



ICLR
International Conference On
Learning Representations

ICLR 2024 Potpourri

Mert Kilickaya

[Website](#)

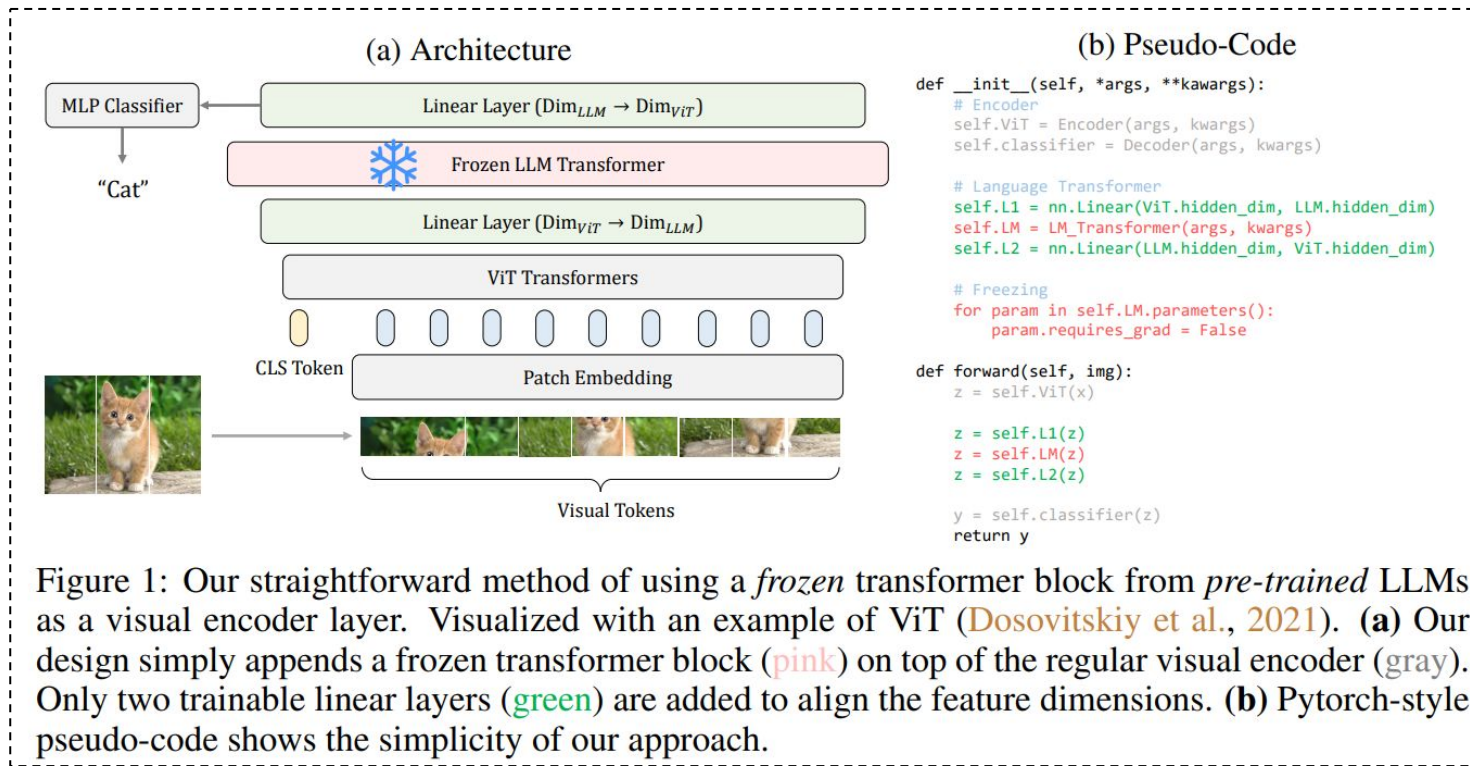
ICLR 2024 received **7500** submissions on 28th September

Pre-filtering: ~**200** papers Post-filtering: ~**20** papers

Not depth-first, but breadth-first discussion

Four Topics: LLM & Efficiency & ViT & Vision

a) LLM for Vision

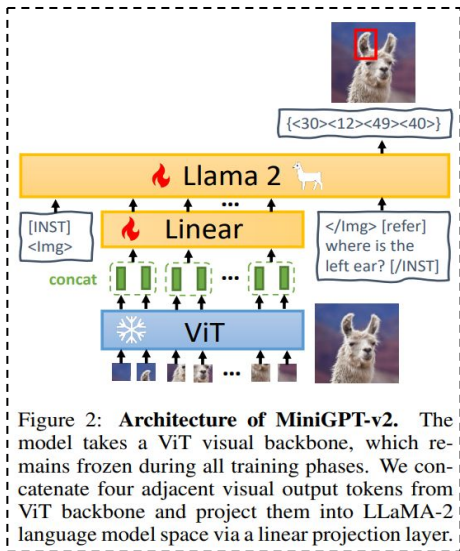


Encode image with ViT + Linear layer → Process the features with (frozen) LLM → Classify & Detect

Model	ImageNet	ImageNet-C	ImageNet-A	ImageNet-SK	ImageNet-R
ViT-T	72.1	43.9	7.7	19.6	32.3
ViT-T-LLaMA	73.2	45.8	8.7	20.6	33.8
ViT-S	80.1	57.2	20.5	28.9	42.1
ViT-S-LLaMA	80.7	58.7	22.7	30.5	42.8
ViT-B*	78.9	58.1	21.6	29.3	40.5
ViT-B-LLaMA	80.6	60.6	24.6	30.4	40.9

