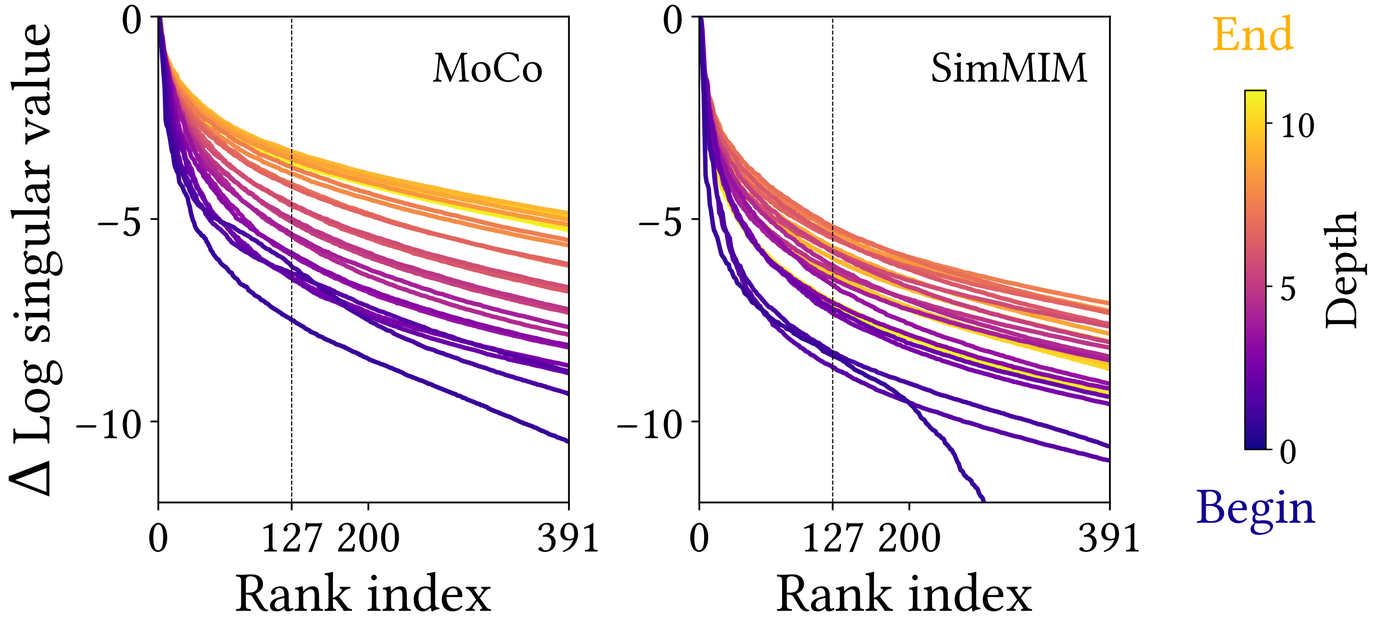


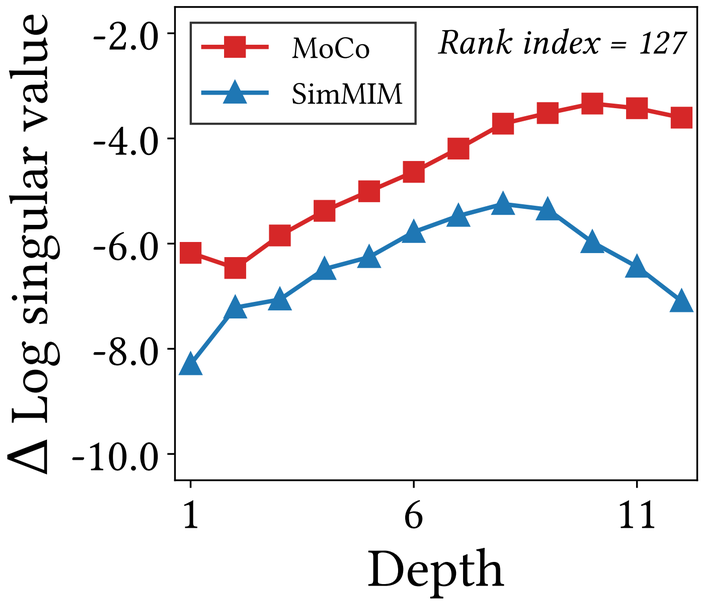


CL almost does not increase the volume of the tokens (from a single image) or even decrease it, but MIM does. It implies that CL has difficulty in distinguishing tokens.

