



MMA : Multi-Modal Adapter for Vision-Language Models

Lingxiao Yang¹, Ru-Yuan Zhang², Yanchen Wang³, Xiaohua Xie^{1*}

¹Sun Yat-sen University, ²Shanghai Jiao Tong University, ³Stanford University

Paper Review

2025. 9. 10. Wed.

중앙대학교 첨단영상대학원 메타버스융합학과 FoVLAB

Hongseok Cho

>> Contents

- 1** Introduction
- 2** Proposed Method : MMA
- 3** Experiments
- 4** Ablation & Analysis
- 5** Conclusion

>> Overview

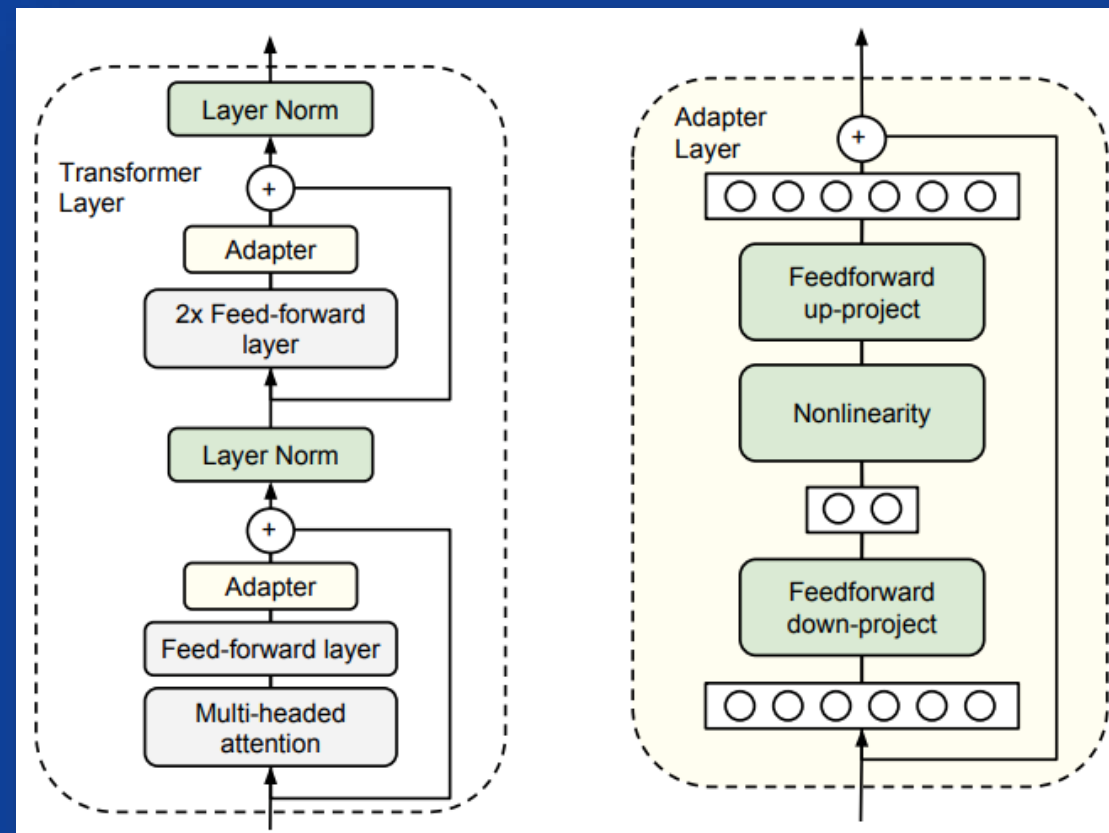
- VLMs are powerful, but adapting them to downstream tasks is challenging
- This led to the emergence of PET (Parameter-Efficient Tuning), including prompt learning and adapter tuning (Representative methods include CoOp/CoCoOp/Clip-Adapter)
- **However**, prompt learning only modifies the text branch, and adapter methods mainly adjust the vision branch
- As a result, they lack effective cross-modal alignment
- **To address this, MMA (Multi-modal Adapter) is proposed**

>> Introduction

□ Background

✓ Parameter-Efficient Tuning(PET)

- Large-scale VLMs (e.g., CLIP) are powerful but hard to adapt due to size
 - Full fine-tuning is costly and prone to overfitting in few-shot setups
 - PET = Adapting large models using few additional parameters
- ✓ Two main PET paradigms:
- Prompt Learning (e.g., CoOp, CoCoOp)
 - Adapter Tuning (e.g., Clip-Adapter, Tip-Adapter)



[Parameter-Efficient Transfer Learning for NLP (Houlsby et al., 2019)]

>> Introduction


❑ Background

✓ Prompt Learning

✓ CoOP(Context Optimization)