Inclusion of LLM robustifies the model **-But-** the improvement is not significant (~1-2%)

Method



Results on Visual Question Answering

Method	Grounding	OKVQA	GQA	VSR (zero-shot)	TextVQA (zero-shot)	IconVQA (zero-shot)	VizWiz (zero-shot)	HM (zero-shot)
Flamingo-9B	✗	44.7	-	31.8	-	-	28.8	57.0
BLIP-2 (13B)	✗	45.9	41.0	50.9	42.5	40.6	19.6	53.7
InstructBLIP (13B)	✗	-	49.5	52.1	50.7	44.8	33.4	57.5
MiniGPT-4 (13B)	✗	37.5	30.8	41.6	19.4	37.6	-	-
LLaVA (13B)	✗	54.4	41.3	51.2	38.9	43.0	-	-
Shikra (13B)	✓	47.2	-	-	-	-	-	-
Ours (7B)	✓	56.9	60.3	60.6	51.9	47.7	30.3	58.2
Ours (7B)-chat	✓	55.9	58.8	63.3	52.3	49.4	42.4	59.5

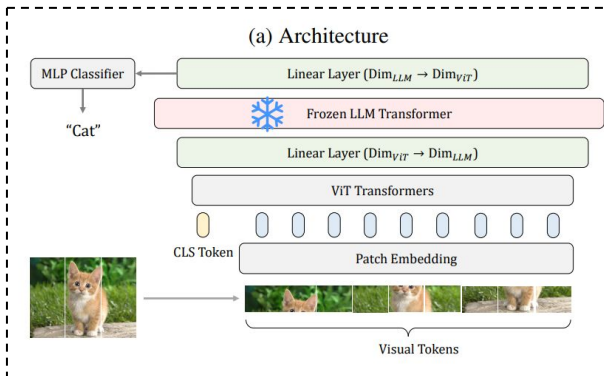
Table 3: **Results on multiple VQA tasks.** We report top-1 accuracy for each task. Grounding column indicates whether the model incorporates visual localization capability. The best performance for each benchmark is indicated in **bold**.

The model leads to a significant boost across many vision-and-language tasks.

1

Frozen-LLM

(a) Architecture



2

MiniGPT-V2

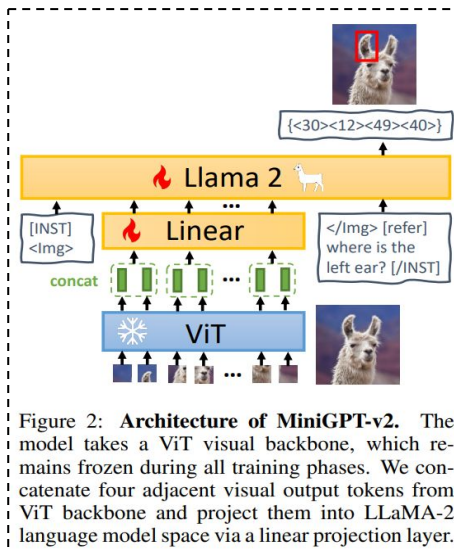
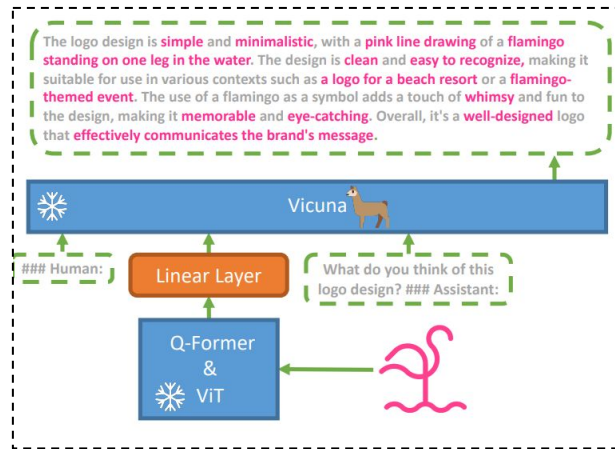


Figure 2: **Architecture of MiniGPT-v2.** The model takes a ViT visual backbone, which remains frozen during all training phases. We concatenate four adjacent visual output tokens from ViT backbone and project them into LLaMA-2 language model space via a linear projection layer.