CL and MIM Transform Representations Differently

SVD analysis: image-level



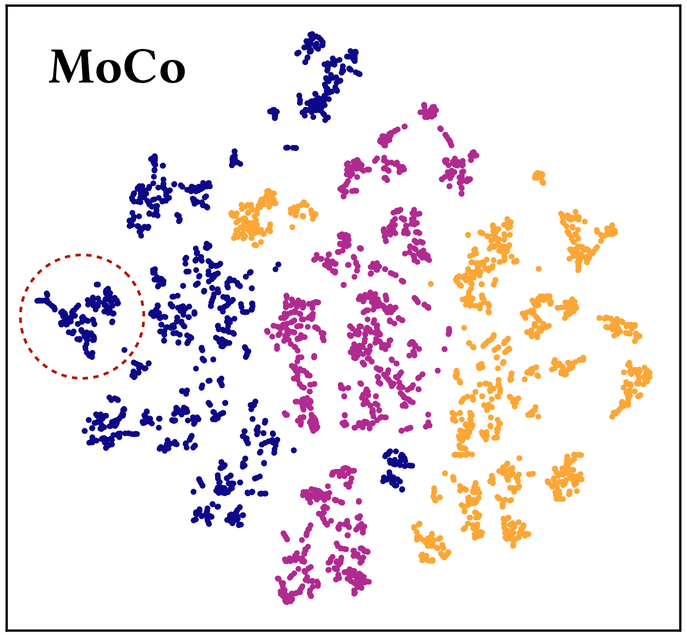


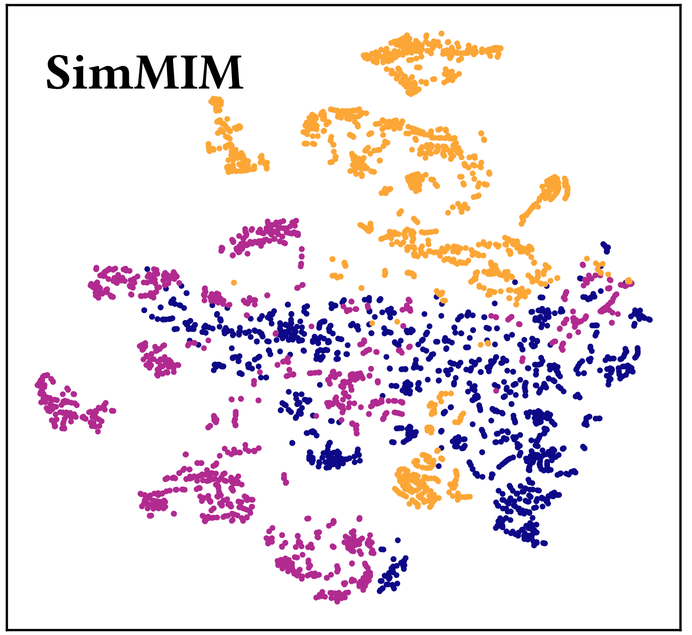
CL significantly increases the volumes occupied by images, but MIM hardly increase the volume.
It implies that MoCo distinguish images

