ICLR 2023 notable top 25%

Vision Transformer Adapter for Dense Predictions

Zhe Chen[1,2*], Yuchen Duan[2,3*], Wenhai Wang[2], Junjun He[2], Tong Lu[1], Jifeng Dai[2,3], Yu Qiao[2]

[1]Nanjing University, [2]Shanghai AI Laboratory, [3]Tsinghua University

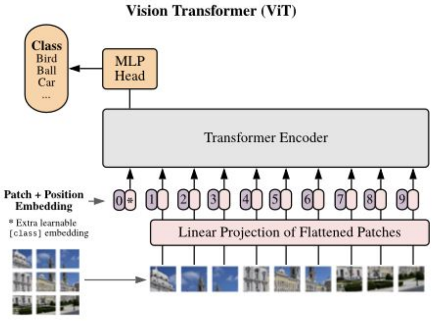
Oct 19, 2023

Minkyu Kim
EffL@POSTECH

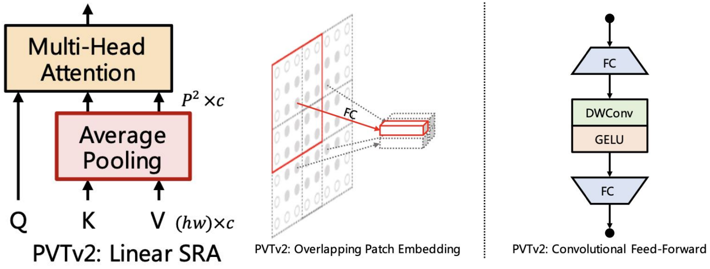
Introduction

Transformer

- Remarkable success in a broad range of computer vision fields
  - Due to dynamic modeling capability & attention mechanism
- Surpassing CNN models and reaching SOTA performance in many vision tasks
- Types : Plain ViT, Its hierarchical variants



Plain ViT



Its hierarchical variants

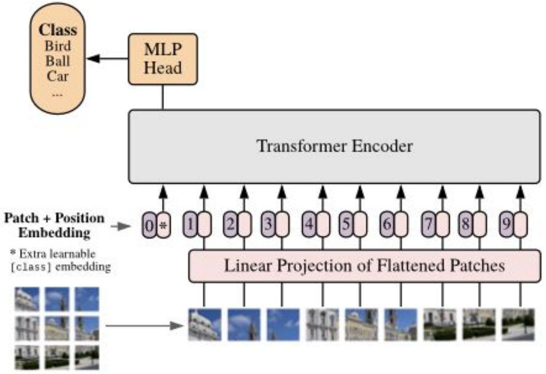
Alexey Dosovitskiy et al., "AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE ", ICLR, 2021.
Wenhai Wang et al. "PVT v2: Improved Baselines with Pyramid Vision Transformer", CVMJ, 2022.

Introduction

Plain Transformer (ViT)

- No assumption of input data
  - Can use massive multi-modal data for pre-training (Image, text, video, ...)
    - Encourages the model to learn semantic-rich representations

- However, defects in dense predictions compared to vision-specific transformers
  - Lacking image-related prior knowledge results in lower performance



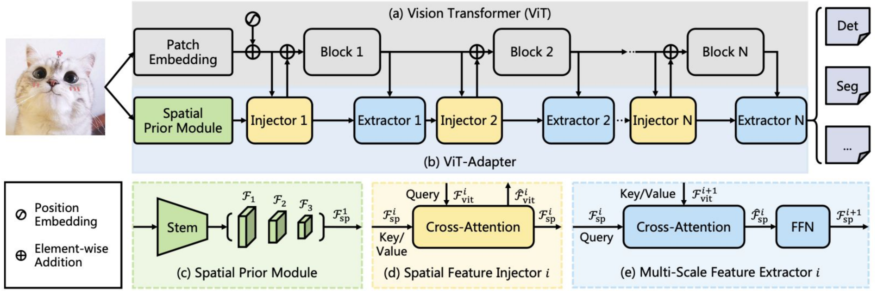
Introduction

💡 Idea : Plain ViT + Adapter

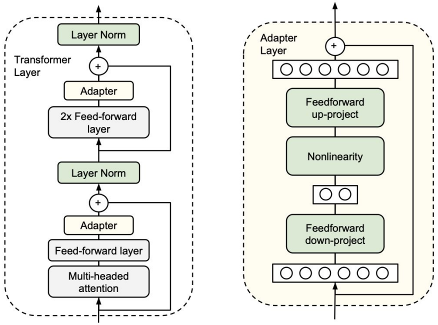
- Goal
  - Develop an adapter to close the performance gap between the plain ViT and vision-specific backbones for dense prediction tasks

- Inspired by the adapters in the NLP field

- result : **ViT-Adapter**



Overall Architecture of ViT-Adapter



Adapter in NLP field(Neil Houlsby et al.)

Neil Houlsby et al., "Parameter-Efficient Transfer Learning for NLP ", ICML, 2019.