✓ CoCoOP(Conditional CoOP)



Flowers102	Prompt	Accuracy
	a photo of a [CLASS].	60.86
	a flower photo of a [CLASS].	65.81
	a photo of a [CLASS], a type of flower.	66.14
	$[V]_1 [V]_2 \dots [V]_M$ [CLASS].	94.51



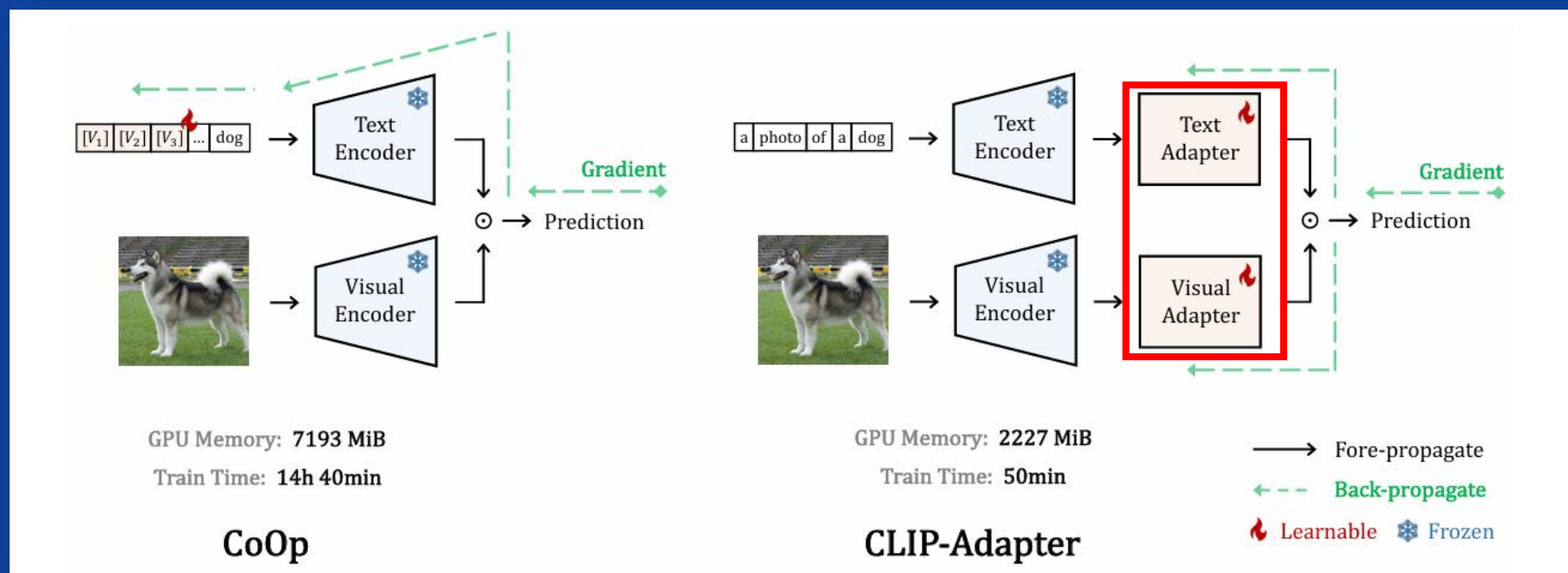
CoOp	CoCoOp
$[v_1] [v_2] \dots [v_M]$ [arrival gate].	$[v_1(x)] [v_2(x)] \dots [v_M(x)]$ [arrival gate].
\vdots	\vdots
$[v_1] [v_2] \dots [v_M]$ [cathedral].	$[v_1(x)] [v_2(x)] \dots [v_M(x)]$ [cathedral].