CL and MIM Transform Representations Differently

Token-level t-SNE visualization

tokens in
a single image

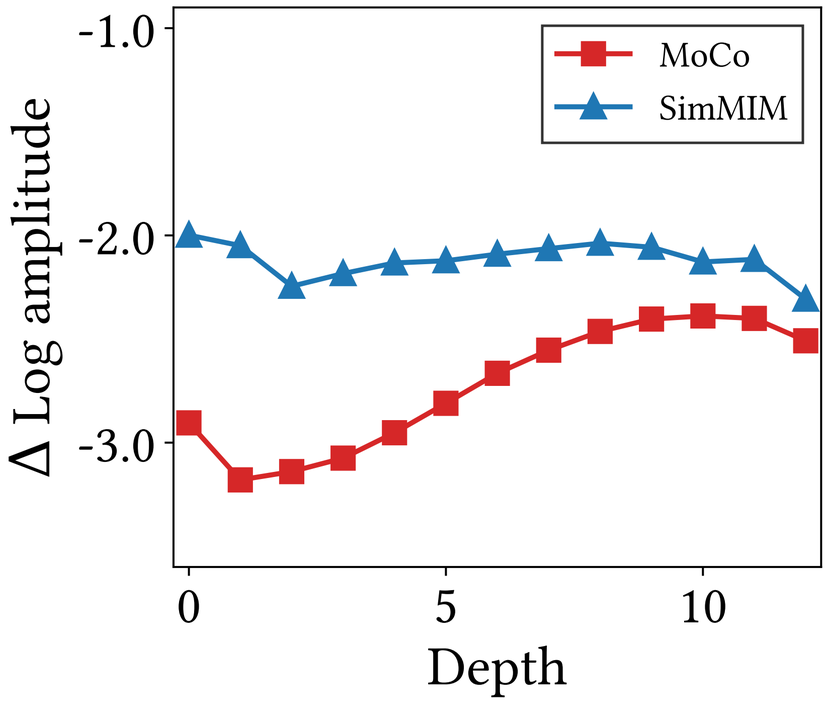




CL significantly increases the volumes occupied by images, but MIM hardly increase the volume.
It also implies that MoCo distinguishes images well, while SimMIM intermingles tokens.

CL and MIM Transform Representations Differently

Fourier analysis

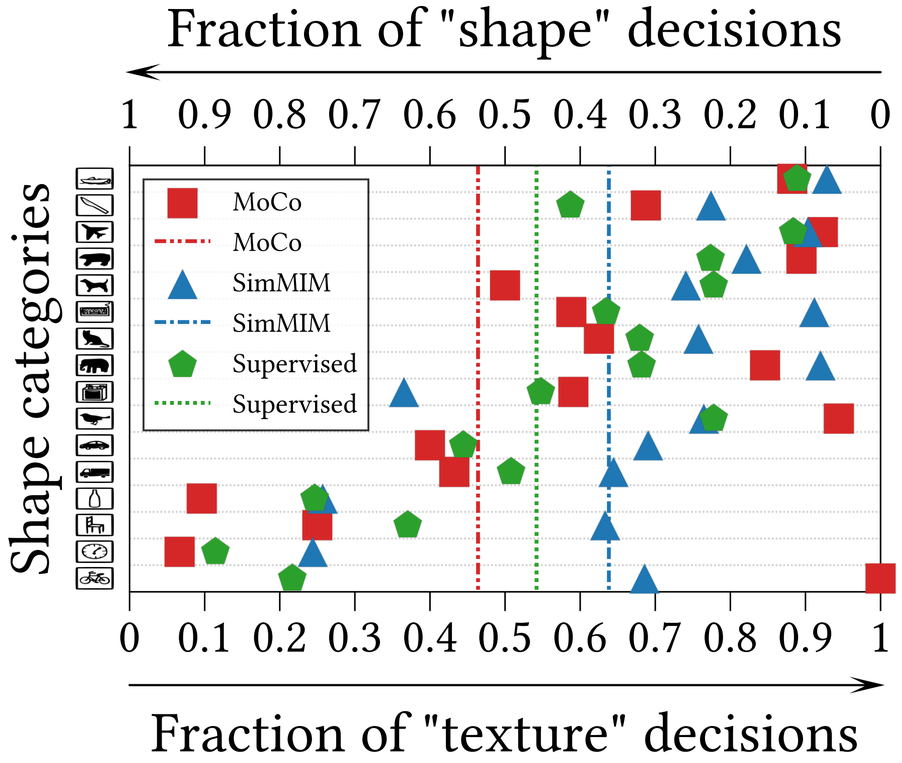


High-frequency of representation in terms of the amplitude difference between the representations' highest and lowest frequencies. It shows that **MoCo captures low-frequencies of representations, but SimMIM captures high-frequencies.**

Low-frequencies correspond to shapes.

High-frequencies correspond to texture.