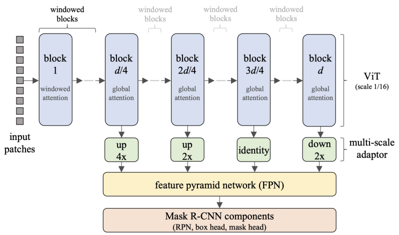
Introduction

Concurrent work

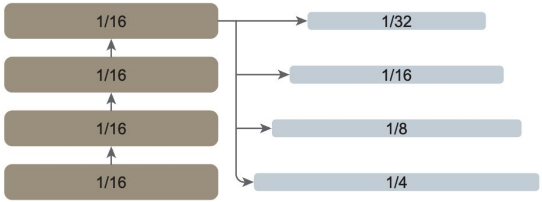
- Yanghao Li et al., ViTDet
  - Employed some upsampling and downsampling modules to adapt the plain ViT for object detection

- Weakness
  - Under regular training settings, their detection performance is still inferior to recent models
    👉 it is still challenging to design a powerful dense prediction task adapter for ViT

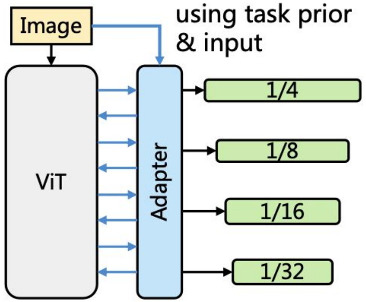
Apply ImageNet supervised pre-training and fine-tune for 36 epochs



Yanghao Li et al.



ViTDet



ViT-Adapter

Yanghao Li et al., "Benchmarking detection transfer learning with vision transformers", arXiv, 2021.
Yanghao Li et al., "Exploring Plain Vision Transformer Backbones for Object Detection", ECCV, 2022.

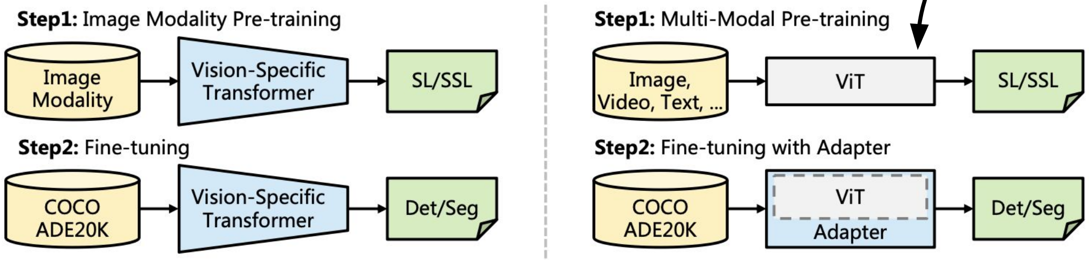
Introduction

ViT-adapter

- Pre-training-free additional network
  - Can efficiently adapt the plain ViT to downstream dense prediction tasks without modifying its original architecture

- Adapter : introduce the vision-specific inductive biases into the plain ViT
  - Spatial prior module
  - Spatial feature injector
  - Multi-scale feature extractor

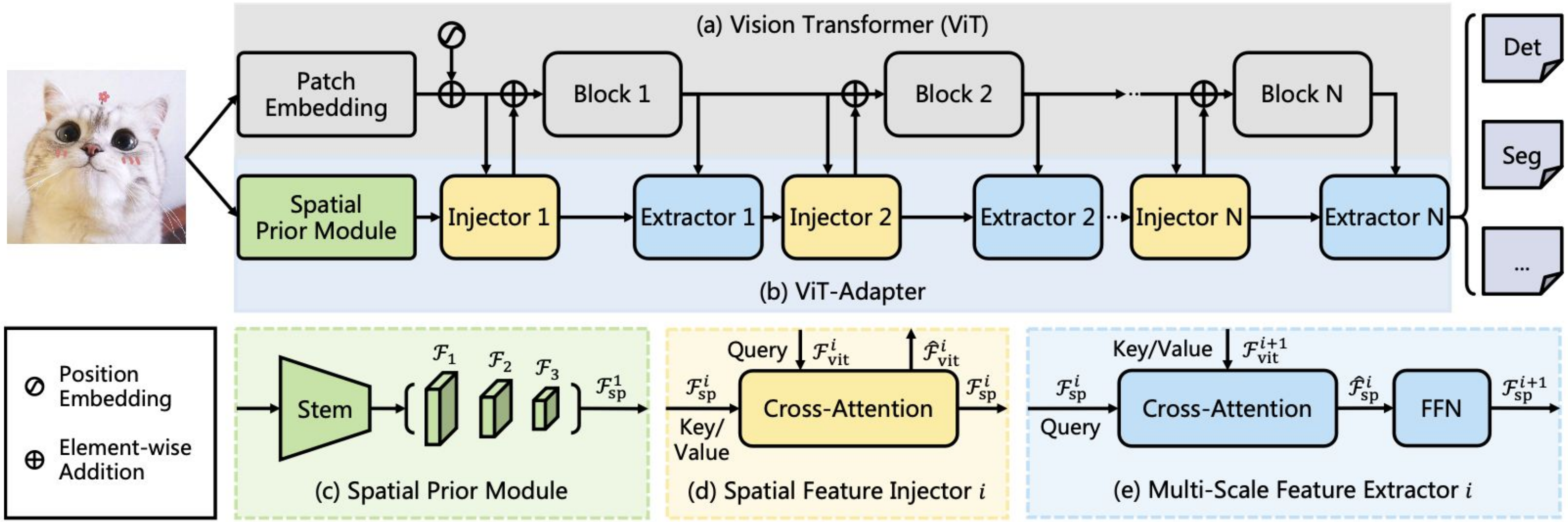
can be pre-trained with not only images but also multi-modal data



