Context Autoencoder for Self-Supervised Representation Learning

ICLR 2023

2023.06.27

目录

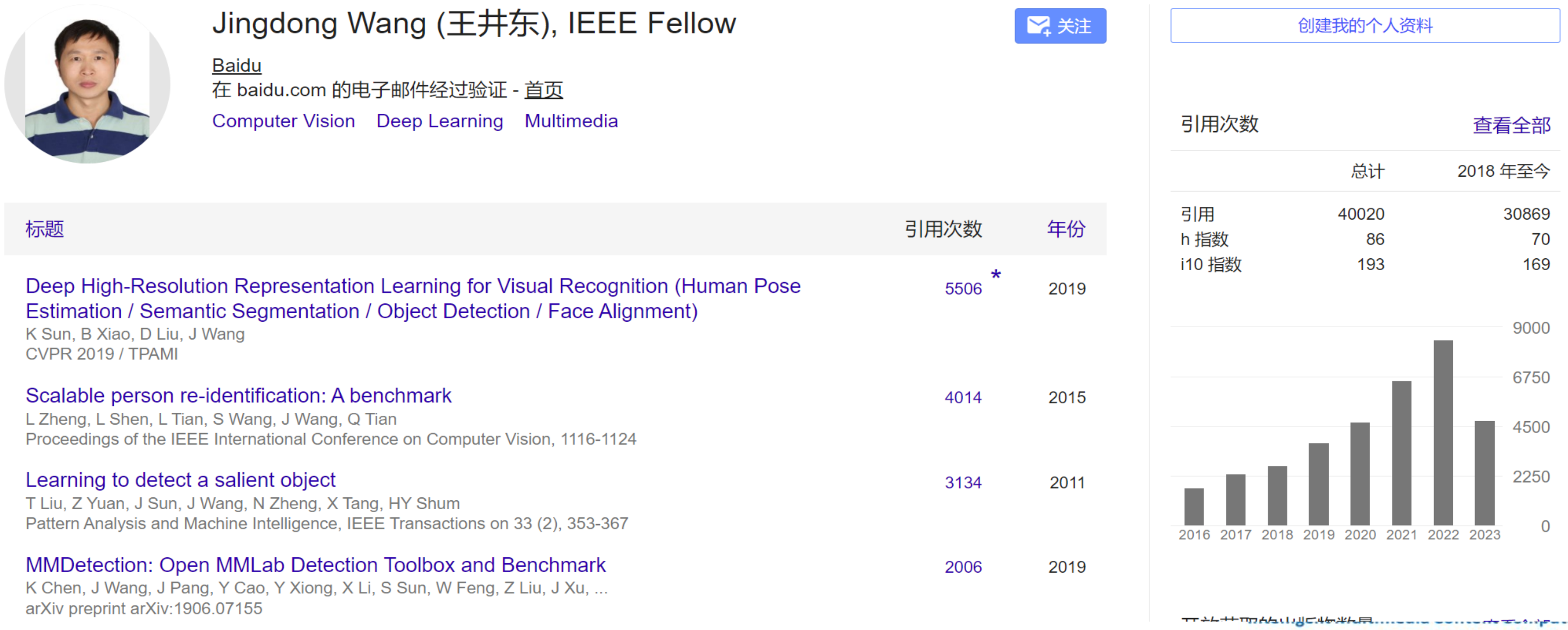
- □ 作者介绍
- □ 研究动机
- □ 本文方法
- □ 实验效果
- □ 总结反思

- 作者介绍
- 研究背景
- 研究动机
- 本文方法
- 实验效果
- 总结反思

作者介绍

Context Autoencoder for Self-Supervised Representation Learning

Xiaokang Chen[1] Mingyu Ding[2] Xiaodi Wang[3] Ying Xin[3] Shentong Mo[3] Yunhao Wang[3] Shumin Han[3]
Ping Luo[2] Gang Zeng[1] Jingdong Wang[3]



Jingdong Wang (王井东), IEEE Fellow

Baidu
在 baidu.com 的电子邮件经过验证 - 首页
Computer Vision   Deep Learning   Multimedia

标题	引用次数	年份
Deep High-Resolution Representation Learning for Visual Recognition (Human Pose Estimation / Semantic Segmentation / Object Detection / Face Alignment) K Sun, B Xiao, D Liu, J Wang CVPR 2019 / TPAMI	5506 *	2019
Scalable person re-identification: A benchmark L Zheng, L Shen, L Tian, S Wang, J Wang, Q Tian Proceedings of the IEEE International Conference on Computer Vision, 1116-1124	4014	2015
Learning to detect a salient object T Liu, Z Yuan, J Sun, J Wang, N Zheng, X Tang, HY Shum Pattern Analysis and Machine Intelligence, IEEE Transactions on 33 (2), 353-367	3134	2011
MMDetection: Open MMLab Detection Toolbox and Benchmark K Chen, J Wang, J Pang, Y Cao, Y Xiong, X Li, S Sun, W Feng, Z Liu, J Xu, ... arXiv preprint arXiv:1906.07155	2006	2019

引用次数 查看全部

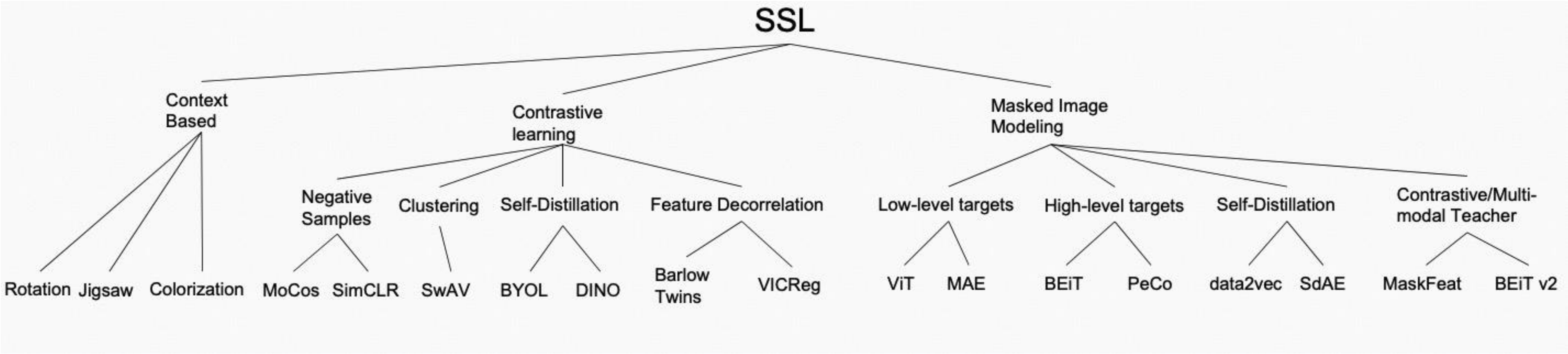
	总计	2018 年至今
引用	40020	30869
h 指数	86	70
i10 指数	193	169

开放获取的出版物数量

- □ 作者介绍
- □ 研究背景
- □ 研究动机
- □ 本文方法
- □ 实验效果
- □ 总结反思

研究背景

- 自监督学习
  - 依据pretext task划分自监督学习的种类
    - 基于上下文的
    - 基于对比学习
    - 基于掩码图像建模（生成式）

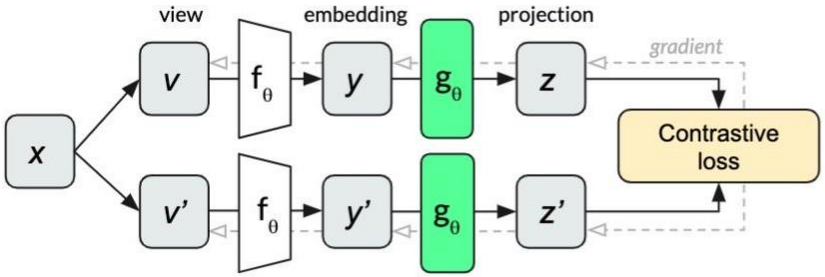


[2] Gui J, Chen T, Cao Q, et al. A Survey of Self-Supervised Learning from Multiple Perspectives: Algorithms, Theory, Applications and Future Trends[J]. arXiv preprint arXiv:2301.05712, 2023.