CL Is Shape-Biased Whereas MIM Is Texture-Biased



(a) Texture image
81.4% Indian elephant
10.3% indri
8.2% black swan

(b) Content image
71.1% tabby cat
17.3% grey fox
3.3% Siamese cat

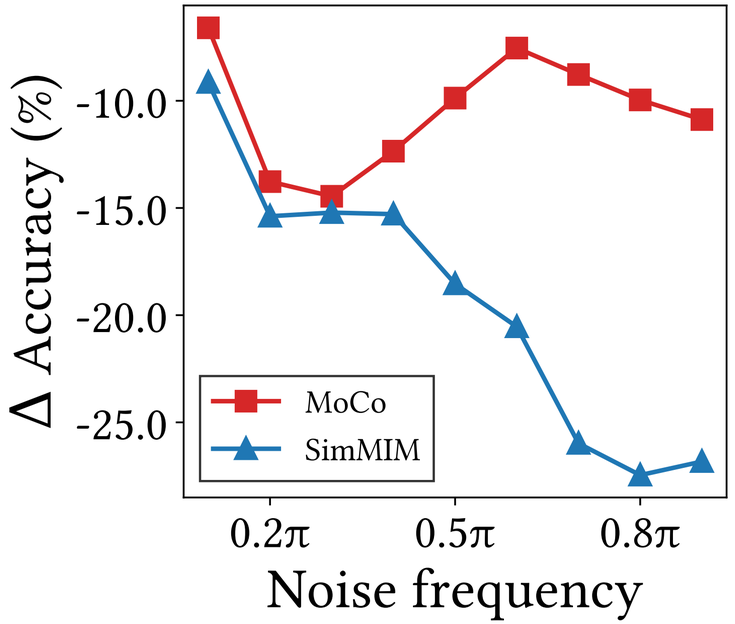
(c) Texture-shape cue conflict
63.9% Indian elephant
26.4% indri
9.6% black swan

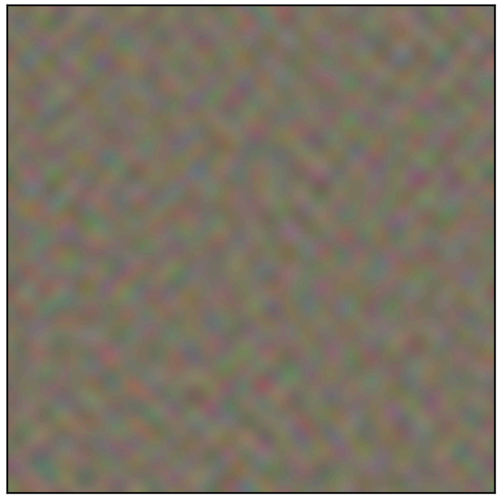
Stylized ImageNet

Left: The classification results on Stylized ImageNet shows that CL is more shape-biased than MIM and even than the supervised pre-trained model. The vertical lines represent the averaged results for the shape categories. *Right*: A sample of Stylized ImageNet.

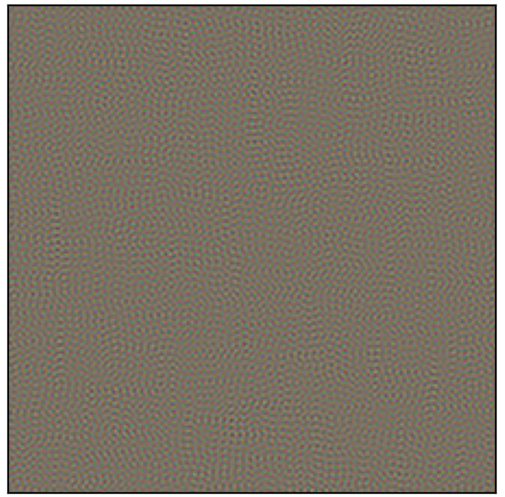
Geirhos, Robert, et al. "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness." ICLR (2019).