(a) Previous Paradigm (b) Our Paradigm

Vision Transformer Adapter

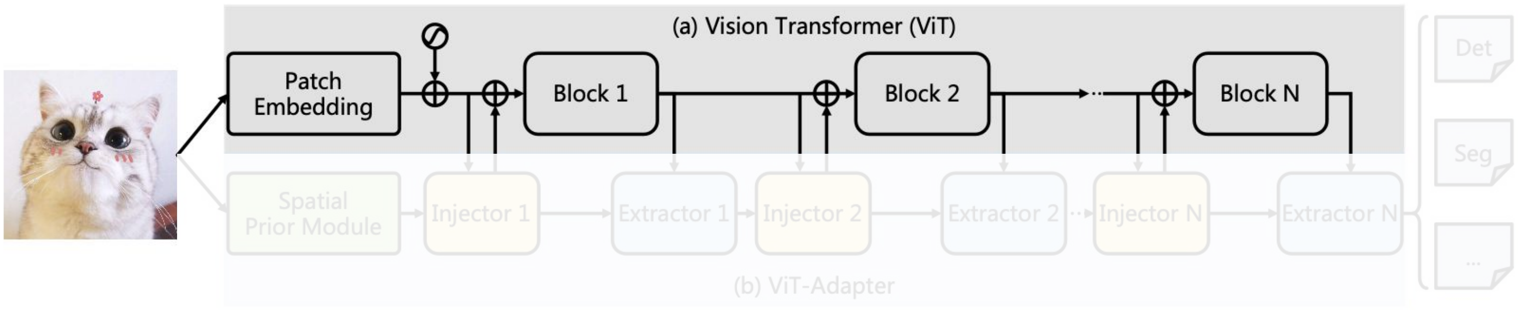
Overall Architecture



Vision Transformer Adapter

Plain ViT (pre-trained)

- Use original architecture
  - Patch embedding : 16x16 non-overlapping patches
    - Feature resolution is reduced to 1/16 of the original image
  - Consist of N blocks (each block contain the ‘L/N’ encoder layers)

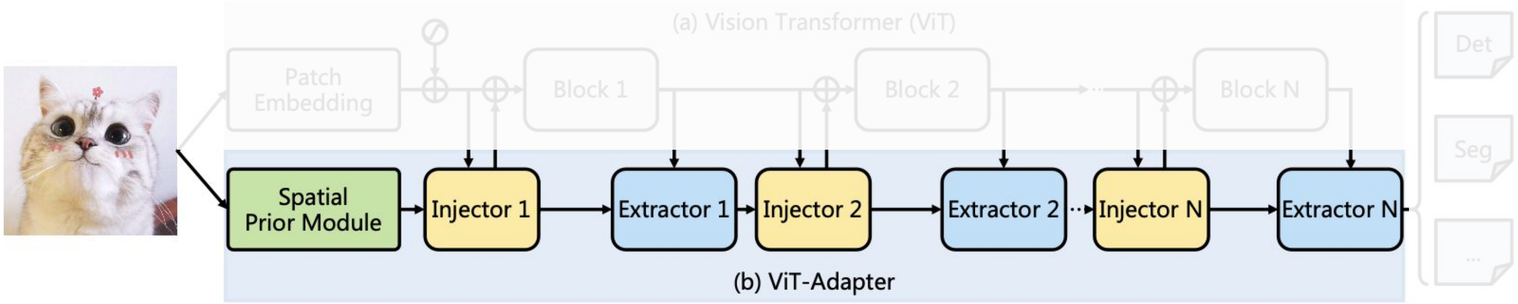


Vision Transformer Adapter

Adapter

- Contains 3 types of module
  - Spatial prior module
  - Spatial feature injector
  - Multi-scale feature extractor

- Injector & Extractor
  - Adopt sparse attention(default : Xizhou Zhu et al.) to reduce computational cost



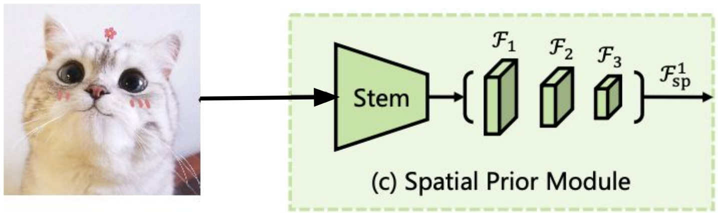
Xizhou Zhu et al., “Deformable DETR: Deformable Transformers for End-to-End Object Detection”, ICLR, 2020.

Vision Transformer Adapter

Adapter : Spatial Prior Module

- Model the local spatial contexts of images parallel with the patch embedding layer

- Standard convolutional stem : three conv. and max-pooling
  - Input : Image
  - Output : Feature pyramid $\{F_1, F_2, F_3\}$ (D-dim. feature maps with resolutions of 1/8, 1/16, and 1/32)
  - Feature pyramid : be flattened and concatenated into feature tokens F_{sp}^{1} 👉 passed to Injector