>> Introduction

□ Background

✓ Adapter Tuning - CLIP Adapter

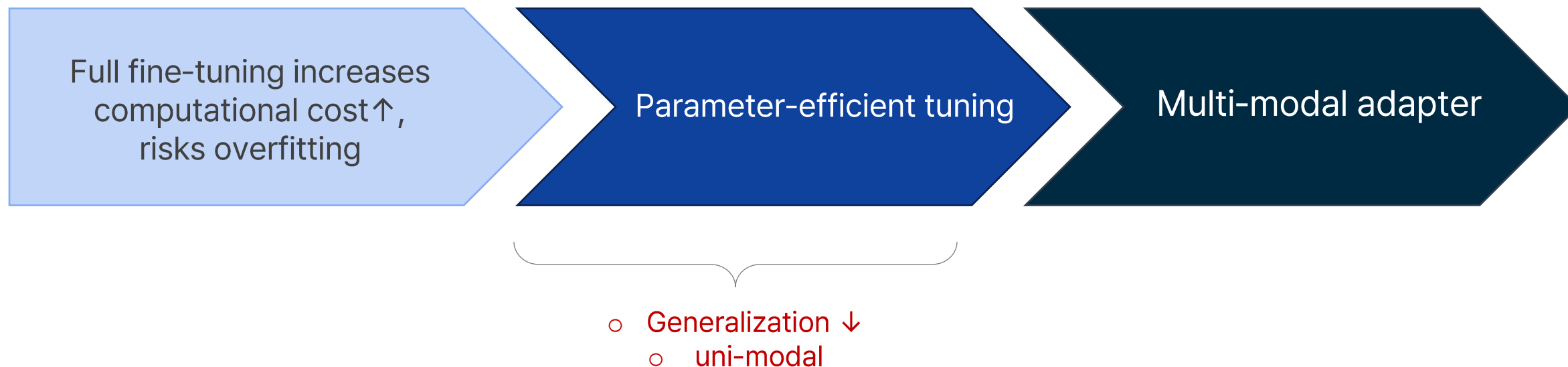
- CLIP-Adapter adds lightweight adapters after frozen CLIP encoders for both vision and text
- This method avoids backpropagation through the entire model, reducing memory and time costs
- Adapter tuning is efficient and maintains good performance with minimal training