CL Is Shape-Biased Whereas MIM Is Shape-Biased





low-frequency noise

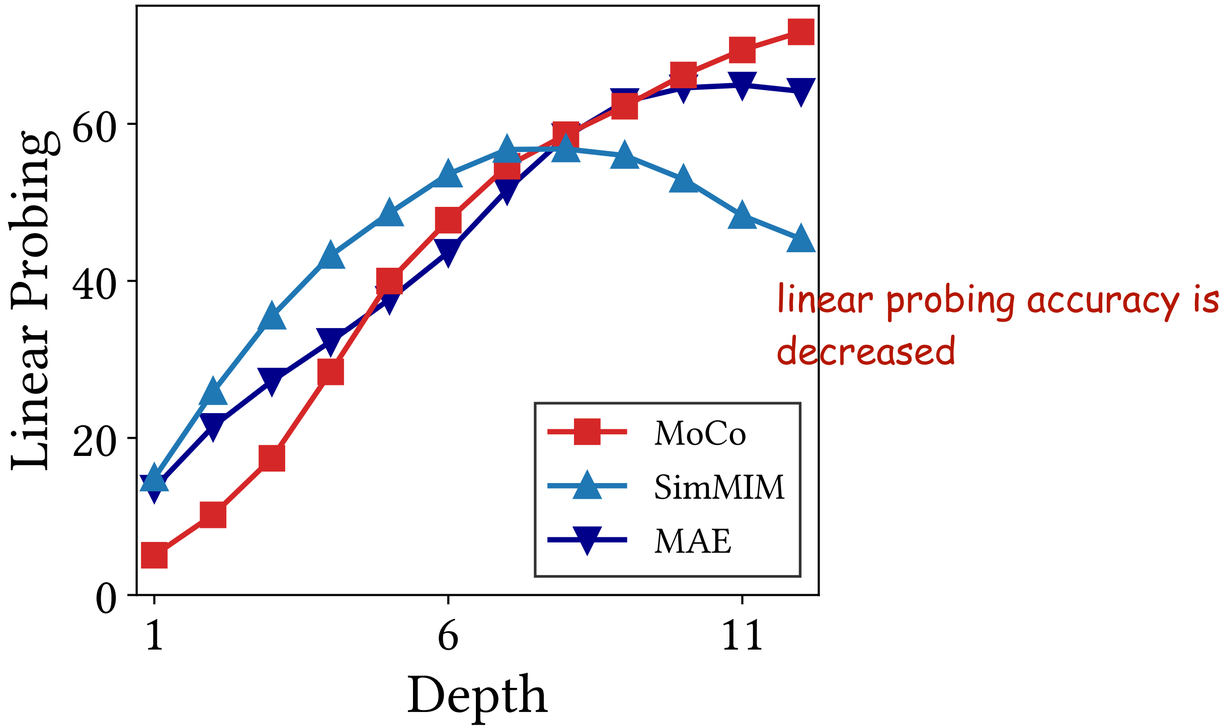


high-frequency noise

CL is vulnerable against high-frequency noise, but MIM is robust.
It also demonstrates MIM's texture-biased property compared to CL.

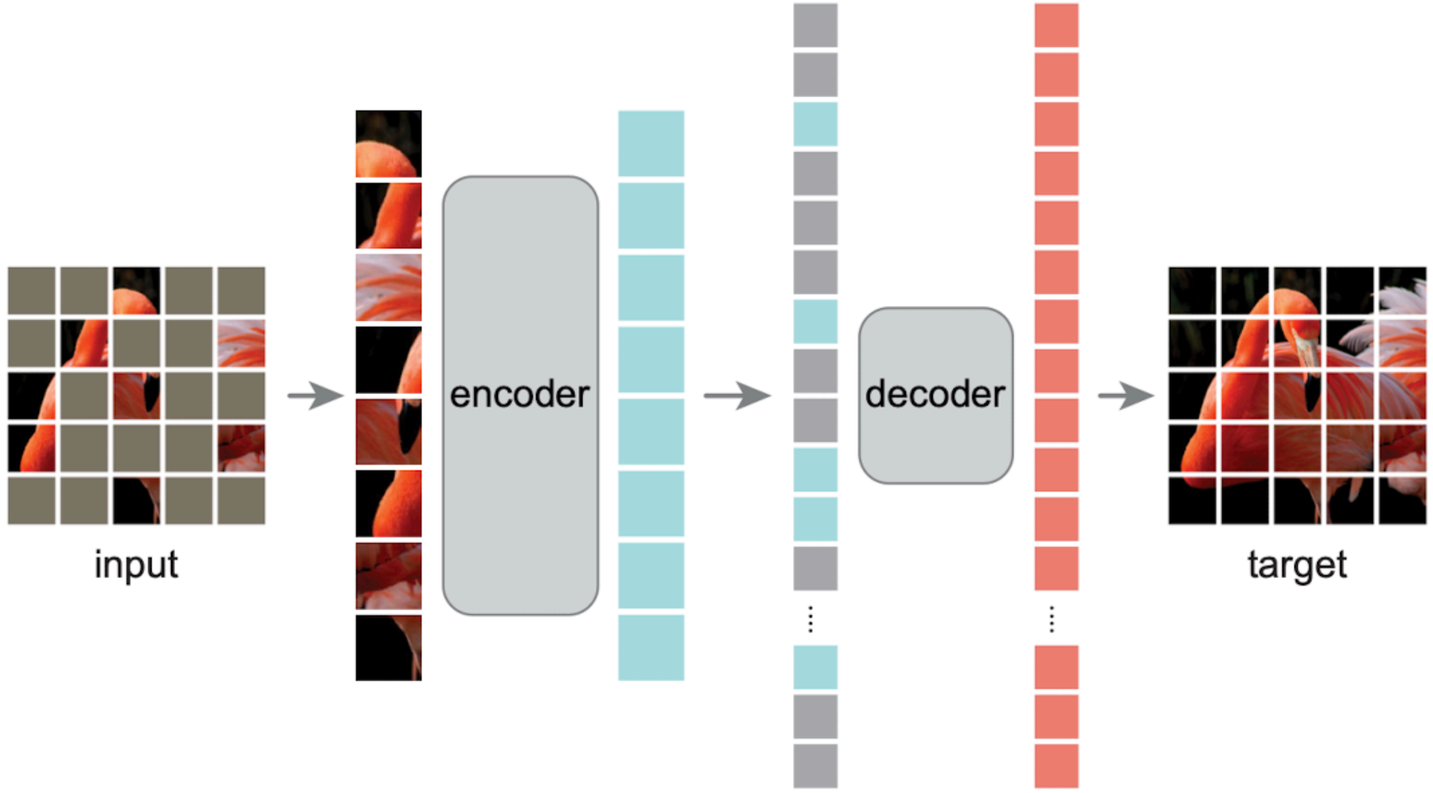
Q3. Which Components Play an Important Role?