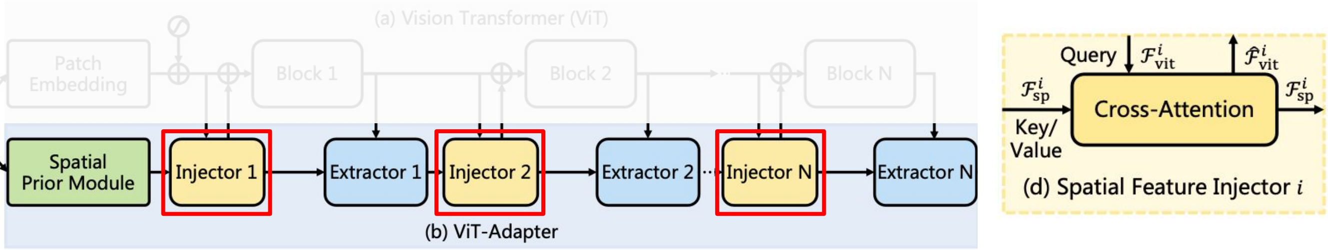
$$\mathcal{F}_{\mathrm{sp}}^1 \in \mathbb{R}^{(\frac{HW}{8^2}+\frac{HW}{16^2}+\frac{HW}{32^2})\times D}$$

Vision Transformer Adapter

Adapter : Spatial Feature Injector

- Inject the spatial priors into ViT
  - Method : cross-attention (equation : $\hat{\mathcal{F}}^i_{\text{vit}} = \mathcal{F}^i_{\text{vit}} + \gamma^i \text{Attention}(\text{norm}(\mathcal{F}^i_{\text{vit}}), \text{norm}(\mathcal{F}^i_{\text{sp}}))$)

- Input for i-th block of the ViT
  - Query : input feature F_{vit}^i
  - Key, Value : spatial feature F_{sp}^i
  - $\gamma^i \in R^D$: balance the attention layer's output and the F_{vit}^i
    - Initialized with 0
      - Ensures that F_{vit}^i will not be modified drastically due to the injection of F_{sp}^i
        👉 making better use of the pre-trained weights of ViT

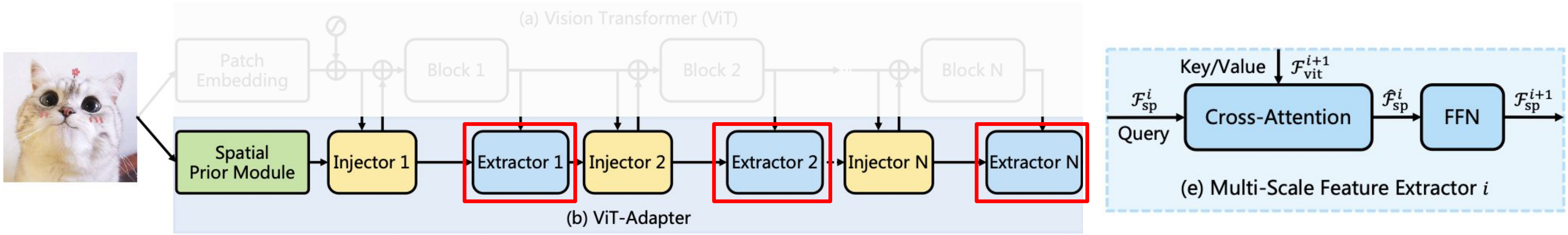


Vision Transformer Adapter

Adapter : Multi-Scale Feature Extractor

- Extract multi-scale features
  - Method : cross-attention (equation : $\hat{\mathcal{F}}_{\text{sp}}^{i} = \mathcal{F}_{\text{sp}}^{i} + \text{Attention}(\text{norm}(\mathcal{F}_{\text{sp}}^{i}), \text{norm}(\mathcal{F}_{\text{vit}}^{i+1}))$, $\mathcal{F}_{\text{sp}}^{i+1} = \hat{\mathcal{F}}_{\text{sp}}^{i} + \text{FFN}(\text{norm}(\hat{\mathcal{F}}_{\text{sp}}^{i}))$)

- Input for i-th block of the ViT
  - Query : spatial feature F_{sp}^{i}
  - Key, Value : input feature F_{vit}^{i}



Experiments

Object Detection & Instance Segmentation : Settings

- Test backbone's performance using various detector

- Detector
  - Mask R-CNN (Kaiming He et al., ICCV 2017)
  - Cascade Mask R-CNN (Zhaowei Cai & Nuno Vasconcelos, TPAMI 2019)
  - ATSS (Shifeng Zhang et al., CVPR 2020)
  - GFL (Xiang Li et al., NeurIPS 2020)

- Dataset : MS COCO 2014

- Edit L-layer ViT (to save time and memory)
  - Use 14x14 window attention except for layers spaced at an interval of L/4

- ETC.
  - AdamW optimizer(learning rate 1e-4, weight decay 0.05)
  - Training schedule : 1x(12 epochs), 3x(36 epochs)

Kaiming He et al., “Mask r-cnn”, ICCV, 2017.
Zhaowei Cai and Nuno Vasconcelos., “Cascade r-cnn: high quality object detection and instance segmentation”, TPAMI, 2019.
Jianwei Yang et al. “Focal self-attention for local-global interactions in vision transformers”, TMLR, 2022.
Xiang Li et al., “Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection”, NeurIPS, 2020.

Experiments

Object Detection & Instance Segmentation : Results

- Pre-trained weights
  - ViT-T/S/B : DeiT released ImageNet-1K weights
  - ViT-L : ImageNet-22K weights from Steinet et al.