研究背景

□ 对比学习(Contrastive Learning)

一、SimCLR



1. augmentation $x = t(I)$ $x' = t'(I)$

2.encode $u = f_\theta(x)$ $u' = f_\theta(x')$

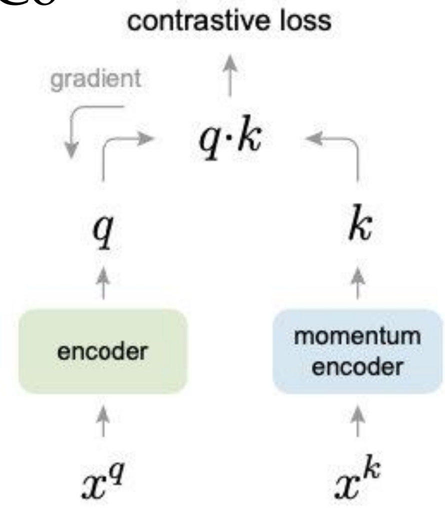
3.project $z = g_\theta(u)$ $z' = g_\theta(u')$

(z, z') Positive (z, z_k) Negative

4.Loss

$$\mathcal{L}_{CL} = -\log \frac{\exp(z \cdot z'/t)}{\sum_{i=0}^{Q} \exp(z \cdot z_i/t)},$$

二、MoCo



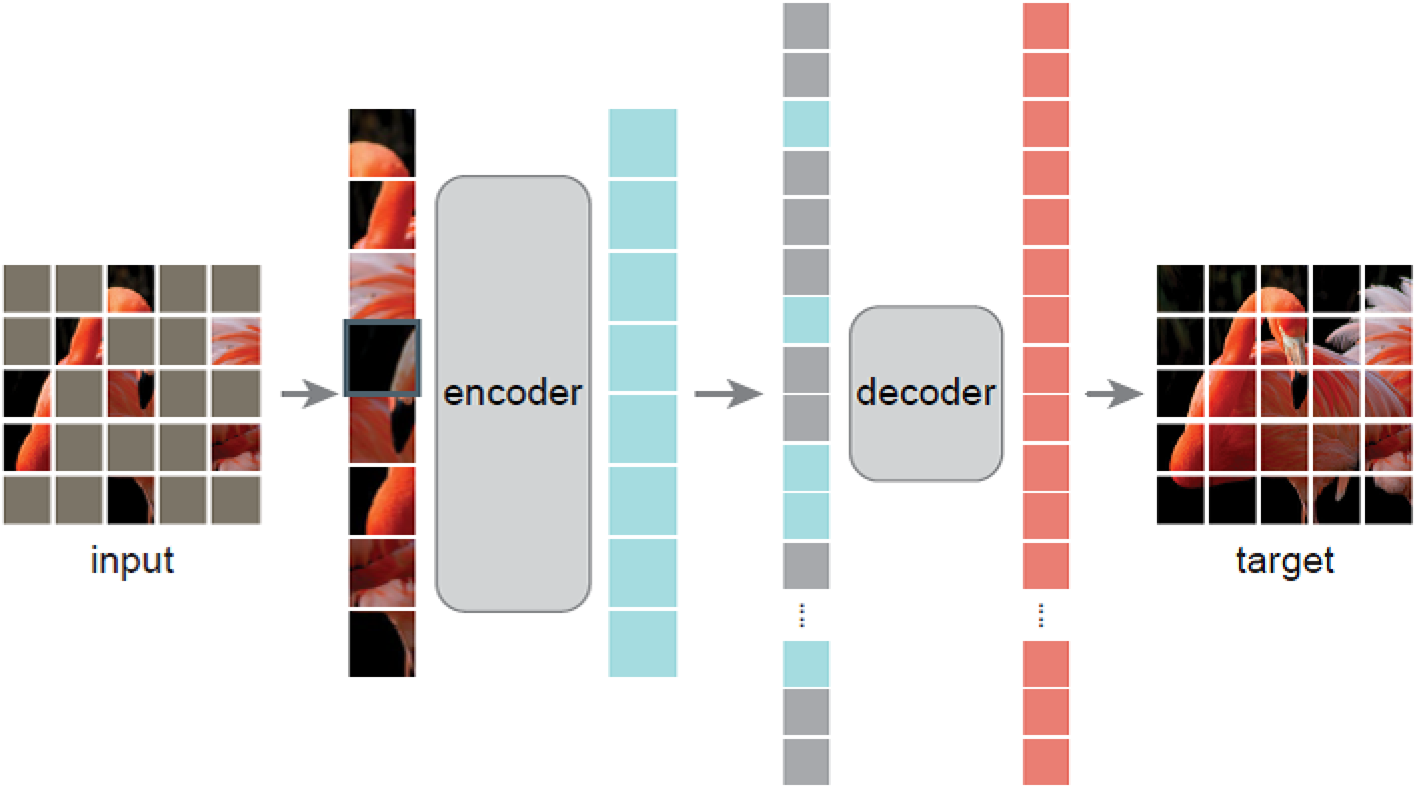
Difference：

1.用queue 保存 Image embedding

2.其中一个encoder 用EMA去update

研究动机

□ 基于掩码图像建模（生成式）



研究动机

- □ 基于掩码图像建模（生成式）

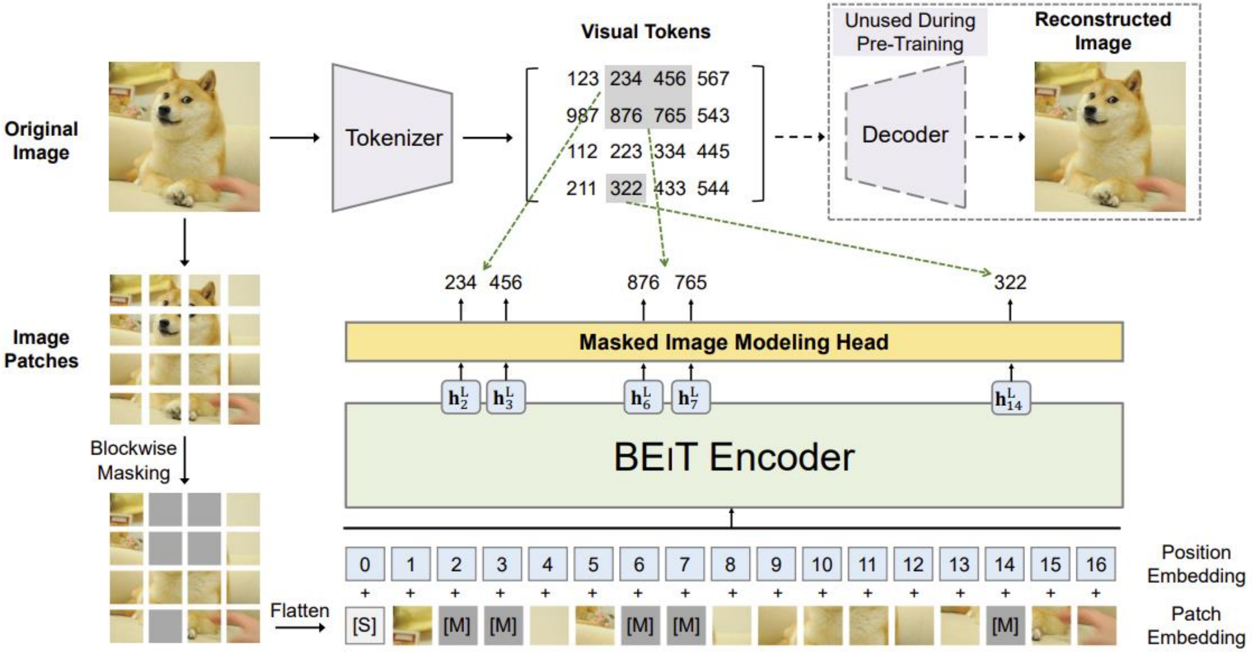


Figure 1: Overview of BEIT pre-training. Before pre-training, we learn an "image tokenizer" via autoencoding-style reconstruction, where an image is tokenized into discrete visual tokens according to the learned vocabulary. During pre-training, each image has two views, i.e., image patches, and visual tokens. We randomly mask some proportion of image patches (gray patches in the figure) and replace them with a special mask embedding [M]. Then the patches are fed to a backbone vision Transformer. The pre-training task aims at predicting the visual tokens of the *original* image based on the encoding vectors of the *corrupted* image.