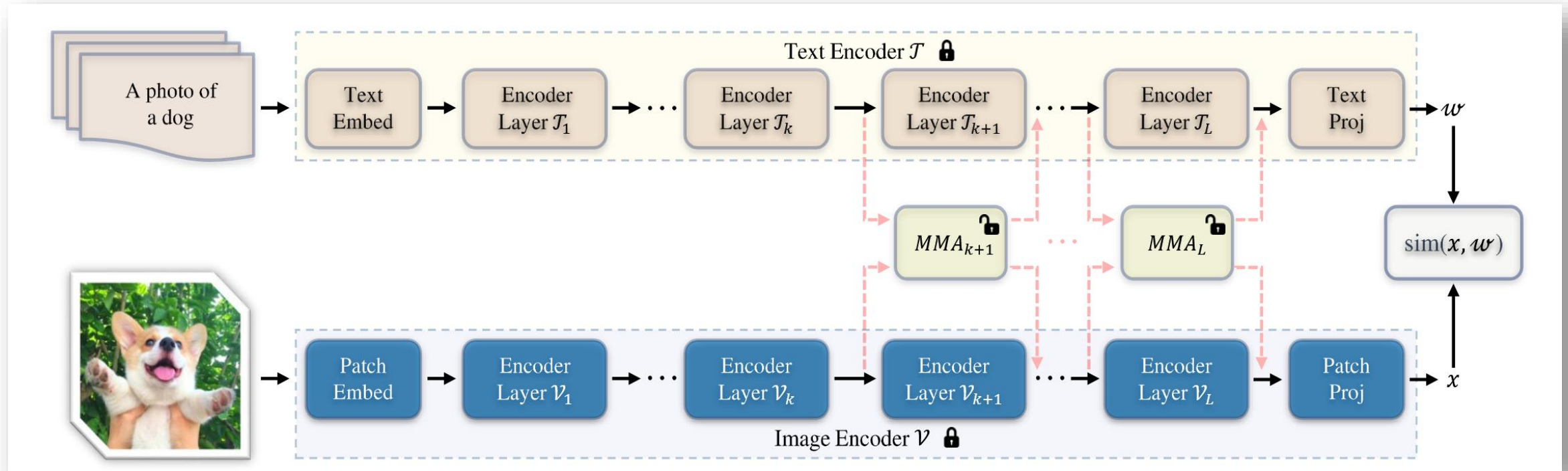
>> Introduction

❑ Related Works

✓ Efficient Transfer Learning for VLMs

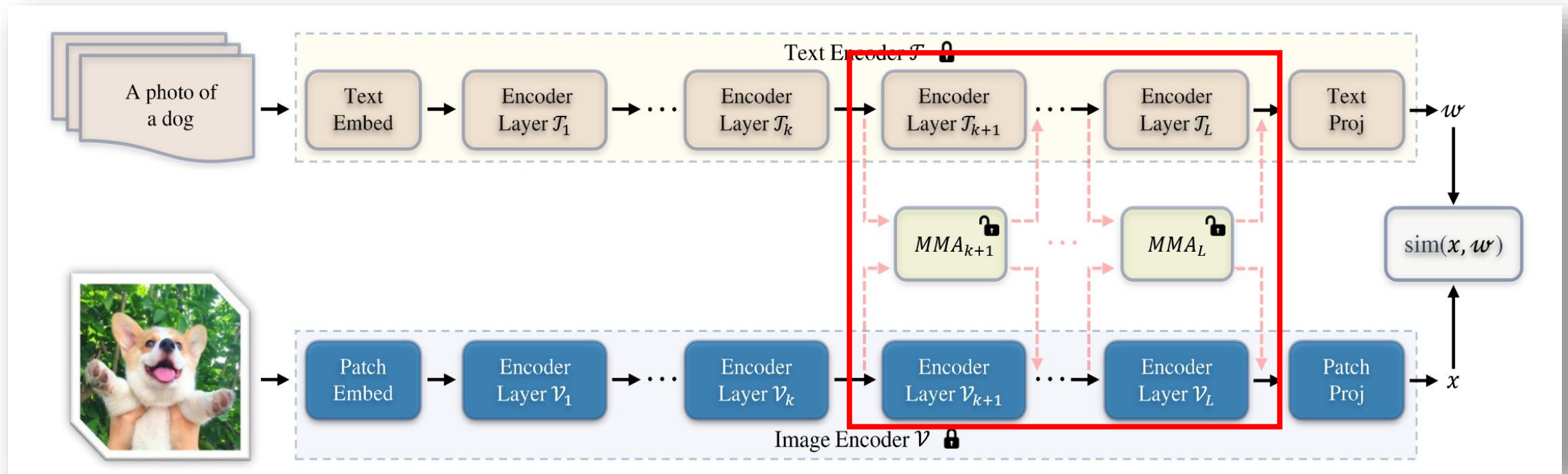


>> Proposed Method : MMA : Multi-modal adapter



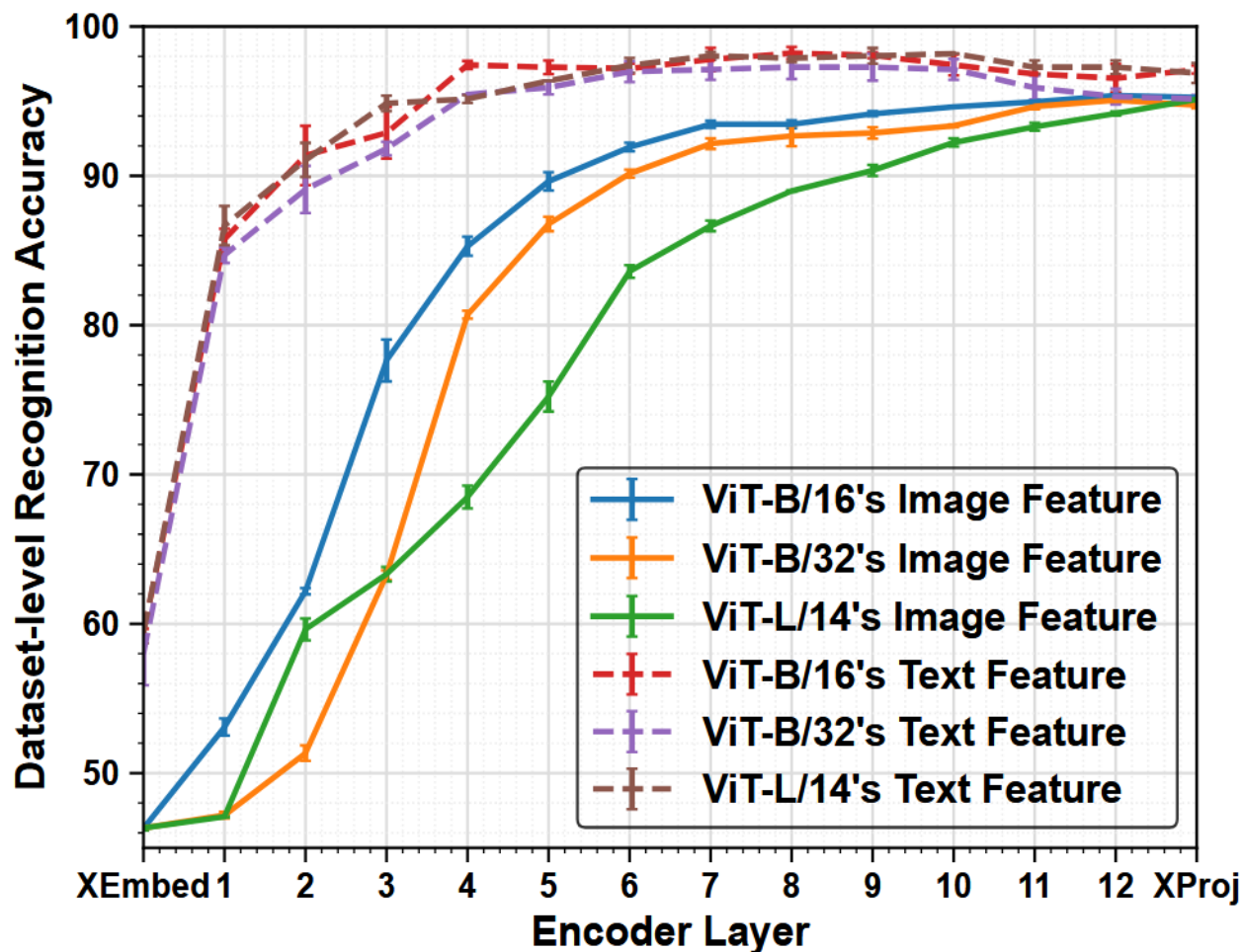
>> Proposed Method : MMA

- MMA builds on the CLIP architecture by inserting adapters