Method	#Param (M)	Mask R-CNN 1× schedule						Mask R-CNN 3×+MS schedule					
		AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
PVT-Tiny (Wang et al., 2021)	32.9	36.7	59.2	39.3	35.1	56.7	37.3	39.8	62.2	43.0	37.4	59.3	39.9
PVTv2-B1 (Wang et al., 2022a)	33.7	41.8	64.3	45.9	38.8	61.2	41.6	44.9	67.3	49.4	40.8	64.0	43.8
ViT-T (Li et al., 2021b)	26.1	35.5	58.1	37.8	33.5	54.9	35.1	40.2	62.9	43.5	37.0	59.6	39.0
ViTDet-T (Li et al., 2022b)	26.6	35.7	57.7	38.4	33.5	54.7	35.2	40.4	63.3	43.9	37.1	60.1	39.3
ViT-Adapter-T (ours)	28.1	41.1	62.5	44.3	37.5	59.7	39.9	46.0	67.6	50.4	41.0	64.4	44.1
PVT-Small (Wang et al., 2021)	44.1	40.4	62.9	43.8	37.8	60.1	40.3	43.0	65.3	46.9	39.9	62.5	42.8
PVTv2-B2 (Wang et al., 2022a)	45.0	45.3	67.1	49.6	41.2	64.2	44.4	47.8	69.7	52.6	43.1	66.8	46.7
Swin-T (Liu et al., 2021b)	47.8	42.7	65.2	46.8	39.3	62.2	42.2	46.0	68.1	50.3	41.6	65.1	44.9
ConvNeXt-T (Liu et al., 2022)	48.1	44.2	66.6	48.3	40.1	63.3	42.8	46.2	67.9	50.8	41.7	65.0	44.9
Focal-T (Yang et al., 2021)	48.8	44.8	67.7	49.2	41.0	64.7	44.2	47.2	69.4	51.9	42.7	66.5	45.9
ViT-S (Li et al., 2021b)	43.8	40.2	63.1	43.4	37.1	59.9	39.3	44.0	66.9	47.8	39.9	63.4	42.2
ViTDet-S (Li et al., 2022b)	45.7	40.6	63.3	43.5	37.1	60.0	38.8	44.5	66.9	48.4	40.1	63.6	42.5
ViT-Adapter-S (ours)	47.8	44.7	65.8	48.3	39.9	62.5	42.8	48.2	69.7	52.5	42.8	66.4	45.9
PVTv2-B5 (Wang et al., 2022a)	101.6	47.4	68.6	51.9	42.5	65.7	46.0	48.4	69.2	52.9	42.9	66.6	46.2
Swin-B (Liu et al., 2021b)	107.1	46.9	-	-	42.3	-	-	48.6	70.0	53.4	43.3	67.1	46.7
ViT-B (Li et al., 2021b)	113.6	42.9	65.7	46.8	39.4	62.6	42.0	45.8	68.2	50.1	41.3	65.1	44.4
ViTDet-B (Li et al., 2022b)	121.3	43.2	65.8	46.9	39.2	62.7	41.4	46.3	68.6	50.5	41.6	65.3	44.5
ViT-Adapter-B (ours)	120.2	47.0	68.2	51.4	41.8	65.1	44.9	49.6	70.6	54.0	43.6	67.7	46.9
ViT-L† (Li et al., 2021b)	337.3	45.7	68.9	49.4	41.5	65.6	44.6	48.3	70.4	52.9	43.4	67.9	46.6
ViTDet-L† (Li et al., 2022b)	350.9	46.2	69.2	50.3	41.4	65.8	44.1	49.1	71.5	53.8	44.0	68.5	47.6
ViT-Adapter-L† (ours)	347.9	48.7	70.1	53.2	43.3	67.0	46.9	52.1	73.8	56.5	46.0	70.5	49.7

Various Backbone + MASK R-CNN

Method	AP^b	AP^b_{50}	AP^b_{75}	#P
Cascade Mask R-CNN 3×+MS schedule				
Swin-T (Liu et al., 2021b)	50.5	69.3	54.9	86M
Shuffle-T (Huang et al., 2021b)	50.8	69.6	55.1	86M
PVTv2-B2 (Wang et al., 2022a)	51.1	69.8	55.3	83M
Focal-T (Yang et al., 2021)	51.5	70.6	55.9	87M
ViT-S (Li et al., 2021b)	47.9	67.1	51.7	82M
ViT-Adapter-S (ours)	51.5	70.1	55.8	86M
Swin-B (Liu et al., 2021b)	51.9	70.9	57.0	145M
Shuffle-B (Huang et al., 2021b)	52.2	71.3	57.0	145M
ViT-B (Li et al., 2021b)	50.1	69.3	54.3	151M
ViT-Adapter-B (ours)	52.1	70.6	56.5	158M

Method	AP^b	AP^b_{50}	AP^b_{75}	#P
ATSS 3×+MS schedule				
Swin-T (Liu et al., 2021b)	47.2	66.5	51.3	36M
Focal-T (Yang et al., 2021)	49.5	68.8	53.9	37M
PVTv2-B2 (Wang et al., 2022a)	49.9	69.1	54.1	33M
ViT-S (Li et al., 2021b)	45.2	64.8	49.0	32M
ViT-Adapter-S (ours)	49.6	68.5	54.0	36M
GFL 3×+MS schedule				
Swin-T (Liu et al., 2021b)	47.6	66.8	51.7	36M
PVTv2-B2 (Wang et al., 2022a)	50.2	69.4	54.7	33M
ViT-S (Li et al., 2021b)	46.0	65.5	49.7	32M
ViT-Adapter-S (ours)	50.0	69.1	54.3	36M

Various backbone + Various Detector

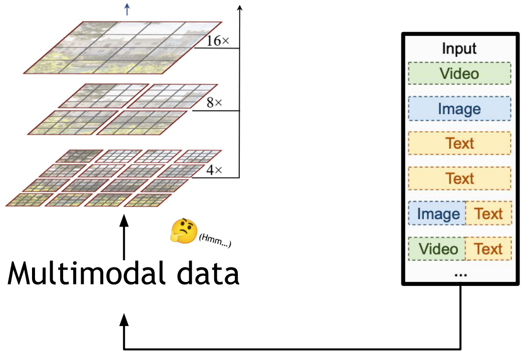
Hugo Touvron et al., "Training data-efficient image transformers & distillation through attention", ICML, 2021.
Andreas Steiner et al., "How to train your vit? data, augmentation, and regularization in vision transformers", TMLR, 2022.