Later Layers of CL and Early Layers of MIM Are Important



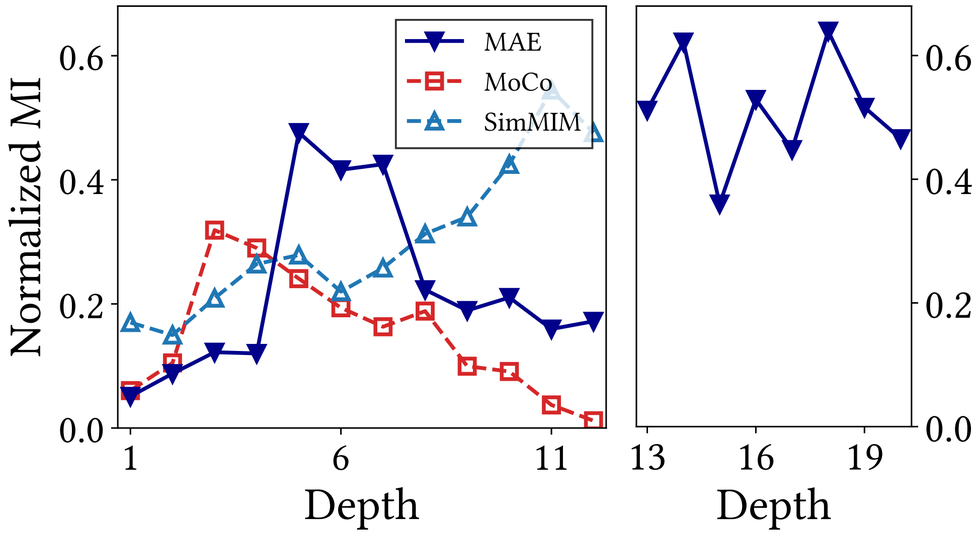
Later layers of CL and early layers of MIM play a key role. We report linear probing accuracies by using representations of the intermediate layer. CL outperforms MIM in later layers, and MIM outperforms CL in early layers.

Key Ideas of Masked Autoencoder (MAE)

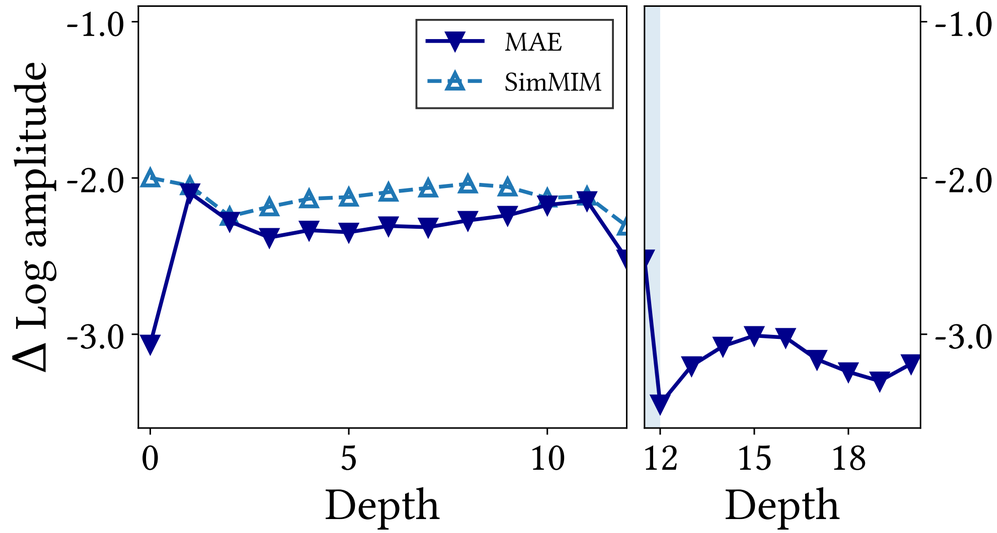


Masked autoencoder (MAE) introduces the following two ideas:
(1) Use separate decoder for MIM. (2) Encode only unmasked tokens.