研究动机

- □ Encoder-Decoder 结构：
  - ⊙ Encoder : 特征提取
  - ⊙ Decoder : 完成代理任务
  - ⊙ Decoder 中也同时输入了 Encoder 输出的编码特征，完成预训练代理任务的时候，会对这部分也进行优化，限制了 Encoder 的表征学习能力
- □ 统一架构：
  - ⊙ 单个ViT进行编解码
  - ⊙ 由于需要同时完成目标任务，限制了模型表征学习的能力。

- □ 作者介绍
- □ 研究动机
- □ 本文方法
- □ 实验效果
- □ 总结反思

本文方法

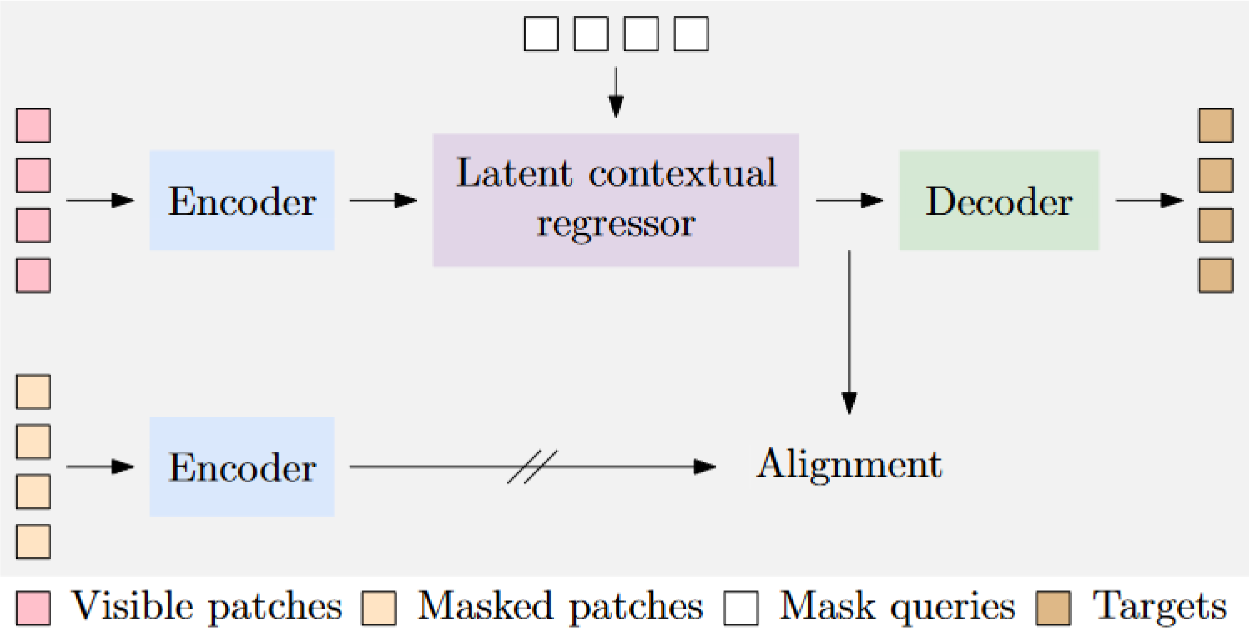
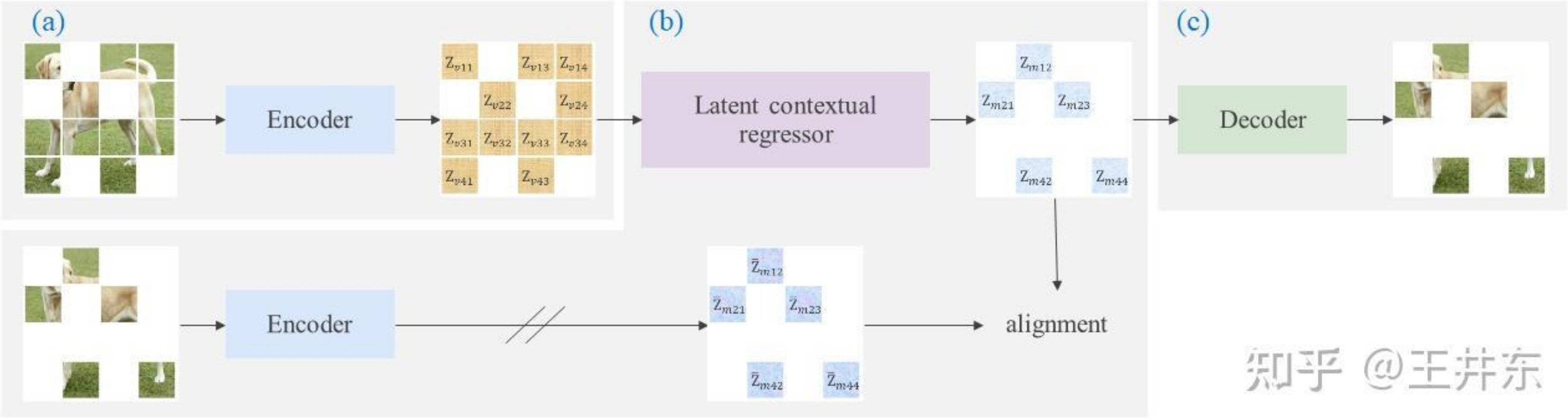
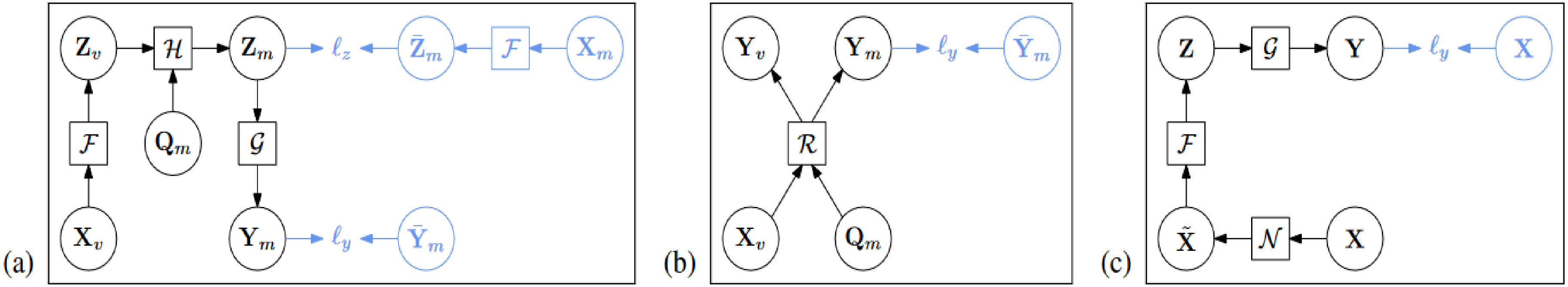


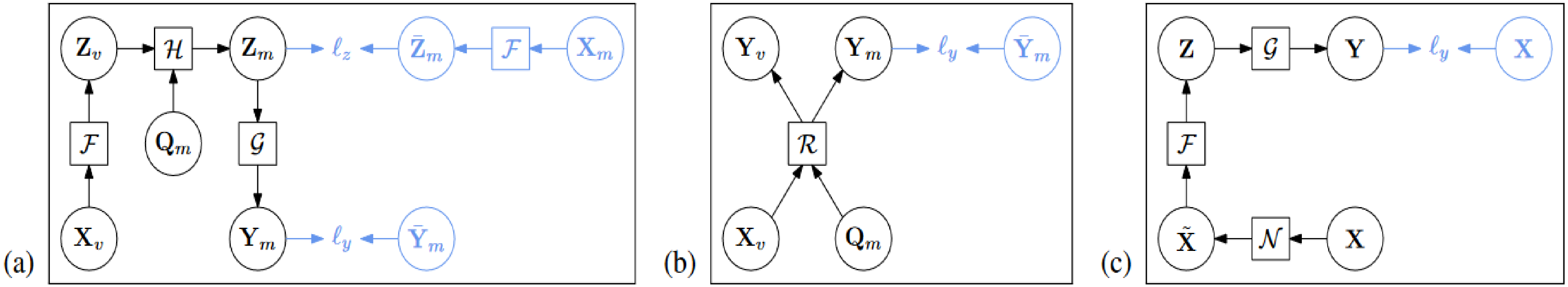
Figure 1: The pipeline of context autoencoder. Our approach pretrains the encoder by making predictions from the visible patches to the masked patches through latent contextual regressor and alignment constraint, and mapping predicted representations of masked patches to the targets.