Experiments

Object Detection & Instance Segmentation : Results

- With Multi-Modal Pre-training
  - Study the effect of multimodal pre-training
  - Fine-tune the ViT-Adapter-B with Mask R-CNN using different pre-trained weights
  - ViT-adapter gain performance with multimodal pre-training
    - Our method can easily derive considerable benefits from advanced multimodal pre-training (which is difficult for vision-specific models)

Method	Pre-train	AP^b	AP^m
Swin-B (Mask R-CNN 3×+MS)	ImageNet-1K	48.6	43.3
	ImageNet-22K	49.6	44.3
	Multi-Modal	N/A	N/A
ViT-Adapter-B (Mask R-CNN 3×+MS)	ImageNet-1K	49.6	43.6
	ImageNet-22K	50.5	44.6
	Multi-Modal	**51.2**	**45.3**



Xizhou Zhu et al., "Uni-Perceiver: Pre-training Unified Architecture for Generic Perception for Zero-shot and Few-shot Tasks", CVPR, 2022.
Ze Liu et al., "Swin transformer v2: Scaling up capacity and resolution", CVPR, 2022.
Ze Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows", ICCV, 2021.

Experiments

Semantic Segmentation : Settings

- Test backbone’s performance using various header

- Segmentation header
  - Semantic FPN (Alexander Kirillov et al., CVPR 2019)
  - UperNet (Tete Xiao et al., ECCV 2018)

- Dataset : ADE20K

Experiments

Semantic Segmentation : Results

- Pre-trained Weights
  - Same as object detection exp.

- Settings for each header
  - FPN : settings of PVT(Wenhai Wang et al.) and train the models for 80k iterations
  - UperNet : the settings of Swin(Ze Liu et al.) to train it for 160k iterations.

Method	Pre-train	Crop Size	Semantic FPN 80k			UperNet 160k		
			#Param	mIoU	+MS	#Param	mIoU	+MS
PVT-Tiny (Wang et al., 2021)	IN-1K	512×512	17.0M	36.6	37.3	43.2M	38.5	39.0
ViT-T (Li et al., 2021b)	IN-1K	512×512	10.2M	39.4	40.5	34.1M	41.7	42.6
ViT-Adapter-T (ours)	IN-1K	512×512	12.2M	41.7	42.1	36.1M	42.6	43.6
PVT-Small (Wang et al., 2021)	IN-1K	512×512	28.2M	41.9	42.3	54.5M	43.7	44.0
PVTv2-B2 (Wang et al., 2022a)	IN-1K	512×512	29.1M	45.2	45.7	-	-	-
Swin-T (Liu et al., 2021b)	IN-1K	512×512	31.9M	41.5	-	59.9M	44.5	45.8
Twins-SVT-S (Chu et al., 2021a)	IN-1K	512×512	28.3M	43.2	-	54.4M	46.2	47.1
ViT-S (Li et al., 2021b)	IN-1K	512×512	27.8M	44.6	45.8	53.6M	44.6	45.7
ViT-Adapter-S (ours)	IN-1K	512×512	31.9M	46.1	46.6	57.6M	46.2	47.1
Swin-B (Liu et al., 2021b)	IN-1K	512×512	91.2M	46.0	-	121.0M	48.1	49.7
Twins-SVT-L (Chu et al., 2021a)	IN-1K	512×512	103.7M	46.7	-	133.0M	48.8	50.2
ViT-B (Li et al., 2021b)	IN-1K	512×512	98.0M	46.4	47.6	127.3M	46.1	47.1
ViT-Adapter-B (ours)	IN-1K	512×512	104.6M	47.9	48.9	133.9M	48.8	49.7
Swin-B† (Liu et al., 2021b)	IN-22K	640×640	-	-	-	121.0M	50.0	51.7
Swin-L† (Liu et al., 2021b)	IN-22K	640×640	-	-	-	234.0M	52.1	53.5
ViT-Adapter-B† (ours)	IN-22K	512×512	104.6M	50.7	51.9	133.9M	51.9	52.5
ViT-Adapter-L† (ours)	IN-22K	512×512	332.0M	52.9	53.7	363.8M	53.4	54.4
ViT-Adapter-L★ (ours)	MM	512×512	332.0M	54.2	54.7	363.8M	55.0	55.4

Various backbone + Various Detector

Wenhai Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions", ICCV, 2021.
Ze Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows", ICCV, 2021.