3

MiniGPT-V4



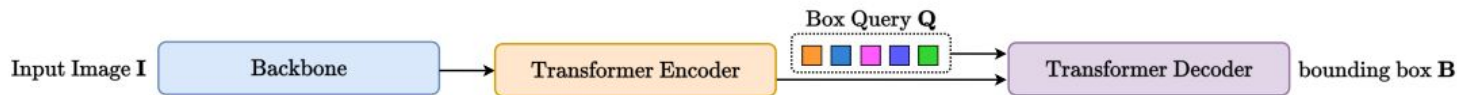
4

PLANTING A SEED OF VISION IN LARGE LANGUAGE MODEL

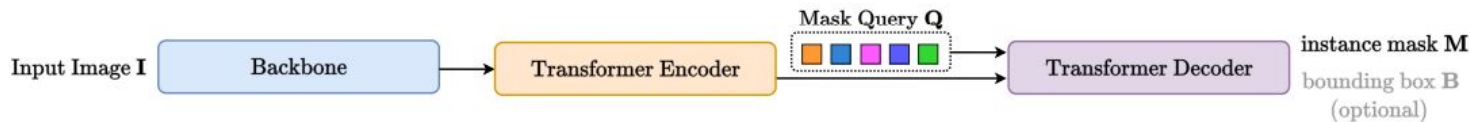
5

LINGUISTIC IMAGE UNDERSTANDING

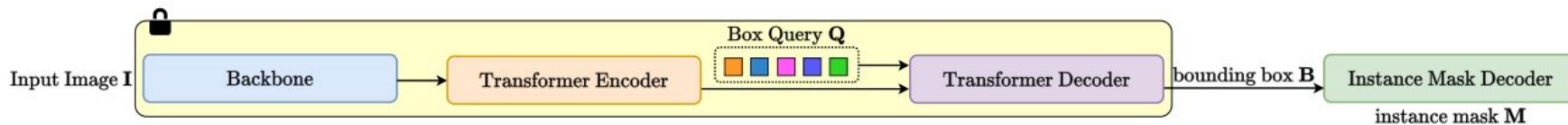
b) Efficient Training



(a) DETR-based object detection framework



(b) DETR-based instance segmentation framework

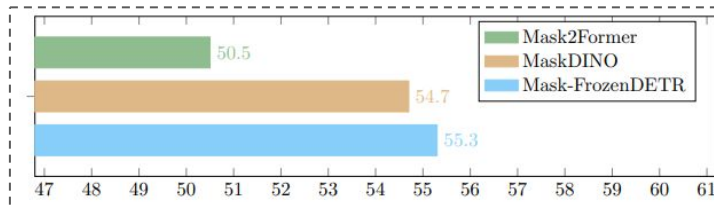


(c) Our approach: Mask Frozen-DETR

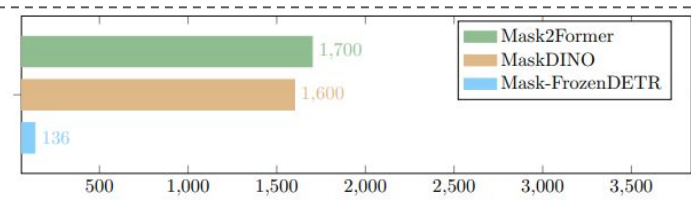
- i. Take a pre-trained detector
- ii. Freeze it
- iii. Fine-tune the mask head
- iv. SOTA segmenter in few epochs

MASK FROZEN-DETR: HIGH QUALITY INSTANCE SEGMENTATION WITH ONE GPU

Mask AP



GPU Hours

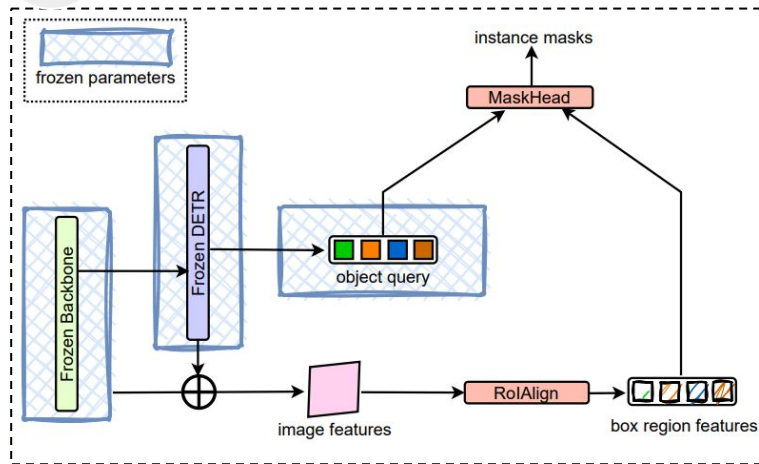


Extensive Comparison

method	backbone	#epochs	Object365	AP ^{box}	AP ^{mask}	AP ^{mask} ₅₀	AP ^{mask} ₇₅	AP ^{mask} _S	AP ^{mask} _M	AP ^{mask} _L	GPU Hours
K-Net-N256 (Zhang et al., 2021)	R50	36	✗	—	38.6	60.9	41.0	19.1	42.0	57.7	—
QueryInst (Fang et al., 2021)	Swin-L	50	✗	56.1	48.9	74.0	53.9	30.8	52.6	68.3	—
Mask2Former (Cheng et al., 2021a)	R50	50	✗	—	43.7	—	—	23.4	47.2	64.8	502
Mask2Former (Cheng et al., 2021a)	Swin-L	100	✗	—	50.1	—	—	29.9	53.9	72.1	1,700
Mask DINO (Li et al., 2022)	R50	50	✗	50.5	46.0	68.9	50.3	26.0	49.3	65.5	1,404
Mask DINO (Li et al., 2022)	Swin-L	50	✗	58.3	52.1	76.5	57.6	32.9	55.4	72.5	2,400
Mask Frozen- \mathcal{H} -DETR	R50	6	✗	49.9	44.1	66.2	47.8	24.4	47.0	62.6	49
Mask Frozen- \mathcal{H} -DETR	Swin-L	6	✗	59.1	51.9	75.8	57.2	31.6	55.1	71.6	179

Mask Frozen-DETR has competitive accuracy while training **>10x** faster.

1 Mask-Frozen DETR



2 HOW TO FINE-TUNE VISION MODELS WITH SGD

	ID accuracy	OOD accuracy
SGD	90.0%	67.9%
AdamW	(+2.1%)	(+8.1%)
SGD (freeze-embed)	(+2.1%)	(+8.0%)
SGD (freeze-embed, no mom.)	(+2.2%)	(+9.0%)

(b) Performance of different fine-tuning methods on a CLIP ViT-B/16 averaged over 5 distribution shift datasets.