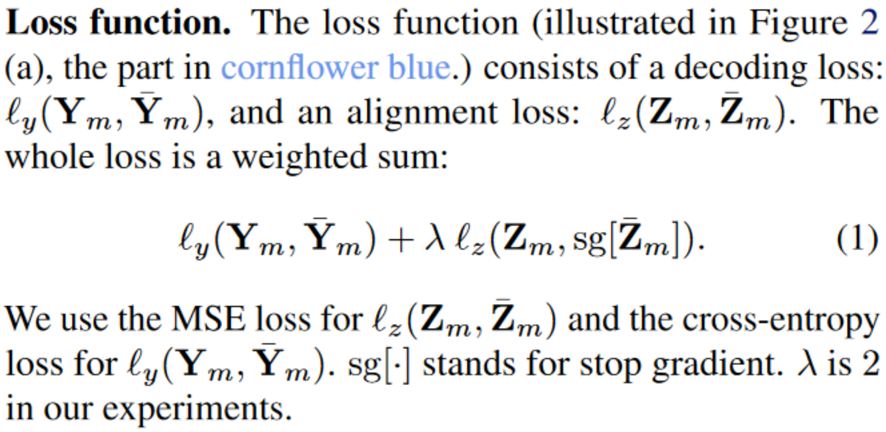
本文方法


(a)
Encoder
Z_{v11} Z_{v13} Z_{v14}
Z_{v22} Z_{v24}
Z_{v31} Z_{v32} Z_{v33} Z_{v34}
Z_{v41} Z_{v43}
(b)
Latent contextual regressor
Z_{m12}
Z_{m21} Z_{m23}
Z_{m42} Z_{m44}
(c)
Decoder
Encoder
$\bar{Z}_{m12}$
$\bar{Z}_{m21}$ $\bar{Z}_{m23}$
$\bar{Z}_{m42}$ $\bar{Z}_{m44}$
alignment


(a)
$\mathbf{Z}_v$
$\mathcal{H}$
$\mathbf{Z}_m$
ℓ_z
$\bar{\mathbf{Z}}_m$
$\mathcal{F}$
$\mathbf{X}_m$
$\mathcal{F}$
$\mathbf{Q}_m$
$\mathcal{G}$
$\mathbf{X}_v$
$\mathbf{Y}_m$
ℓ_y
$\bar{\mathbf{Y}}_m$
(b)
$\mathbf{Y}_v$
$\mathbf{Y}_m$
ℓ_y
$\bar{\mathbf{Y}}_m$
$\mathcal{R}$
$\mathbf{X}_v$
$\mathbf{Q}_m$
(c)
$\mathbf{Z}$
$\mathcal{G}$
$\mathbf{Y}$
ℓ_y
$\mathbf{X}$
$\mathcal{F}$
$\tilde{\mathbf{X}}$
$\mathcal{N}$
$\mathbf{X}$

本文方法





Loss function. The loss function (illustrated in Figure 2 (a), the part in cornflower blue.) consists of a decoding loss: $\ell_y(\mathbf{Y}_m, \bar{\mathbf{Y}}_m)$, and an alignment loss: $\ell_z(\mathbf{Z}_m, \bar{\mathbf{Z}}_m)$. The whole loss is a weighted sum:

$$\ell_y(\mathbf{Y}_m, \bar{\mathbf{Y}}_m) + \lambda\, \ell_z(\mathbf{Z}_m, \mathrm{sg}[\bar{\mathbf{Z}}_m]). \qquad (1)$$

We use the MSE loss for $\ell_z(\mathbf{Z}_m, \bar{\mathbf{Z}}_m)$ and the cross-entropy loss for $\ell_y(\mathbf{Y}_m, \bar{\mathbf{Y}}_m)$. $\mathrm{sg}[\cdot]$ stands for stop gradient. λ is 2 in our experiments.

本文方法

- 分析：
  - CAE能够关注全局图像信息

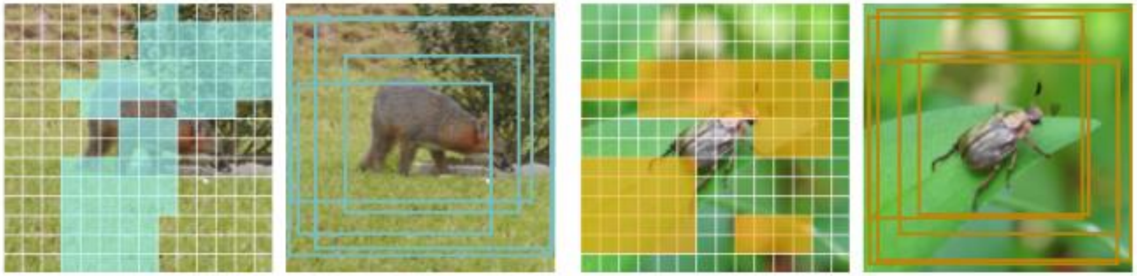


Figure 3: Illustration of random block-wise sampling and random cropping. Random block-wise sampling is used in our approach. Random cropping is a key data-augmentation scheme for contrastive pretraining.

本文方法

- 分析：
  - CAE能够关注全局图像信息

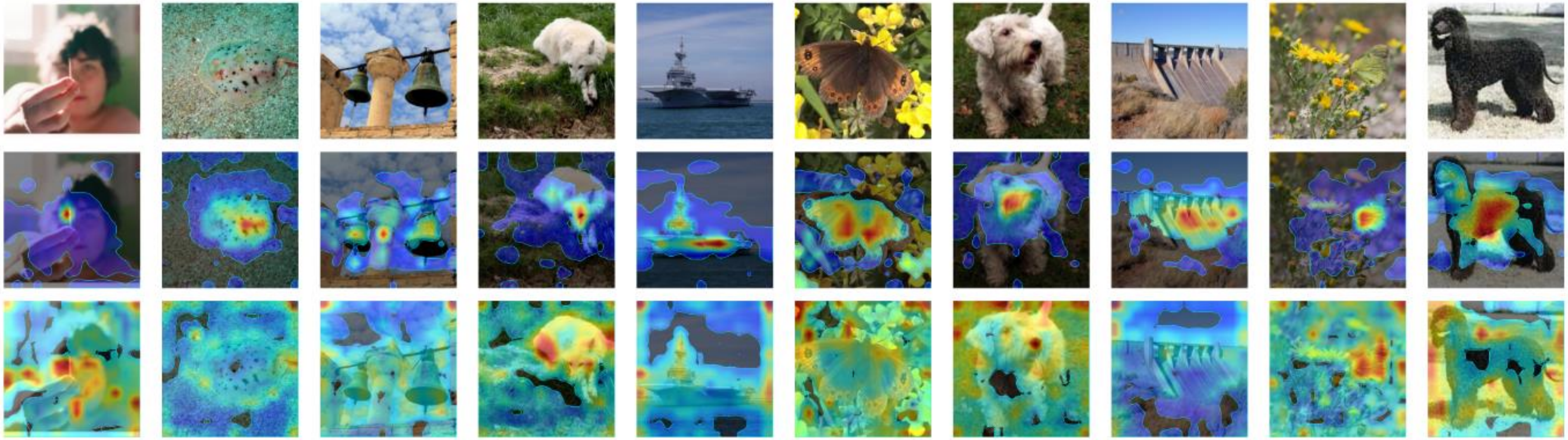
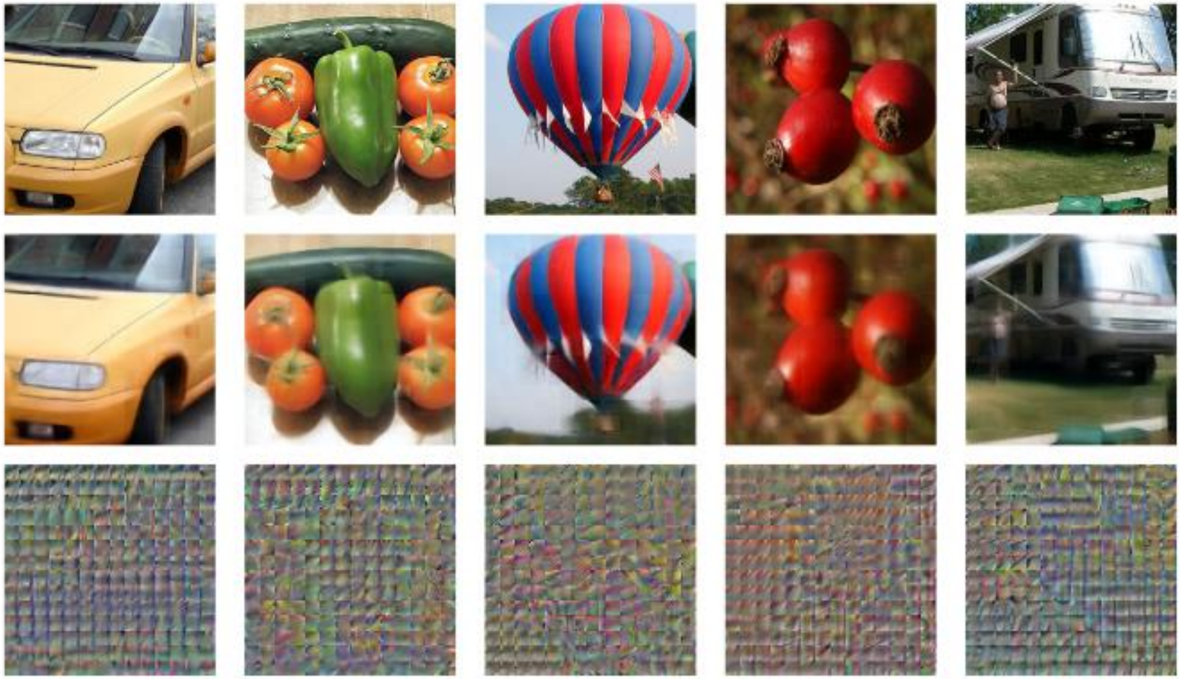