Experiments

Comparisons With SOTA

- Combine our ViT-Adapter with SOTA detection/segmentation frameworks
  - MM : multimodal pre-training, sup : supervised pre-training
  - Plain backbone detectors/segmenters can challenge the entrenched position of hierarchical backbones

Method	Framework	Epoch	Backbone Pre-train	val AP^b	val AP^m	val (+MS) AP^b	val (+MS) AP^m	test-dev AP^b	test-dev AP^m	test-dev (+MS) AP^b	test-dev (+MS) AP^m
Swin-L	HTC++	72	IN-22K, sup	57.1	49.5	58.0	50.4	57.7	50.2	58.7	51.1
Focal-L	HTC++	36	IN-22K, sup	57.0	49.9	58.1	50.9	-	-	58.4	51.3
MViTv2-L	Cascade	50	IN-22K, sup	56.9	48.6	58.7	50.5	-	-	-	-
MViTv2-H	Cascade	50	IN-22K, sup	57.1	48.8	58.4	50.1	-	-	-	-
CBV2-Swin-L	HTC	36	IN-22K, sup	59.1	51.0	59.6	51.8	59.4	51.6	60.1	52.3
ViT-Adapter-L	HTC++	36	IN-22K, sup	56.6	49.0	57.7	49.9	57.4	50.0	58.4	50.7
Swin-L	HTC++	36	IN-1K, UM-MAE	57.4	49.8	58.7	50.9	-	-	-	-
ViTDet-L	Cascade	100	IN-1K, MAE	**59.6**	51.1	60.4	52.2	-	-	-	-
ViT-Adapter-L	HTC++	36	IN-22K, BEiT	58.4	50.8	60.2	52.2	58.9	51.3	60.4	52.5
ViT-Adapter-L	HTC++	36	MM†, BEiTv2	58.8	**51.1**	**60.5**	**52.5**	**59.5**	**51.8**	**60.9**	**53.0**

Object Detection

Method	Framework	Backbone Pre-train	Extra Pre-train	Crop Size	Iters	ADE20K val mIoU	ADE20K val +MS	#Param
Swin-L	Mask2Former	IN-22K, sup	-	640	160k	56.1	57.3	215M
Swin-L-FaPN	Mask2Former	IN-22K, sup	-	640	160k	56.4	57.7	217M
SeMask-Swin-L	Mask2Former	IN-22K, sup	-	640	160k	57.0	58.2	-
HorNet-L	Mask2Former	IN-22K, sup	-	640	160k	57.5	57.9	-
ViT-Adapter-L	Mask2Former	IN-22K, sup	-	640	160k	56.8	57.7	438M
BEiT-L	UperNet	IN-22K, BEiT	-	640	160k	56.7	57.0	441M
ViT-Adapter-L	UperNet	IN-22K, BEiT	-	640	160k	58.0	58.4	451M
BEiTv2-L	UperNet	IN-22K, BEiTv2	-	512	160k	57.5	58.0	441M
ViT-Adapter-L	UperNet	IN-22K, BEiTv2	-	512	160k	58.0	58.5	451M
ConvNeXt-XL*	Mask2Former	IN-22K, sup	COCO-Stuff, sup	896	80k	57.1	58.4	588M
Swin-L*	Mask2Former	IN-22K, sup	COCO-Stuff, sup	896	80k	57.3	58.3	434M
SwinV2-G	UperNet	IN-22K, sup	Ext-70M, sup	896	160k	59.3	59.9	3.0B
FD-SwinV2-G	UperNet	IN-22K, sup	Ext-70M, sup	896	160k	-	61.4	3.0B
Swin-L	Mask DINO	IN-22K, sup	Objects365, sup	-	160k	59.5	60.8	223M
ViT-Adapter-L	Mask2Former	IN-22K, BEiT	COCO-Stuff, sup	896	80k	59.4	60.5	571M
ViT-Adapter-L	Mask2Former	MM†, BEiTv2	COCO-Stuff, sup	896	80k	**61.2**	**61.5**	571M
BEiT-3 (w/ ViT-Adapter)	Mask2Former	MM, BEiT-3	COCO-Stuff, sup	896	80k	**62.0**	**62.8**	1.3B

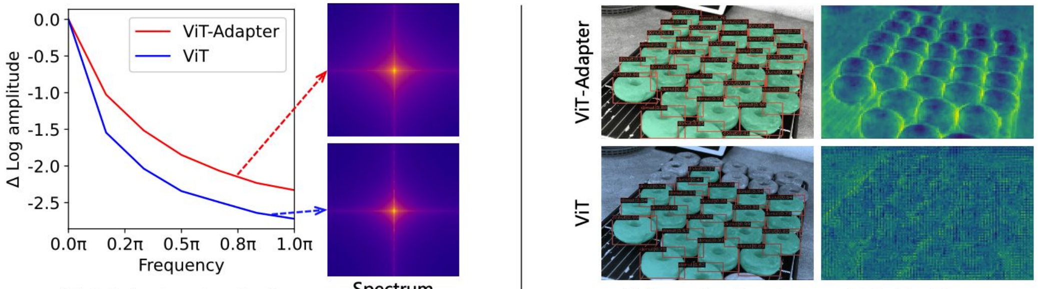
Instance Segmentation

Zhiliang Peng et al., "Beit v2: Masked image modeling with vector-quantized visual tokenizers", arXiv, 2022.