only into the top $k \sim L$ layers of the text and image encoders
- The encoders are frozen, only the adapters are trained, ensuring parameter efficiency
- This design allows for minimal modification while maximizing alignment performance

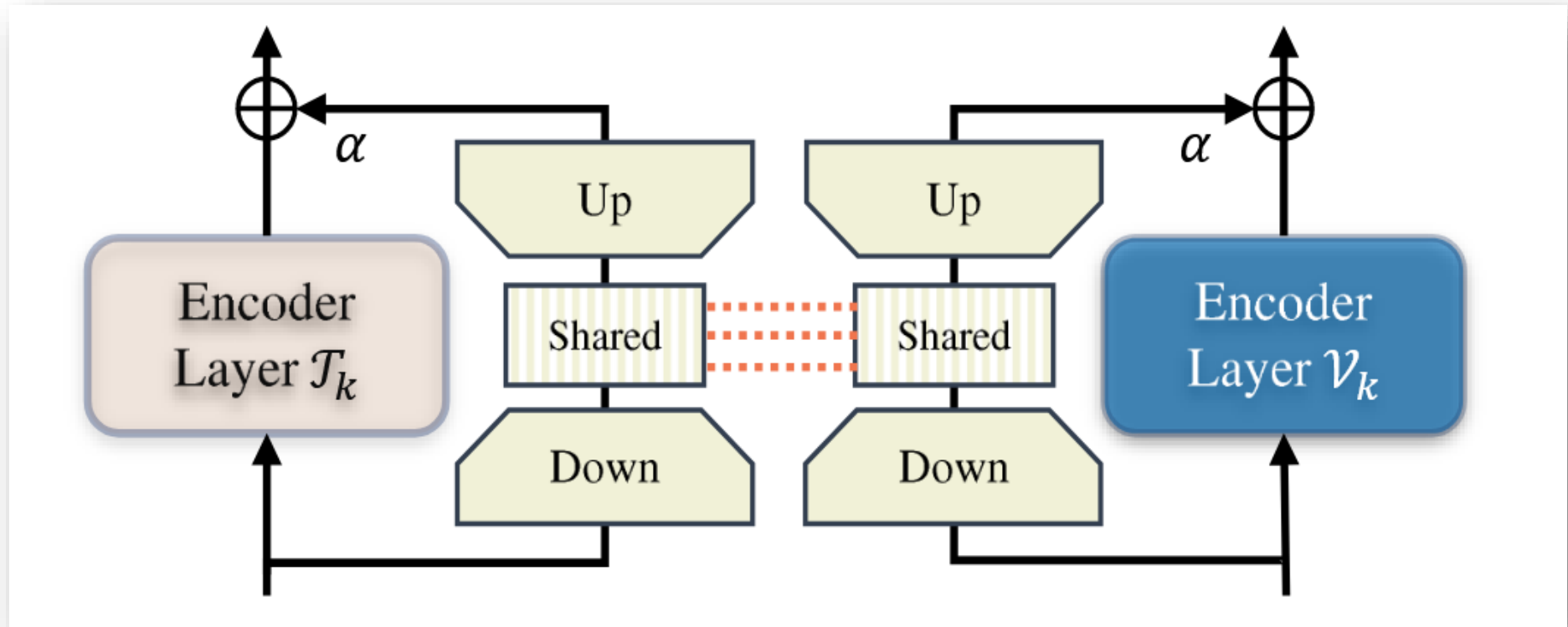


>> Proposed Method : MMA

✓ Why High Layers Only?