Figure 6: Illustrating the attention map averaged over 12 attention heads between the class token and the patch tokens in the last layer of the ViT encoder pretrained on ImageNet-1K. The region inside the blue contour is obtained by thresholding the attention weights to keep 50% of the mass. Top: Input image, Middle: MoCo v3, a typical contrastive learning method, and Bottom: our CAE. One can see that MoCo v3 tends to focus mainly on the centering regions and little on other patches, and our CAE tends to consider almost all the patches.

本文方法

- 分析：
  - 预测是在编码空间中进行的，对齐约束是有效的



- □ 作者介绍
- □ 研究动机
- □ 本文方法
- □ 实验效果
- □ 总结反思

实验效果

- Linear probing
  - 训练与标签对应的线性分类器
- Attentive probing
  - 标签只包含图像中心部分区域，不能体现对全局效果的关注

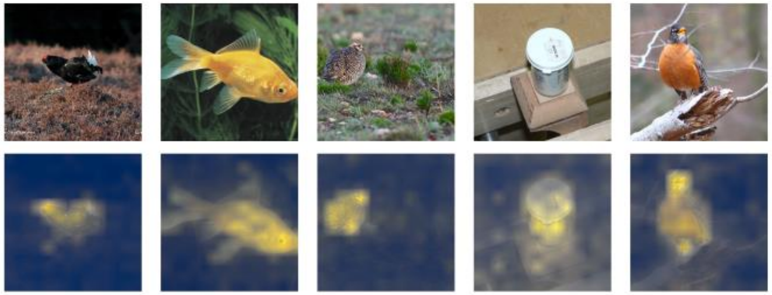
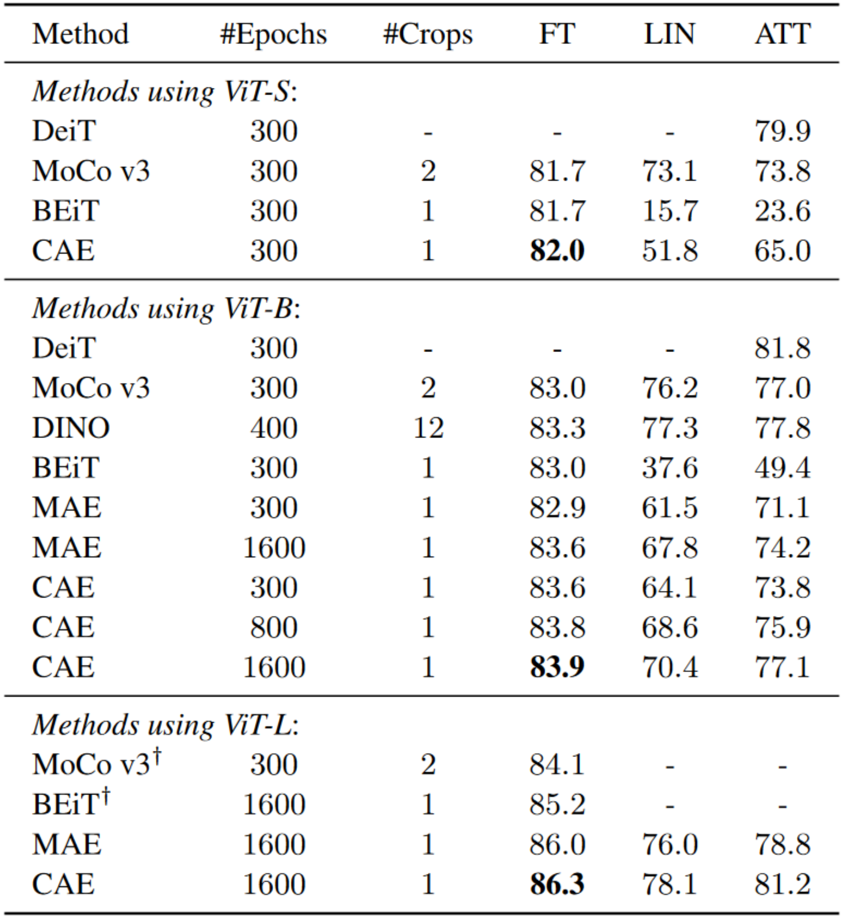


Figure 7: Illustrating the cross-attention unit in attentive probing. The attention map (bottom) is the average of cross-attention maps over 12 heads between the extra class token and the patches. One can see that the attended region lies mainly in the object, which helps image classification.

实验效果



Method	#Epochs	#Crops	FT	LIN	ATT
Methods using ViT-S:					
DeiT	300	-	-	-	79.9
MoCo v3	300	2	81.7	73.1	73.8
BEiT	300	1	81.7	15.7	23.6
CAE	300	1	**82.0**	51.8	65.0
Methods using ViT-B:					
DeiT	300	-	-	-	81.8
MoCo v3	300	2	83.0	76.2	77.0
DINO	400	12	83.3	77.3	77.8
BEiT	300	1	83.0	37.6	49.4
MAE	300	1	82.9	61.5	71.1
MAE	1600	1	83.6	67.8	74.2
CAE	300	1	83.6	64.1	73.8
CAE	800	1	83.8	68.6	75.9
CAE	1600	1	**83.9**	70.4	77.1
Methods using ViT-L:					
MoCo v3†	300	2	84.1	-	-
BEiT†	1600	1	85.2	-	-
MAE	1600	1	86.0	76.0	78.8
CAE	1600	1	**86.3**	78.1	81.2

Table 1: Pretraining quality evaluation in terms of fine-tuning (FT), linear probing (LIN), and attentive probing (ATT). #Epochs refers to the number of pretraining epochs. For reference, we report the top-1 accuracy (in the column ATT) of the supervised training approach DeiT (Touvron et al., 2020) to show how far our ATT score is from supervised training. The results for other models and our models are based on our implementations for fine-tuning, linear probing, and attentive probing. MoCo v3 and DINO adopt multi-crop pretraining augmentation in each mini-batch. MoCo v3: 2 global crops of 224 × 224. DINO: 2 global crops of 224 × 224 and 10 local crops of 96 × 96. †: these results are from (He et al., 2021).