3 LESS IS MORE: SELECTIVE LAYER FINE-TUNING WITH SUBTUNING

SubTuning



c) Vision-Transformers

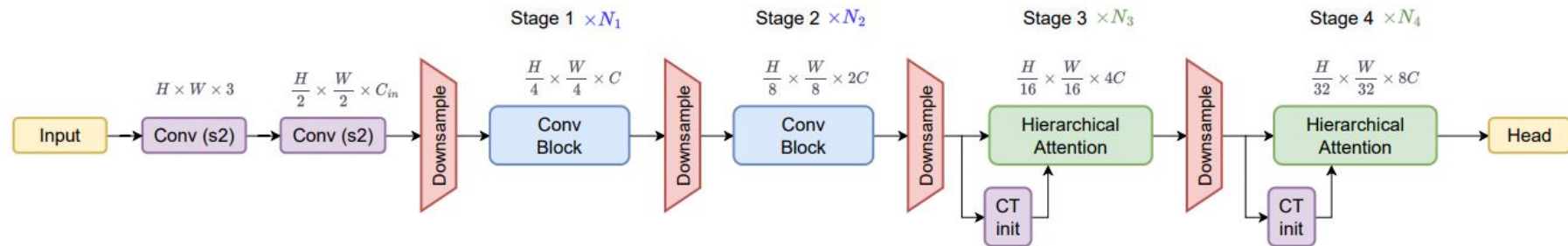


Figure 3: Overview of the FasterViT architecture. We use a multi-scale architecture with CNN and transformer-based blocks in stages 1, 2 and 3, 4, respectively. Best viewed in color.

CT Init: Carrier token initialization

FasterViT combines Convolutional blocks (stage 1-2) with Transformer blocks (stage 3-4).

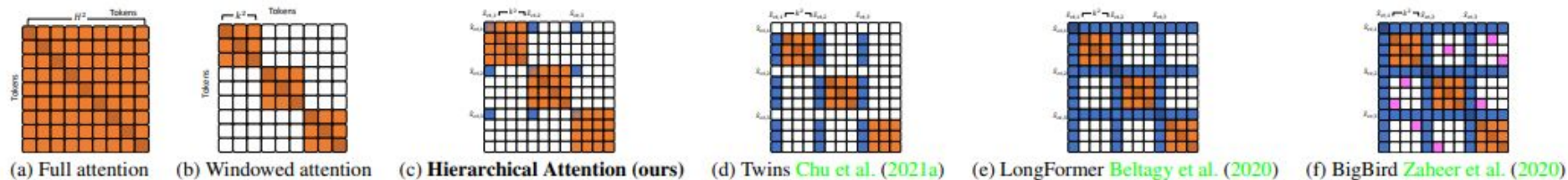


Figure 5: Attention map comparison for a feature map of size $H \times H \times d$. \square - no attention, \blacksquare - normal token attention, \blacksquare - carrier token attention, \blacksquare - random token attention. Full attention (a) has complexity of $O(H^4d)$, windowed attention significantly reduces it to $O(k^2H^2d)$ but lacks global context.

Hierarchical attention extends windowed attention via carrier tokens to pool from distinct windows.

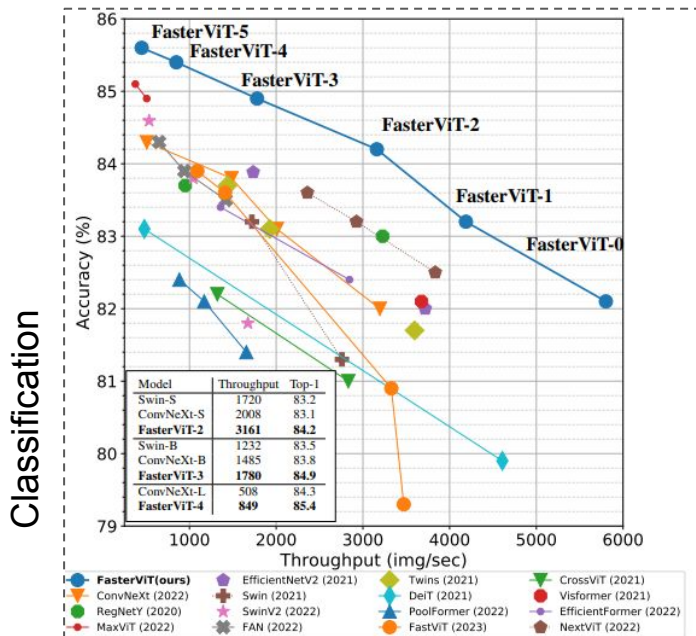
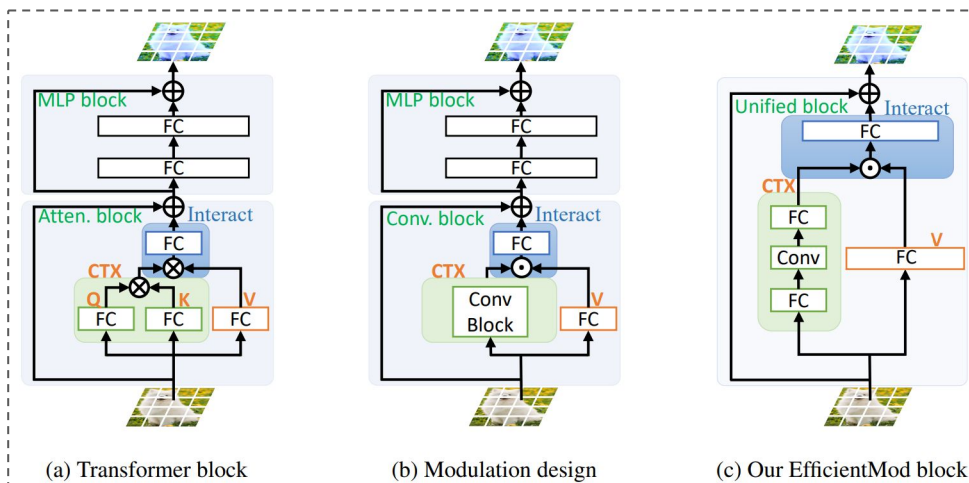