>> Proposed Method : MMA



>> Experiments

□ Setting

✓ Generalization from Base-to-Novel Classes

- **Purpose** : Evaluating whether a model trained on one dataset also performs well on datasets from other domains
- **Dataset** :
 - **General Object Recognition**: ImageNet, Caltech101
 - **Fine-Grained Recognition**: *OxfordPets*, StanfordCars, Flowers102, Food101, FGVC Aircraft
 - **Other Domains**: SUN397 (scenes), DTD (textures), EuroSAT (satellite), UCF101 (action)

✓ Cross-dataset Evaluation

- **Purpose** : Evaluating whether a model trained on one dataset also performs well on datasets from other domains
- **Dataset** : ImageNet

✓ Domain generalization

- **Purpose** : Measuring how robust the model is to domain shift (Out-of-distribution evaluation)
- **Dataset** : ImageNet

>> Experiments

✓ Generalization from Base-to-Novel Classes

Methods	Average			ImageNet			Caltech101			OxfordPets		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP [ICML2021] [50]	69.34	74.22	71.70	72.43	68.14	70.22	96.84	94.00	95.40	91.17	97.26	94.12
CoOp [ICV2022] [84]	82.69	63.22	71.66	76.47	67.88	71.92	98.00	89.81	93.73	93.67	95.29	94.47
CoOpOp [CVPR2022] [85]	80.47	71.69	75.83	75.98	70.43	73.10	97.96	93.81	95.84	95.20	97.69	96.43
ProDA [CVPR2022] [43]	81.56	72.30	76.65	75.40	70.23	72.72	98.27	93.23	95.68	95.43	97.83	96.62
KgCoOp [CVPR2023] [67]	80.73	73.60	77.00	75.83	69.96	72.78	97.72	94.39	96.03	94.65	97.76	96.18
MaPLe [CVPR2023] [33]	82.28	75.14	78.55	76.66	70.54	73.47	97.74	94.36	96.02	95.43	97.76	96.58
LASP [CVPR2023] [4]	82.70	74.90	78.61	76.20	70.95	73.48	98.10	94.24	96.16	95.90	97.93	96.90
LASP-V [CVPR2023] [4]	83.18	76.11	79.48	76.25	71.17	73.62	98.17	94.33	96.43	95.73	97.87	96.79
RPO [ICCV2023] [38]	81.13	75.00	77.78	76.60	71.57	74.00	97.97	94.37	96.03	94.63	97.50	96.05
MMA [this work]	83.20	76.80	79.87	77.31	71.00	74.02	98.40	94.00	96.15	95.40	98.07	96.72

$$*HM = \frac{2 \cdot \text{Base} \cdot \text{Novel}}{\text{Base} + \text{Novel}} \quad (\text{Ex. Base}=90, \text{Novel}=50 \rightarrow \text{HM}=64.3)$$

Methods	StanfordCars			Flowers102			Food101			FGVCAircraft		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP [ICML2021] [50]	63.37	74.89	68.65	72.08	77.80	74.83	90.10	91.22	90.66	27.19	36.29	31.09
CoOp [ICV2022] [84]	78.12	60.40	68.13	97.60	59.67	74.06	88.33	82.26	85.19	40.44	22.30	28.75
CoOpOp [CVPR2022] [85]	70.49	73.59	72.01	94.87	71.75	81.71	90.70	91.29	90.99	33.41	23.71	27.74
ProDA [CVPR2022] [43]	74.70	71.20	72.91	97.70	68.68	80.66	90.30	88.57	89.43	36.90	34.13	35.46
KgCoOp [CVPR2022] [67]	71.76	75.04	73.36	95.00	74.73	83.65	90.50	91.70	91.09	36.21	33.55	34.83
MaPLe [CVPR2022] [33]	72.94	74.00	73.47	95.92	72.46	82.56	90.71	92.05	91.38	37.44	35.61	36.50
LASP [CVPR2022] [4]	75.17	71.60	73.34	97.00	74.00	83.95	91.20	91.70	91.44	34.53	30.57	32.43
LASP-V [CVPR2022] [4]	75.23	71.77	73.46	97.17	73.53	83.71	91.20	91.90	91.54	38.05	33.20	35.46
RPO [ICCV2023] [38]	73.87	75.53	74.69	94.13	76.67	84.50	90.33	90.83	90.58	37.33	34.20	35.70
MMA [this work]	78.50	73.10	75.70	97.77	75.93	85.48	90.13	91.30	90.71	40.57	36.33	38.33

Methods	SUN397			DTD			EuroSAT			UCF101		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP [ICML2021] [50]	69.36	75.35	72.23	53.24	59.90	56.37	56.48	64.05	60.03	70.53	77.50	73.85
CoOp [ICV2022] [84]	80.60	65.89	72.51	79.44	41.18	54.24	92.19	54.74	68.69	84.69	56.05	67.46
CoOpOp [CVPR2022] [85]	79.74	76.86	78.27	77.01	56.00	64.85	87.49	60.04	71.21	82.33	73.45	77.64
ProDA [CVPR2022] [43]	78.67	76.93	77.79	80.67	56.48	66.44	83.90	66.00	73.88	85.23	71.97	78.04
KgCoOp [CVPR2023] [67]	80.29	76.53	78.36	77.55	54.99	64.35	85.64	64.34	73.48	82.89	76.67	79.65
MaPLe [CVPR2023] [33]	80.82	78.70	79.75	80.36	59.18	68.16	94.07	73.23	82.35	83.00	78.66	80.77
LASP [CVPR2023] [4]	80.70	78.60	79.63	81.40	58.60	68.14	94.60	77.78	85.36	84.77	78.03	81.26
LASP-V [CVPR2023] [4]	80.70	79.30	80.00	81.10	62.57	70.64	95.00	83.37	88.86	85.53	78.20	81.70
RPO [ICCV2023] [38]	80.60	77.80	79.18	76.70	62.13	68.61	86.63	68.97	76.79	83.67	75.43	79.34
MMA [this work]	82.27	78.57	80.38	83.20	65.63	73.38	85.46	82.34	83.87	86.23	80.03	82.20