实验效果

- □ Ablation study

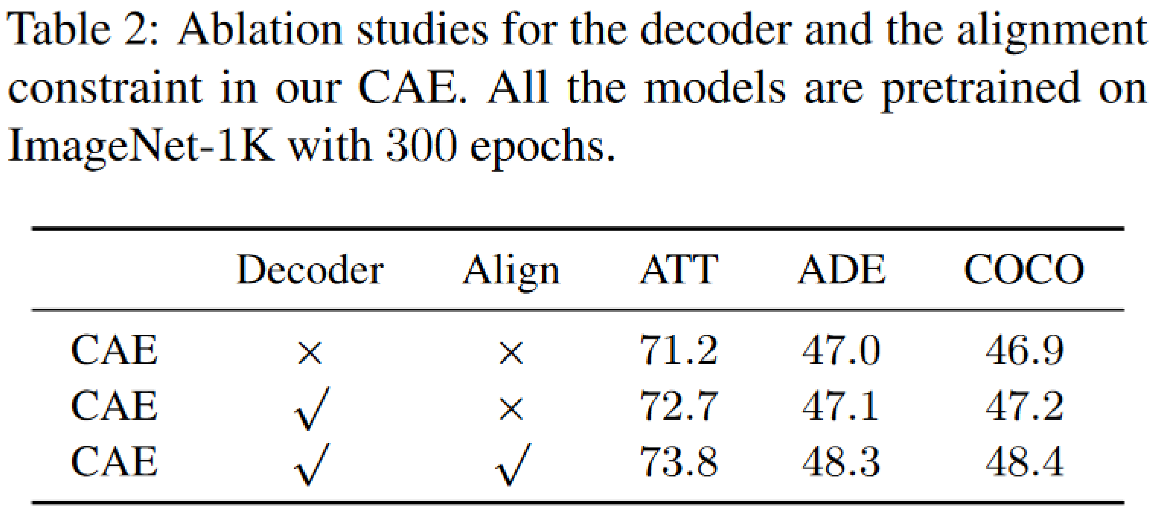
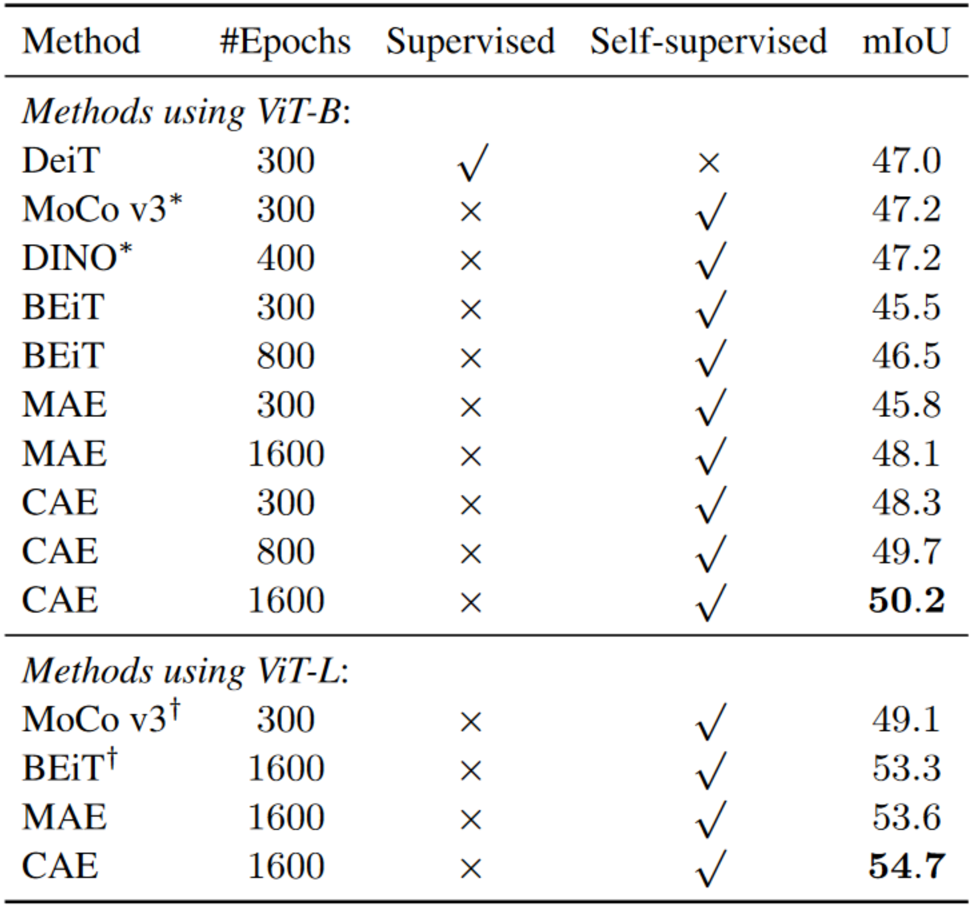


Table 2: Ablation studies for the decoder and the alignment constraint in our CAE. All the models are pretrained on ImageNet-1K with 300 epochs.

	Decoder	Align	ATT	ADE	COCO
CAE	×	×	71.2	47.0	46.9
CAE	√	×	72.7	47.1	47.2
CAE	√	√	73.8	48.3	48.4

实验效果

□ Downstream Tasks



Method	#Epochs	Supervised	Self-supervised	mIoU
Methods using ViT-B:				
DeiT	300	√	×	47.0
MoCo v3*	300	×	√	47.2
DINO*	400	×	√	47.2
BEiT	300	×	√	45.5
BEiT	800	×	√	46.5
MAE	300	×	√	45.8
MAE	1600	×	√	48.1
CAE	300	×	√	48.3
CAE	800	×	√	49.7
CAE	1600	×	√	**50.2**
Methods using ViT-L:				
MoCo v3†	300	×	√	49.1
BEiT†	1600	×	√	53.3
MAE	1600	×	√	53.6
CAE	1600	×	√	**54.7**

Table 3: Semantic segmentation on ADE20K. All the results are based on the same implementation for semantic segmentation. #Epochs refers to the number of pretraining epochs. *: use multi-crop pretraining augmentation (See Table 1) and equivalently take a larger number of epochs compared to one-crop augmentation. †: these results are from (He et al., 2021).

实验效果

□ Downstream Tasks

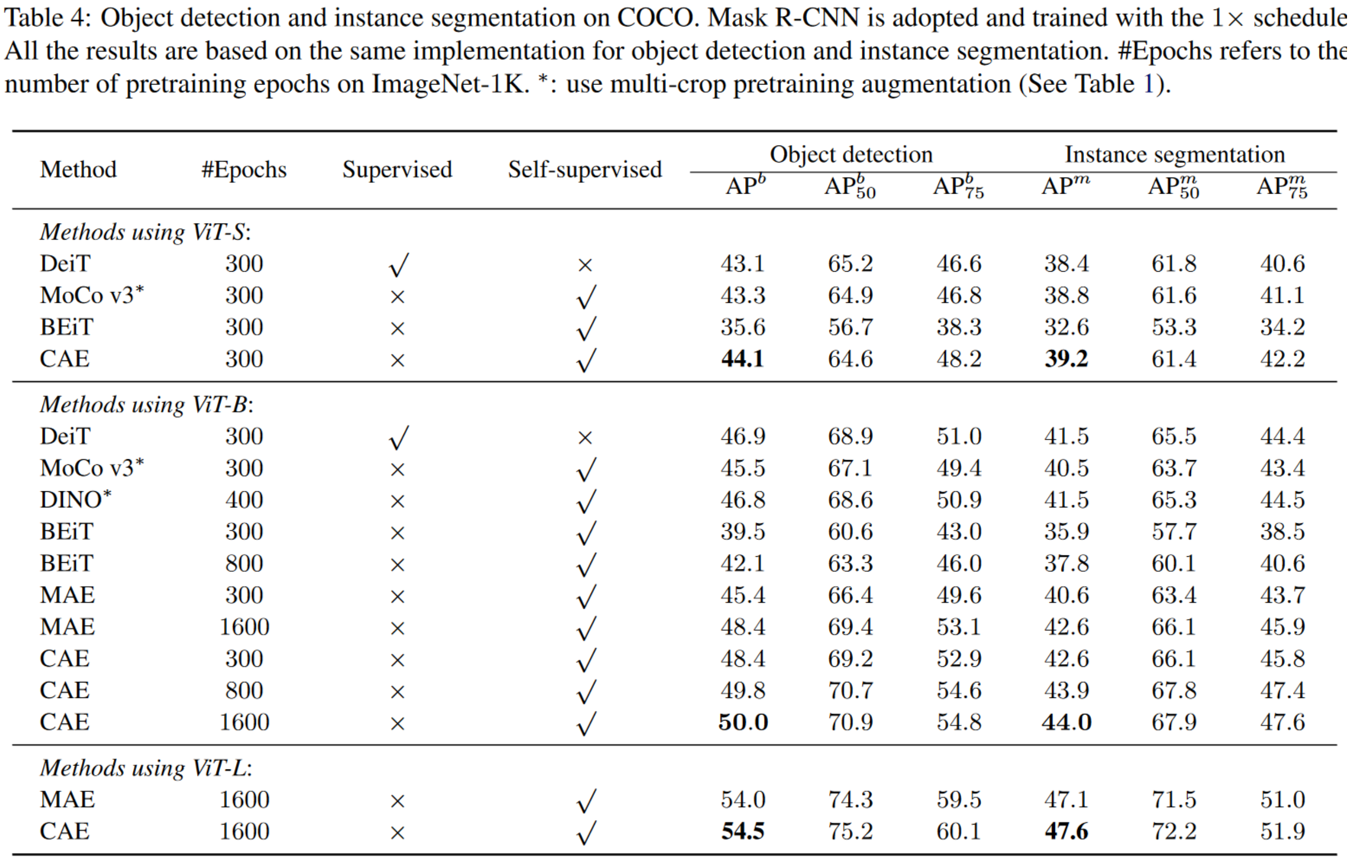


Table 4: Object detection and instance segmentation on COCO. Mask R-CNN is adopted and trained with the 1× schedule All the results are based on the same implementation for object detection and instance segmentation. #Epochs refers to the number of pretraining epochs on ImageNet-1K. *: use multi-crop pretraining augmentation (See Table 1).