Figure 1: Comparison of image throughput and imageNet-1K Top-1 accuracy. For all models, throughput is measured on A100 GPU with batch size of 128.

Model	Throughput	FLOPs (G)	IoU(ss/ms)
Swin-T Liu et al. (2021)	350	945	44.5/45.8
ConvNeXt-T Liu et al. (2022b)	363	939	- /46.7
FasterViT-2	377	974	47.2/48.4
Twins-SVT-B Chu et al. (2021a)	204	-	47.7/48.9
Swin-S Liu et al. (2021)	219	1038	47.6/49.5
ConvNeXt-S Liu et al. (2022b)	234	1027	- /49.6
FasterViT-3	254	1076	48.7/49.7
Twins-SVT-L Chu et al. (2021a)	164	-	48.8/50.2
Swin-B Liu et al. (2021)	172	1188	48.1/49.7
ConvNeXt-B Liu et al. (2022b)	189	1170	- /49.9
FasterViT-4	202	1290	49.1/50.3

Table 4: Semantic segmentation on ADE20K Zhou et al. (2017) with UPerNet Xiao et al. (2018).

FasterViT yields the highest accuracy-efficiency tradeoff for classification/segmentation tasks (ADE20K).

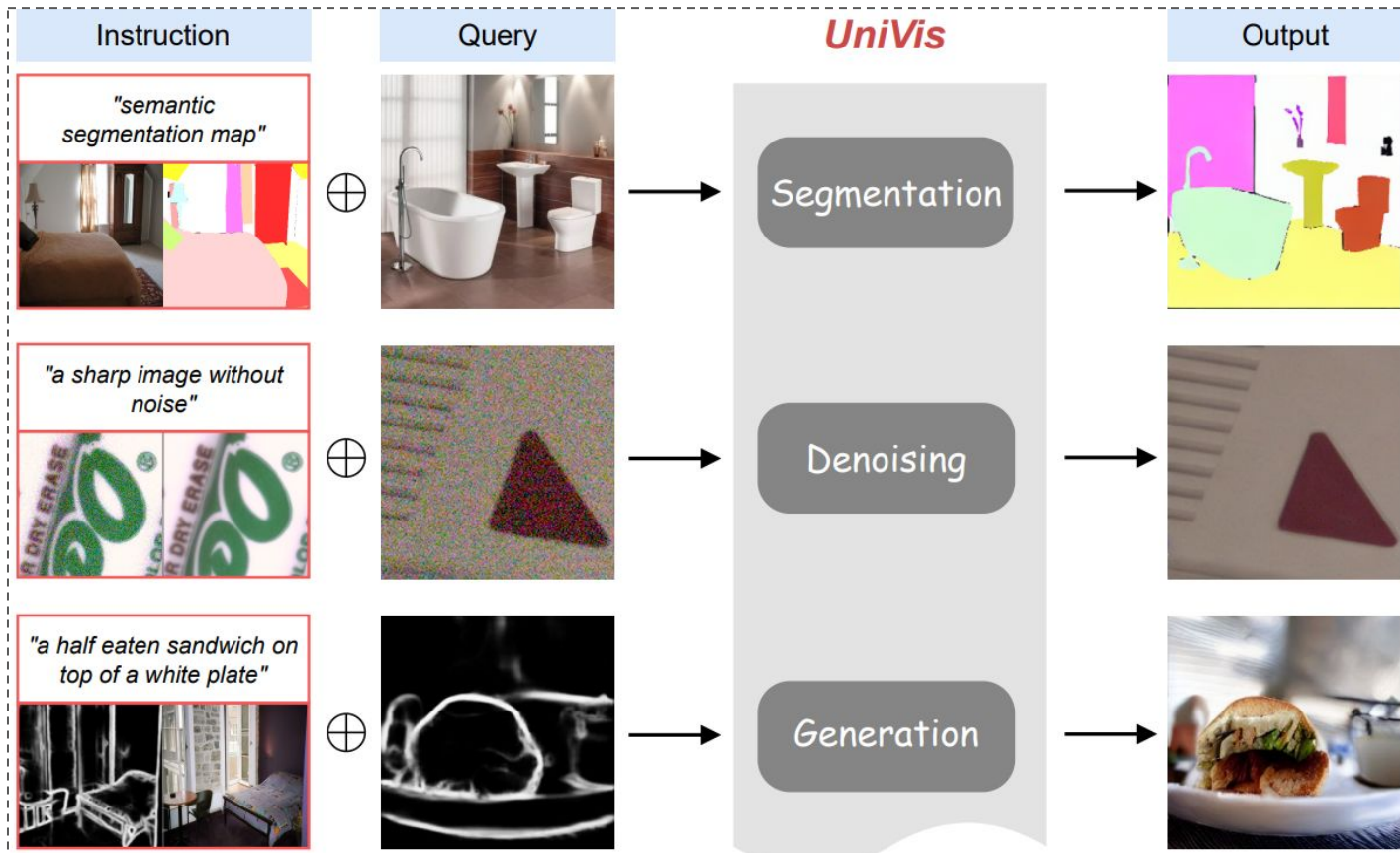


Model	Top-1(%)	Latency (ms)		Params (M)	FLOPs (G)	Size.
		GPU	CPU			
MobileNetV2×1.0 (2018)	71.8	2.1	3.8	3.5	0.3	224 ²
FasterNet-T0 (2023)	71.9	2.5	6.8	3.9	0.3	224 ²
EdgeViT-XXS (2022)	74.4	8.8	15.7	4.1	0.6	224 ²
MobileOne-S1 (2023)	74.6 (75.9)	1.5	6.9	4.8	0.8	224 ²