>> Experiments

✓ Cross-Dataset Evaluation setting

- MMA achieves the highest average accuracy (66.61) across 10 datasets in the cross-dataset generalization setting
- It consistently performs well across diverse domains, surpassing CoOp, CoCoOp, MaPLe, and PromptSRC

Methods	ImageNet	Catech101	OxfordPets	StanfordCars	Flowers101	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
CoOp [IJCV2022] [84]	71.51	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp [CVPR2022] [85]	71.02	94.43	90.14	65.32	71.88	86.06	22.94	67.36	45.73	45.37	68.21	65.74
MaPLe [CVPR2023] [33]	70.72	93.53	90.49	65.57	72.23	86.20	24.74	67.01	46.49	48.06	68.69	66.30
PromptSRC [ICCV2023] [34]	71.27	93.60	90.25	65.70	70.25	86.15	23.90	67.10	46.87	45.50	68.75	65.81
MMA [this work]	71.00	93.80	90.30	66.13	72.07	86.12	25.33	68.17	46.57	49.24	68.32	66.61

>> Experiments

✓ Domain Generalization setting

- MMA achieves the best performance on 3 out of 4 domain-shifted datasets, showing strong robustness to out-of-distribution data
- It outperforms CLIP, CoOp, CoCoOp, and MaPLe in most settings while maintaining high accuracy on ImageNet

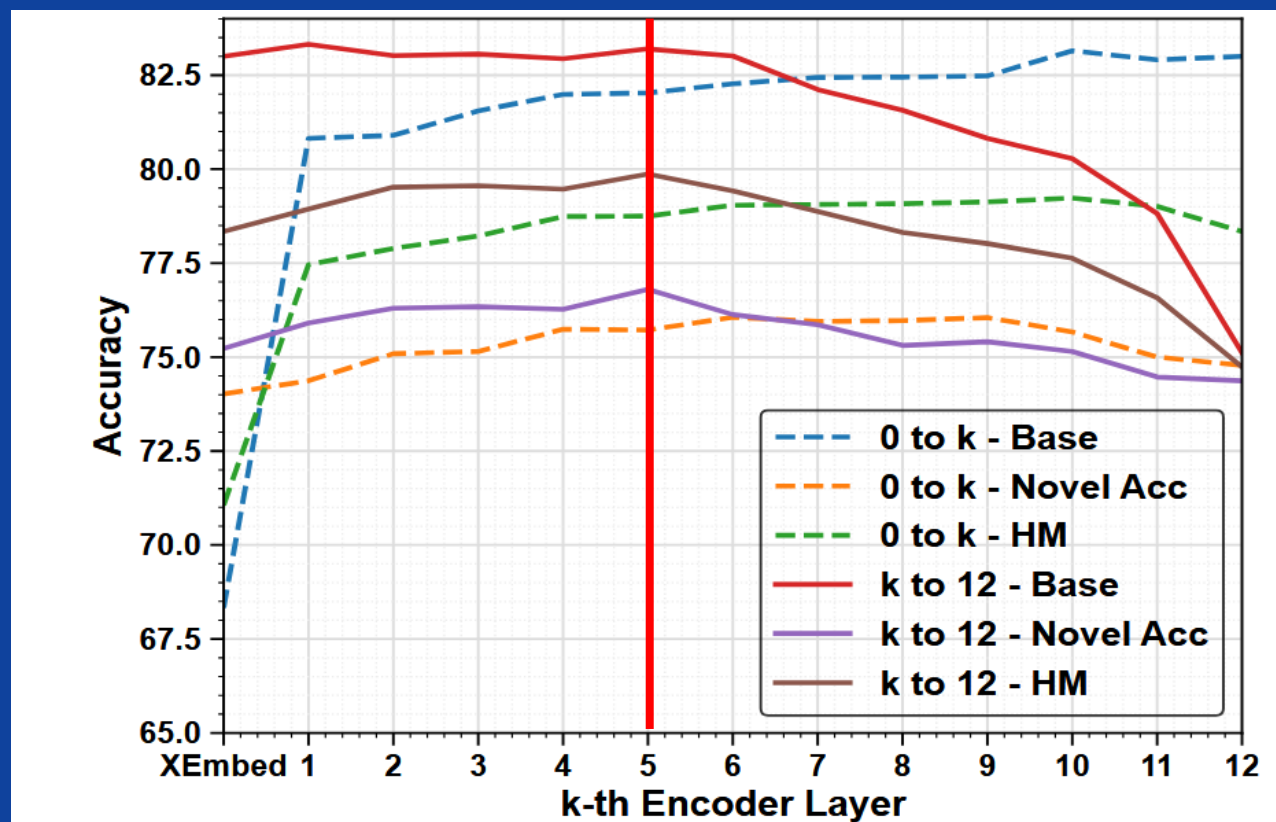
Methods	ImageNet	-V2	-S	-A	-R
CLIP [ICML2021] [50]	66.73	60.83	46.15	47.77	73.96
CoOp [IJCV2022] [84]	71.51	64.20	47.99	49.71	75.21
CoCoOp [CVPR2022] [85]	71.02	64.07	48.75	50.63	76.18
MaPLe [CVPR2023] [33]	70.72	64.07	49.15	50.90	76.98
MMA [this work]	71.00	64.33	49.13	51.12	77.32