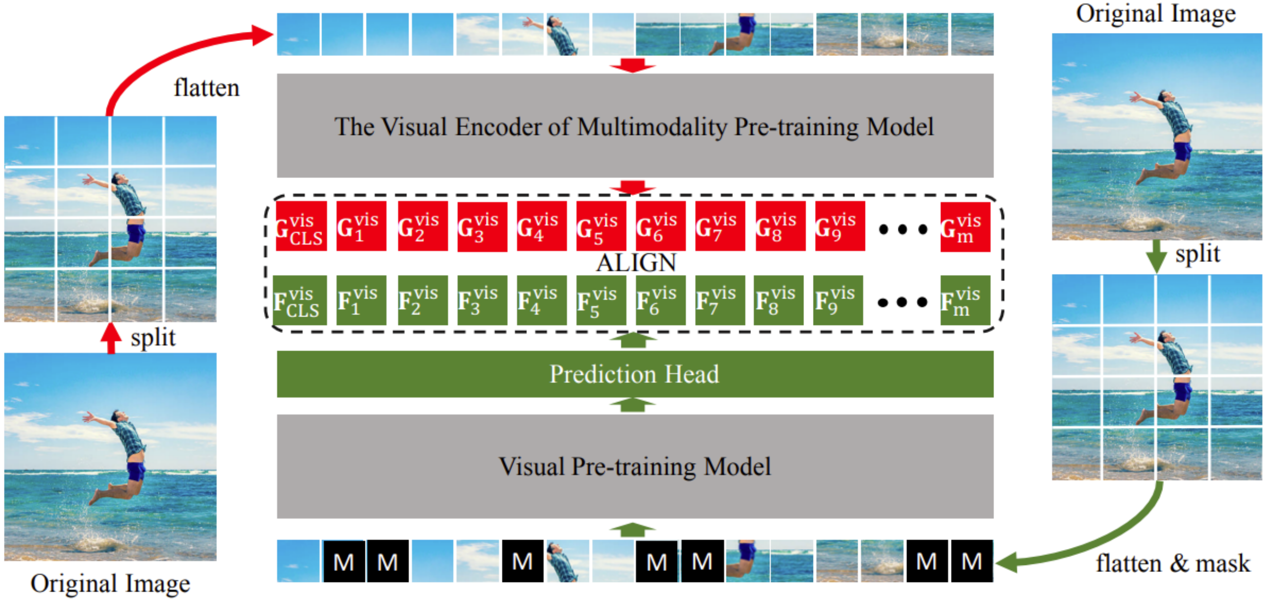
Method	#Epochs	Supervised	Self-supervised	Object detection			Instance segmentation		
				AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
Methods using ViT-S:									
DeiT	300	√	×	43.1	65.2	46.6	38.4	61.8	40.6
MoCo v3*	300	×	√	43.3	64.9	46.8	38.8	61.6	41.1
BEiT	300	×	√	35.6	56.7	38.3	32.6	53.3	34.2
CAE	300	×	√	**44.1**	64.6	48.2	**39.2**	61.4	42.2
Methods using ViT-B:									
DeiT	300	√	×	46.9	68.9	51.0	41.5	65.5	44.4
MoCo v3*	300	×	√	45.5	67.1	49.4	40.5	63.7	43.4
DINO*	400	×	√	46.8	68.6	50.9	41.5	65.3	44.5
BEiT	300	×	√	39.5	60.6	43.0	35.9	57.7	38.5
BEiT	800	×	√	42.1	63.3	46.0	37.8	60.1	40.6
MAE	300	×	√	45.4	66.4	49.6	40.6	63.4	43.7
MAE	1600	×	√	48.4	69.4	53.1	42.6	66.1	45.9
CAE	300	×	√	48.4	69.2	52.9	42.6	66.1	45.8
CAE	800	×	√	49.8	70.7	54.6	43.9	67.8	47.4
CAE	1600	×	√	**50.0**	70.9	54.8	**44.0**	67.9	47.6
Methods using ViT-L:									
MAE	1600	×	√	54.0	74.3	59.5	47.1	71.5	51.0
CAE	1600	×	√	**54.5**	75.2	60.1	**47.6**	72.2	51.9

CAE v2: Context Autoencoder with CLIP Target

Xinyu Zhang[1*], Jiahui Chen[2,1*], Junkun Yuan[3,1], Qiang Chen[1], Jian Wang[1], Xiaodi Wang[1], Shumin Han[1],
Xiaokang Chen[4,1], Jimin Pi[1], Kun Yao[1], Junyu Han[1], Errui Ding[1], Jingdong Wang[1†]
[1]Baidu VIS [2]Beihang University [3]Zhejiang University [4]Peking University
{zhangxinyu14,chenjiahui06,yuanjunkun,chenqiang13,wangjian33,wangxiaodi03,hanshumin,}
{chenxiaokang03,pijimin01,hanjunyu,dingerrui,wangjingdong}@baidu.com

研究动机

- MIM方法在线性验证上效果比对比学习方法差
  - MIM特征缺少语义信息 ---多模态引导视觉预训练 (MVP)



flatten
The Visual Encoder of Multimodality Pre-training Model
G_{CLS}^{vis} G_1^{vis} G_2^{vis} G_3^{vis} G_4^{vis} G_5^{vis} G_6^{vis} G_7^{vis} G_8^{vis} G_9^{vis} ••• G_m^{vis}
ALIGN
F_{CLS}^{vis} F_1^{vis} F_2^{vis} F_3^{vis} F_4^{vis} F_5^{vis} F_6^{vis} F_7^{vis} F_8^{vis} F_9^{vis} ••• F_m^{vis}
Prediction Head
Visual Pre-training Model
M M M M M M M
split
Original Image
Original Image
split
flatten & mask

Fig. 2. Framework of the proposed MVP, where each M (MASK) denotes a masked token, and $\mathbf{F}_m^{\text{vis}}/\mathbf{G}_m^{\text{vis}}$ (or $\mathbf{F}_{\text{CLS}}^{\text{vis}}/\mathbf{G}_{\text{CLS}}^{\text{vis}}$) denotes the extracted features of each normal (or CLS) token. MVP is designed with the token-level multimodal information prediction pretext task to guide the pre-training of visual model.