MobileViT-XS (2022)	74.8	4.1	21.0	2.3	1.1	256 ²
EfficientFormerV2-S0 (2023b)	73.7 (75.7)	3.3	10.7	3.6	0.4	224 ²
EfficientMod-xxs	76.0	3.0	10.2	4.7	0.6	224 ²
MobileNetV2×1.4 (2018)	74.7	2.8	6.0	6.1	0.6	224 ²
DeiT-T (2021a)	74.5	2.7	16.5	5.9	1.2	224 ²
FasterNet-T1 (2023)	76.2	3.3	12.9	7.6	0.9	224 ²
EfficientNet-B0 (2019)	77.1	3.4	10.9	5.3	0.4	224 ²
MobileOne-S2 (2023)	- (77.4)	2.0	10.0	7.8	1.3	224 ²
EdgeViT-XS (2022)	77.5	11.8	21.4	6.8	1.1	224 ²
MobileViTv2-1.0 (2023)	78.1	5.4	30.9	4.9	1.8	256 ²
EfficientFormerV2-S1 (2023b)	77.9 (79.0)	4.5	15.4	6.2	0.7	224 ²
EfficientMod-xs	78.3	3.6	13.4	6.6	0.8	224 ²
PoolFormer-s12 (2022a)	77.2	5.0	22.3	11.9	1.8	224 ²
FasterNet-T2 (2023)	78.9	4.4	18.4	15.0	1.9	224 ²
EfficientFormer-L1 (2022)	79.2	3.7	19.7	12.3	1.3	224 ²
MobileFormer-508M (2022b)	79.3	13.4	142.5	14.8	0.6	224 ²
MobileOne-S4▲ (2023)	- (79.4)	4.8	26.6	14.8	3.0	224 ²
MobileViTv2-1.5 (2023)	80.4	7.2	59.0	10.6	4.1	256 ²
EdgeViT-S (2022)	81.0	20.5	34.7	13.1	1.9	224 ²
EfficientFormerV2-S2 (2023b)	80.4 (81.6)	7.3	26.5	12.7	1.3	224 ²
EfficientMod-s	81.0	5.5	23.5	12.9	1.4	224 ²

Replace Query-Key subnets with an FC-Conv-FC block → Low latency across GPU & CPU.