Experiments

Ablation Study

- ViT vs. ViT-Adapter Feature
    - Fourier analysis as a toolkit for visualization
        - ViT-Adapter captures more high-frequency signals than the ViT baseline
    - Stride-8 feature map
        - ViT : blurry and coarse
        - ViT-Adapter : more fine-grained and have more local edges and textures
    - Our method grafts the merit of CNN for capturing high-frequency information to ViT



(a) Relative Log Amplitudes (b) Detection Results (c) Stride-8 Feature

Experiments

Ablation Study

- Ablation for Components
  - Gradually extend the ViT-S baseline to our ViT-Adapter-S
  - Add : directly resizing and adding the spatial features from SPM

Method	Components			Interaction	Mask R-CNN 1×		
	SPM	Injector	Extractor	Mode	AP^b	AP^m	#Param
ViT-S (Li et al., 2021b)				-	40.2	37.1	43.8M
Variant 1	✓			Add	41.6	38.0	45.1M
Variant 2	✓	✓		Attention	42.6	38.8	46.6M
ViT-Adapter-S (ours)	✓	✓	✓	Attention	**44.7**	**39.9**	47.8M

- Number of Interactions
  - N : num of Interaction(Injector & Extractor) blocks

N	AP^b	AP^m	#Param
0	40.2	37.1	43.8M
1	43.2	38.9	45.5M
2	43.9	39.4	46.2M
4	**44.7**	**39.9**	47.8M
6	44.7	39.8	49.4M

Experiments

Ablation Study

- Attention Type
  - Show that our method is a general framework in which the attention mechanism is replaceable
  - Adopt ViT-Adapter-S as the basic model and study 4 different attention mechanisms
  - Deformable attention with linear complexity is more suitable for our adapter
    👉 Use deformable attention as the default configuration

Attention Mechanism	Complexity	AP^b	AP^m	FLOPs	#Param	Train Time	Memory
Global Attention (Vaswani et al., 2017)	Quadratic	43.7	39.3	1080G	50.3M	1.61s	*19.0G
CSwin Attention (Dong et al., 2021)	Linear	43.5	39.2	456G	50.3M	0.56s	15.6G
Pale Attention (Wu et al., 2022a)	Linear	44.2	39.8	458G	50.3M	0.75s	17.4G
Deformable Attention (Zhu et al., 2020)	Linear	**44.7**	**39.9**	**403G**	**47.8M**	**0.36s**	**13.7G**

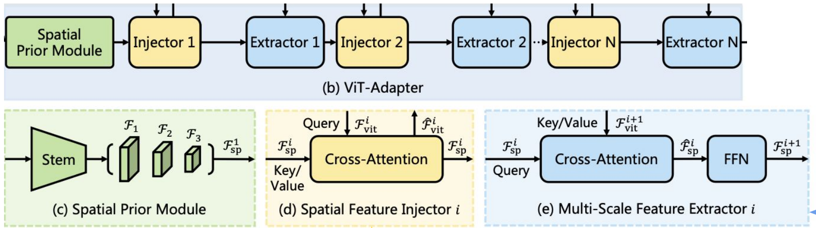
Conclusion

- Explores a new paradigm, namely ViT-Adapter
  - Bridge the gap between the plain ViT and vision-specific transformers on dense prediction tasks
  - Flexibly inject image-related inductive biases into the ViT
    👉 Reconstruct fine-grained multi-scale features required by dense predictions

- Extensive experiments on various tasks
  - Show that our method can achieve comparable or even better performance than SOTA
  - Further derive considerable benefits from advanced multimodal pre-training

Limitations

- Need a lot of computation resource
  - When calculating cross-attention…
  - If HW = 256x256, $HW/8^2 = 1024$, $HW/16^2=256$, $HW/32^2 = 64$
    👉 Need to compute 1024 + 256 + 64 = 1344 tokens as input Query or (Key&Value) in each Cross-Attention



$$\hat{\mathcal{F}}^i_{vit} = \mathcal{F}^i_{vit} + \gamma^i \text{Attention}(\text{norm}(\mathcal{F}^i_{vit}), \text{norm}(\mathcal{F}^i_{sp}))$$

$\mathcal{F}^i_{vit} \in \mathbb{R}^{\frac{HW}{16^2}\times D}$ $\mathcal{F}^i_{sp} \in \mathbb{R}^{(\frac{HW}{8^2}+\frac{HW}{16^2}+\frac{HW}{32^2})\times D}$ $\mathcal{F}^{i+1}_{vit} \in \mathbb{R}^{\frac{HW}{16^2}\times D}$

$$\hat{\mathcal{F}}^i_{sp} = \mathcal{F}^i_{sp} + \text{Attention}(\text{norm}(\mathcal{F}^i_{sp}), \text{norm}(\mathcal{F}^{i+1}_{vit}))$$

method	segmentor	pre-train	#param	#FLOPs	train time	train mem.	FPS	mIoU (ss)	mIoU (ms)
ViT-B	SETR-PUP [1]	IN-1K	98M	170G	0.16s/iter	9.5G	30.3	46.3	47.3
ViT-B	Semantic FPN [4]	IN-1K	98M	147G	0.15s/iter	5.6G	29.7	46.4	47.6
ViT-Adapter-B	Semantic FPN [4]	IN-1K	105M	183G	0.16s/iter	7.5G	26.7	**47.9**	**48.9**
ViT-L	SETR-PUP [1]	IN-22K	318M	425G	0.25s/iter	16.8G	14.0	48.6	50.1
ViT-L	Semantic FPN [4]	IN-22K	321M	414G	0.21s/iter	14.1G	15.5	51.5	52.0
ViT-Adapter-L_{light}	Semantic FPN [4]	IN-22K	324M	445G	0.23s/iter	15.2G	13.5	52.7	53.5
ViT-Adapter-L	Semantic FPN [4]	IN-22K	332M	473G	0.25s/iter	16.0G	12.9	**52.9**	**53.7**

Q & A

Thank you