MAE Helps ViTs Fully Leverage MIM



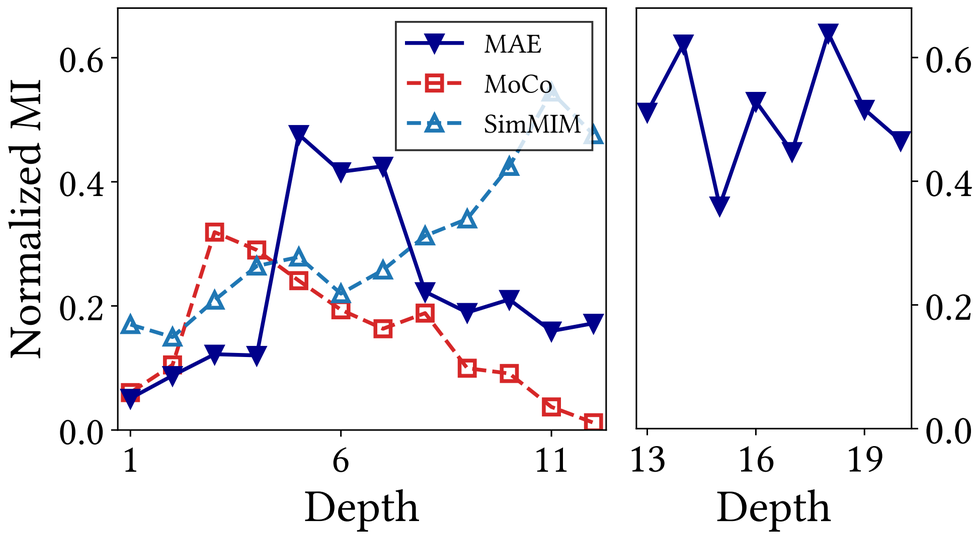
(a) Self-attention



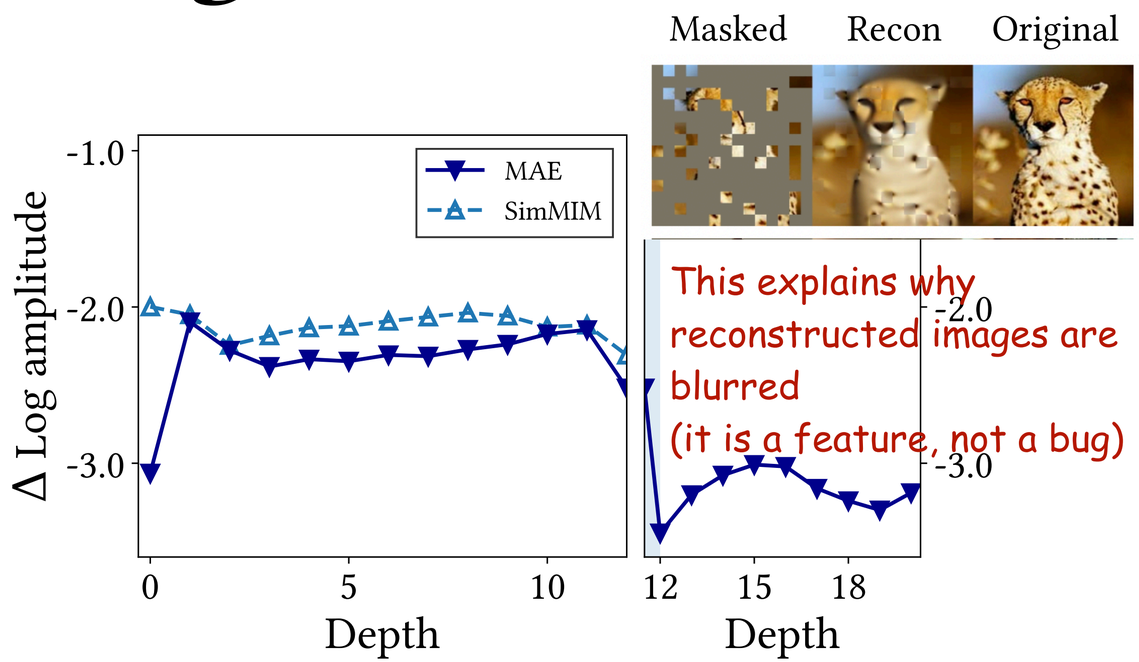
(b) Fourier analysis

Left: Fourier analysis shows that decoder captures low-frequency information. Therefore, MIM backbone combined with a decoder utilizes higher frequency information. *Right*: In contrast to SimMIM, the magnitude of the displacement of MIM does not decrease with increasing depth. Moreover, unlike the backbone, self-attention modules dominates the change of representation in the decoder.

MAE Helps ViTs Fully Leverage MIM



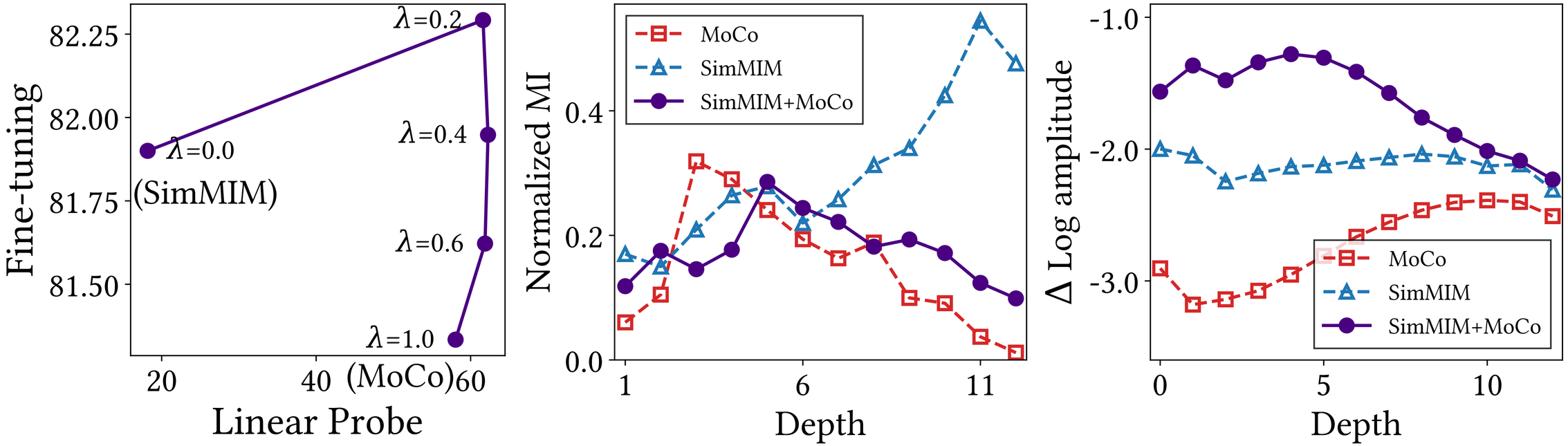
(a) Self-attention



(b) Fourier analysis

Left: Fourier analysis shows that decoder captures low-frequency information. Therefore, MIM backbone combined with a decoder utilizes higher frequency information. *Right*: In contrast to SimMIM, the magnitude of the displacement of MIM does not decrease with increasing depth. Moreover, unlike the backbone, self-attention modules dominates the change of representation in the decoder.

CL and MIM Are Complementary



(a) Performance (b) Self-attention (c) Fourier analysis

The simple linear combination of CL and MIM loss $\mathscr{L} = (1-\lambda)\mathscr{L}_{MIM} + \lambda\mathscr{L}_{CL}$ leverages the advantages, implying that CL and MIM are complementary. *Left*: "CL + MIM" outperforms CL and MIM in both linear probing and fine-tuning accuracy. *Middle*: Only the self-attentions of later layers collapse into homogeneity and capture the same object shape information. *Right*: "CL + MIM" exploits high-frequency at the beginning and low-frequency at the end.

	Contrastive Learning *learns image-level invariants*	**Masked Image Modeling** *learns token-level similarities*
BEHAVIOUR	Linear probing & small model	Fine tuning & large model
SELF-ATTENTION	Capture globalities & shapes	Capture localities & textures
REPRESENTATION	Distinguish images	Distinguish tokens
ARCHITECTHRE	Focus on later layers	Focus on early layers