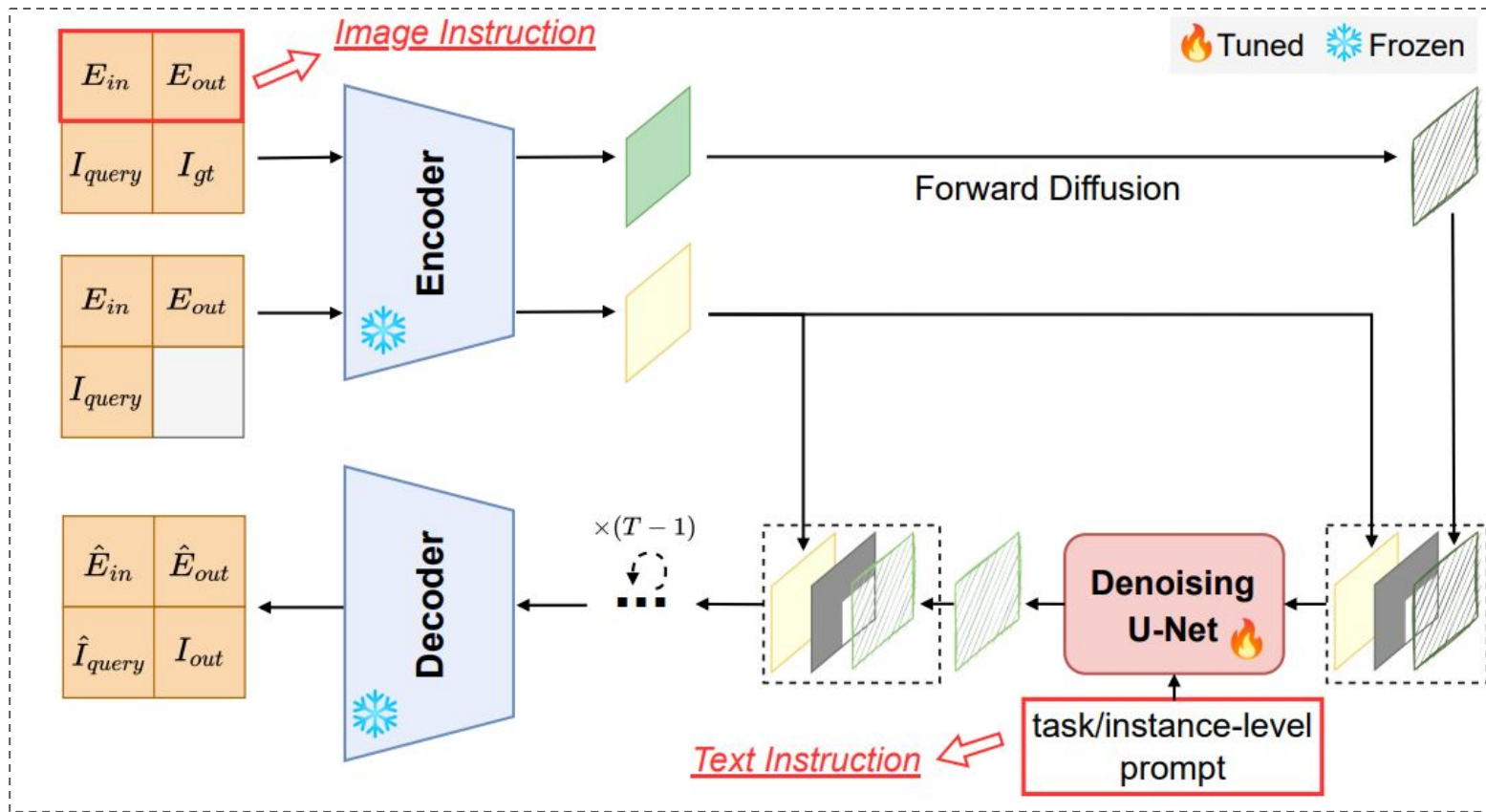
d) Vision

UNIVIS: A UNIVERSAL FRAMEWORK FOR COMPUTER VISION TASKS



Perform many different vision tasks (segmentation, denoising, generation) by differing the instruction.

UNIVIS: A UNIVERSAL FRAMEWORK FOR COMPUTER VISION TASKS



The model builds upon Stable Diffusion, and generates unified input-output prompts for different tasks.

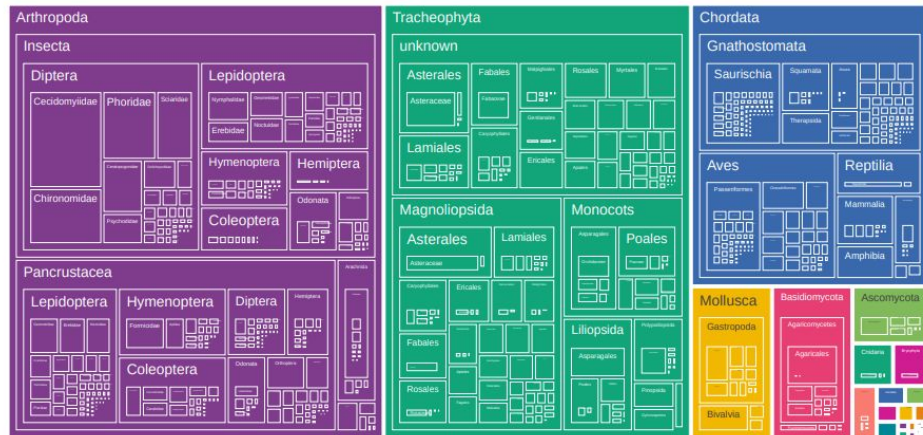


Figure 1: **SLiMe**. Using just one user-annotated image with various granularity (as shown in the leftmost column), **SLiMe** learns to segment different unseen images in accordance with the same granularity (as depicted in the other columns).

SLIME can learn to segment objects at different granularity from a single image, interactively.

3

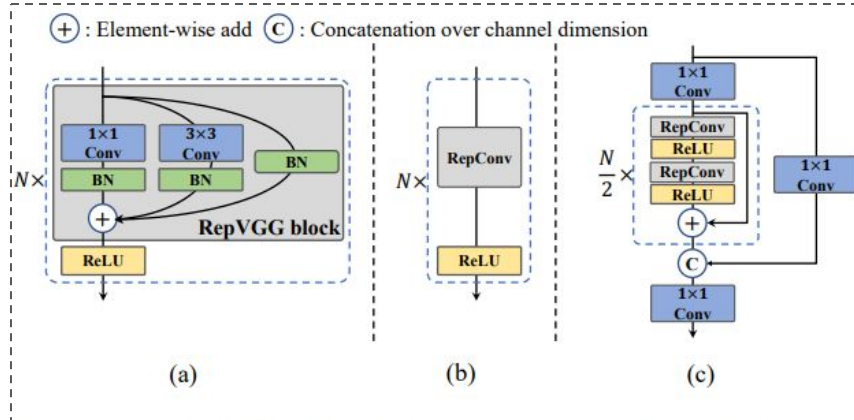
BIOCLIP: A VISION FOUNDATION MODEL FOR THE TREE OF LIFE



(a) Treemap of the different phyla in TREEOFLIFE-10M. Different colors are different phyla; nested boxes represent classes, orders, etc.