本文方法

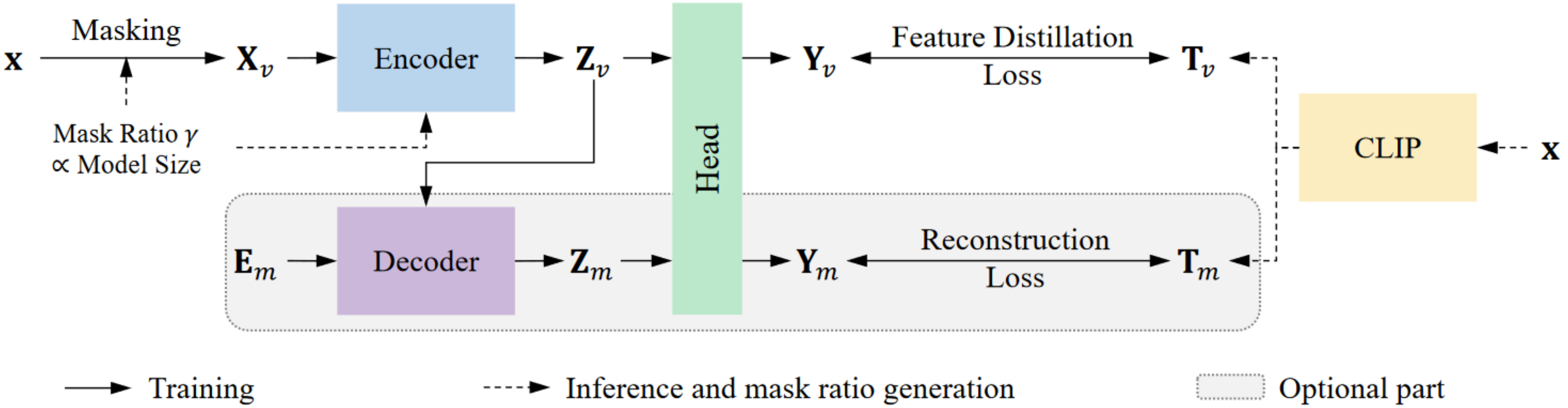
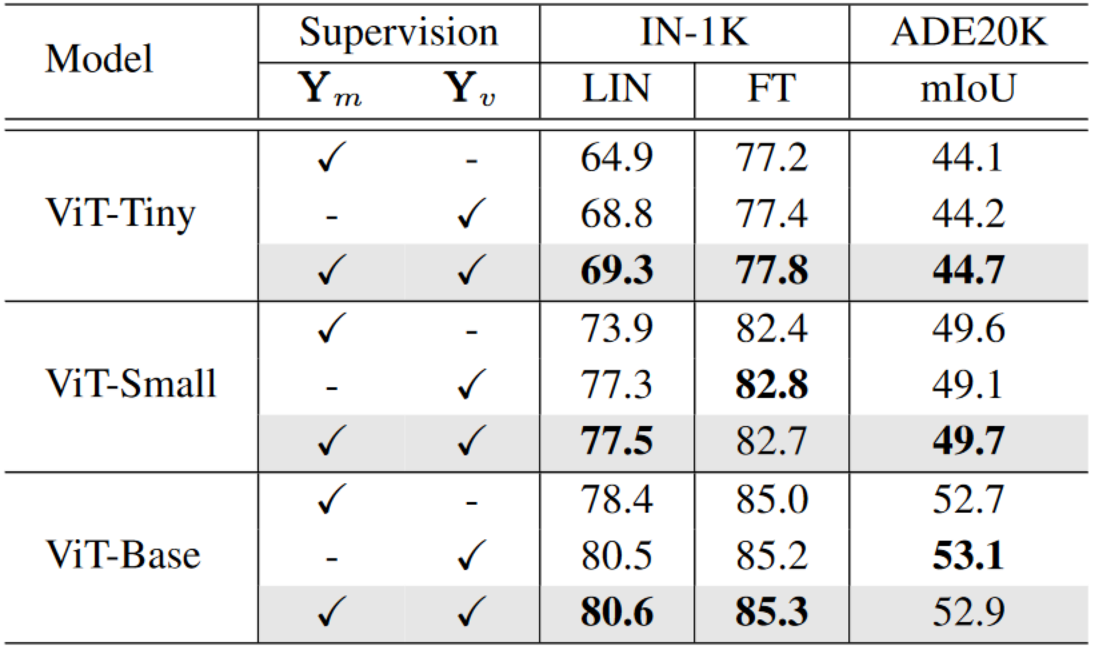


Figure 1. Overview of the proposed CAE v2. CAE v2 first masks the input image $\mathbf{x}$ with the mask ratio γ, which is positively correlated with the model size of encoder. $\propto$ represents the positive correlation. Then, CAE v2 inputs the visible patches $\mathbf{X}_v$ into the encoder to obtain the latent representation $\mathbf{Z}_v$. The decoder receives $\mathbf{Z}_v$ and the mask token $\mathbf{E}_m$ to recover the latent representations of the masked patches $\mathbf{Z}_m$. After a lightweight head, $\mathbf{Z}_v$ and $\mathbf{Z}_m$ are projected to $\mathbf{Y}_v$ and $\mathbf{Y}_m$. CAE v2 also inputs $\mathbf{x}$ into the CLIP model to generate the target supervisions, which are split to $\mathbf{T}_v$ and $\mathbf{T}_m$ according to the absolute positions of $\mathbf{X}_v$ and $\mathbf{X}_m$. The optimization is applied on the prediction $\mathbf{Y}_v$ and the target supervision $\mathbf{T}_v$ of visible patches. Meanwhile, the loss on $\mathbf{Y}_m$ and $\mathbf{T}_m$ for masked patches is optional.

本文方法

- □ 对于可见patch的监督效果好于mask的



Model	Supervision		IN-1K		ADE20K
	$\mathbf{Y}_m$	$\mathbf{Y}_v$	LIN	FT	mIoU
ViT-Tiny	✓	-	64.9	77.2	44.1
	-	✓	68.8	77.4	44.2
	✓	✓	**69.3**	**77.8**	**44.7**
ViT-Small	✓	-	73.9	82.4	49.6
	-	✓	77.3	**82.8**	49.1
	✓	✓	**77.5**	82.7	**49.7**
ViT-Base	✓	-	78.4	85.0	52.7
	-	✓	80.5	85.2	**53.1**
	✓	✓	**80.6**	**85.3**	52.9

Table 1. Influences of the supervision position in our CAE v2. Default settings are marked in gray.

本文方法

- □ 最优mask比例与模型大小一致

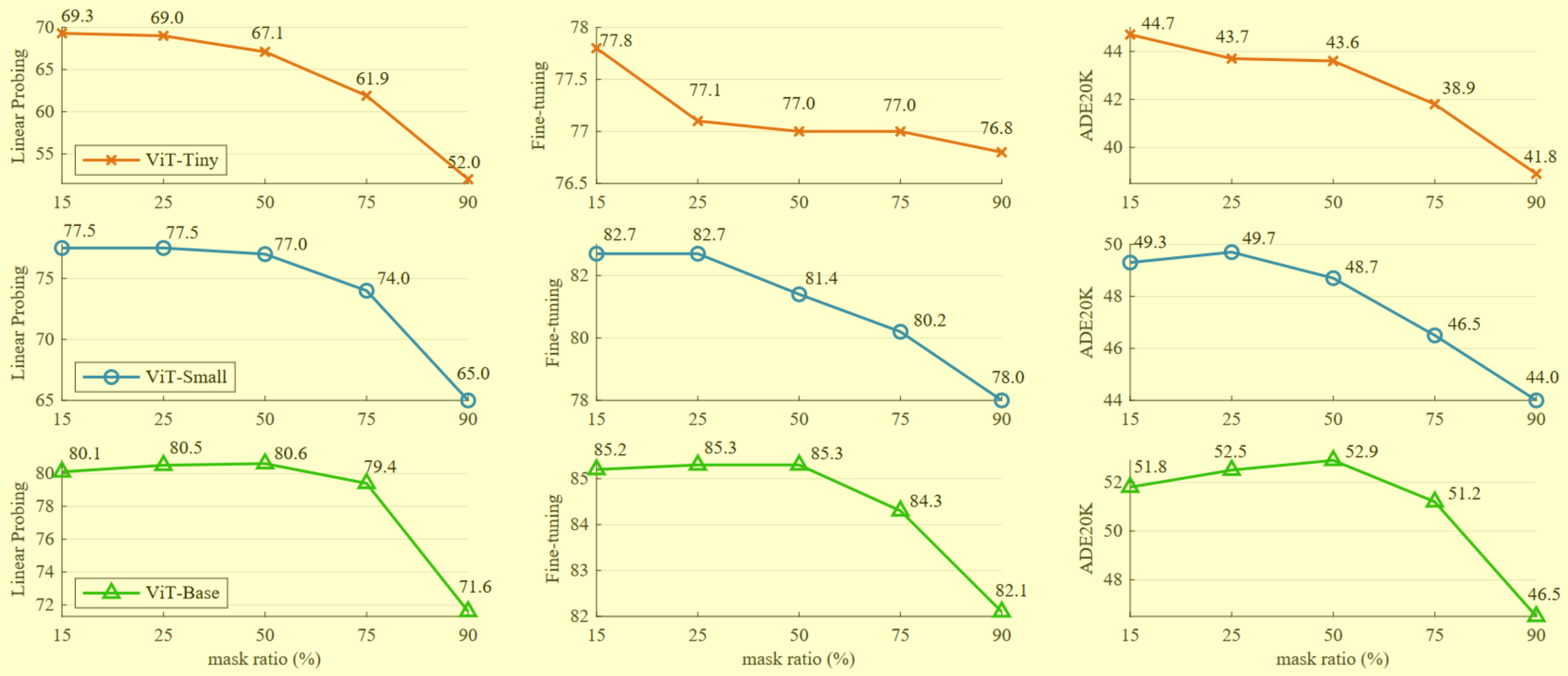


Figure 3. Influences of the mask ratio in our CAE v2 on different model sizes, including (top row) ViT-Tiny, (middle row) ViT-Small and (bottom row) ViT-Base. The optimal mask ratio is positively correlated to the model size. A higher mask ratio is more appropriate to a larger model, while the smaller model prefers a lower mask ratio. The y-axes is the Top-1 accuracy (%) on (left column) linear probing and (middle column) fine-tuning on ImageNet-1K, and (right column) mIoU (%) on ADE20K.

总结反思

- 解耦表征学习和解决prefix task
- 提出更适合该方法的评价指标?
- 小数据集（细粒度）MIM方法与对比学习方法的比较

Thank for your attention !