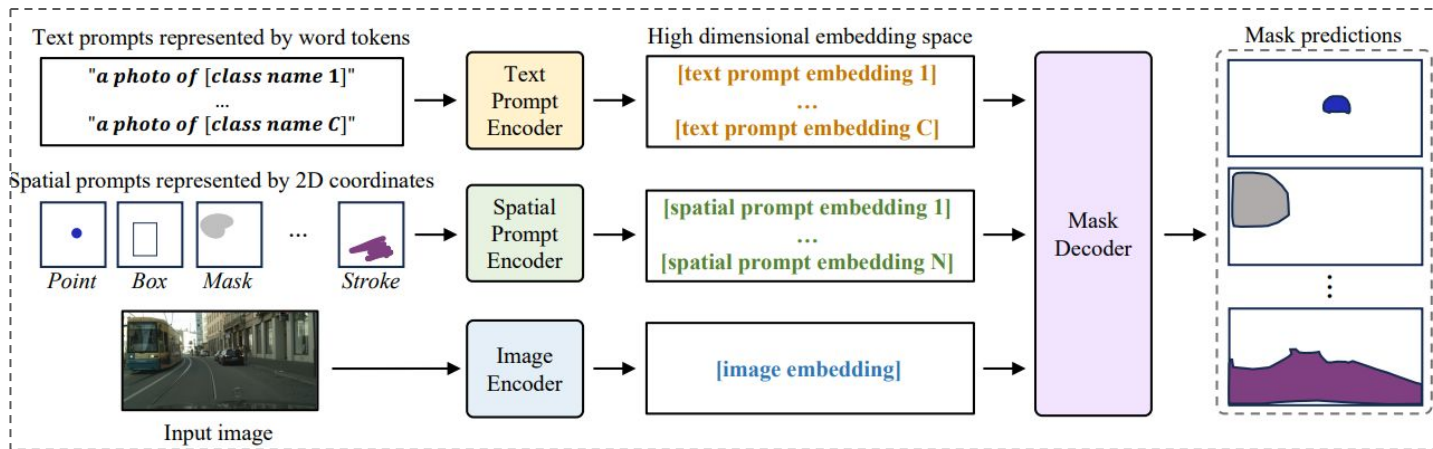
4

YOLOV6: OBJECT DETECTION FOR INDUSTRY

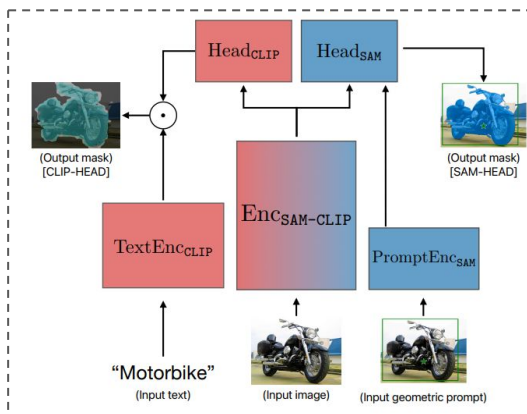


Train with (a) RepVGG & Infer with (c) RepConv.

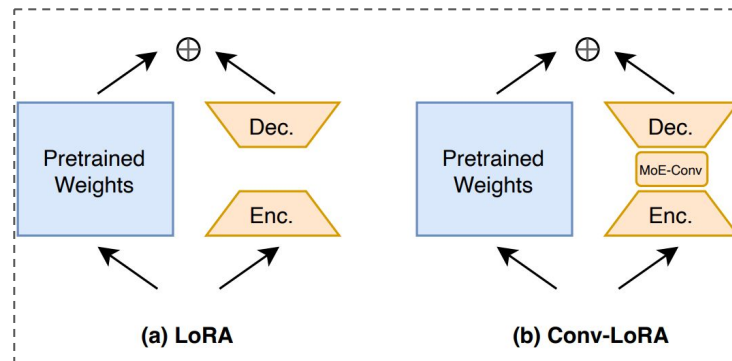
5 LEARNING TO PROMPT SAM



6 SAM-CLIP: MERGING CLIP and SAM



7 CONV-LORA: Efficient Fine-tuning of SAM



Other Papers

- 1 [THE ALL-SEEING PROJECT: TOWARDS PANOPTIC VISUAL RECOGNITION AND UNDERSTANDING OF THE OPEN WORLD](#)
- 2 [IN DEFENSE OF SEMI-SUPERVISED LEARNING: SELF SUPERVISION IS NOT ALL YOU NEED](#)
- 3 [SIMPLIFYING SELF-SUPERVISED OBJECT DETECTION PRETRAINING](#)
- 4 [LARGE LANGUAGE MODELS AS OPTIMIZERS](#)
- 5 [IMAGE COMPRESSION IS AN EFFECTIVE OBJECTIVE FOR VISUAL REPRESENTATION LEARNING](#)
- 6 [AUTOVP: AN AUTOMATED VISUAL PROMPTING FRAMEWORK AND BENCHMARK](#)

Code Repositories

Paper	Code
FROZEN TRANSFORMERS IN LANGUAGE MODELS ARE EFFECTIVE VISUAL ENCODER LAYERS	Link
MiniGPT-V2	Link
MiniGPT-V4	Link
PLANTING A SEED OF VISION IN LARGE LANGUAGE MODEL	Link
FASTERVIT: FAST VISION TRANSFORMERS WITH HIERARCHICAL ATTENTION	Link
EFFICIENT MODULATION FOR VISION NETWORKS	Link
SLIME: SEGMENT LIKE ME	Link
YOLOV6: OBJECT DETECTION FOR INDUSTRY	Link

Discussion

~Model Interactivity~

Soon, will most models become interactive (with the help of an LLM)?

Can vision annotations turn into human language form rather than spatial (e.g. bounding box)?

~Model Efficiency~

Replace FC blocks with Convs for faster ViT variants → Can this be automated?

~Training Efficiency~

How to find the best (few) layers to optimize for our downstream tasks and data?

Thanks! Questions?