Notation	Name	Description
-V2	ImageNet-V2	ImageNet의 재구성 버전으로, 데이터 분포가 다름
-S	ImageNet-Sketch	스케치 스타일의 이미지, 시각적 형태만 유지
-A	ImageNet-A	ImageNet의 어려운 예시 들로 구성된 벤치마크 (adversarial-like samples)
-R	ImageNet-Rendition	예술적 스타일/렌더링으로 변형된 이미지들 (e.g., cartoon, painting 등)

>> Ablation & Analysis

✓ Different choices of adding our proposed multi-modal units

- Lower layers → discrimination ↑, generalization ↓
- Higher layers → generalization ↑, but too high → base performance ↓
- Starting from layer **k = 5** provides the best trade-off, achieving the highest harmonic mean of 79.87



>> Ablation & Analysis

✓ Variants of Adding MMA

- Using adapters in both vision and language branches performs better than uni-modal setups
- Adding a shared projection layer further improves alignment and boosts HM score

(a) Performance with Different Model Variants				(b) Dimensions of Shared Layers				(c) Scaling Factor α			
Model Variants	Base	Novel	HM	Dims	Base	Novel	HM	α	Base	Novel	HM
Only L-Adapter	80.36	75.81	78.02	8	82.66	76.17	79.28	0.0001	79.40	75.57	77.44
Only V-Adapter	80.39	74.18	77.16	16	82.80	76.48	79.52	0.0005	81.81	76.08	78.84
No SharedProj	82.43	76.21	79.20	32	83.20	76.80	79.87	0.001	83.20	76.80	79.87
FCAA [1]	79.11	75.64	77.34	64	83.41	76.17	79.63	0.005	83.80	75.37	79.36
MMA	83.20	76.80	79.87	128	82.98	76.54	79.58	0.01	84.27	74.32	78.98

>> Ablation & Analysis

✓ Variants of Adding MMA

- Mid-sized shared layers (dim ≈ 32) offer the best generalization
- Too large dimensions cause overfitting, hurting performance on novel classes

(a) Performance with Different Model Variants				(b) Dimensions of Shared Layers				(c) Scaling Factor α			
Model Variants	Base	Novel	HM	Dims	Base	Novel	HM	α	Base	Novel	HM
Only L-Adapter	80.36	75.81	78.02	8	82.66	76.17	79.28	0.0001	79.40	75.57	77.44
Only V-Adapter	80.39	74.18	77.16	16	82.80	76.48	79.52	0.0005	81.81	76.08	78.84
No SharedProj	82.43	76.21	79.20	32	83.20	76.80	79.87	0.001	83.20	76.80	79.87
FCAA [1]	79.11	75.64	77.34	64	83.41	76.17	79.63	0.005	83.80	75.37	79.36
MMA	83.20	76.80	79.87	128	82.98	76.54	79.58	0.01	84.27	74.32	78.98

>> Ablation & Analysis

✓ Variants of Adding MMA

- $\alpha = 0.001$ yields the best trade-off between base and novel accuracy
- Too high or too low α values harm either generality or adaptability

(a) Performance with Different Model Variants				(b) Dimensions of Shared Layers				(c) Scaling Factor α			
Model Variants	Base	Novel	HM	Dims	Base	Novel	HM	α	Base	Novel	HM
Only L-Adapter	80.36	75.81	78.02	8	82.66	76.17	79.28	0.0001	79.40	75.57	77.44
Only V-Adapter	80.39	74.18	77.16	16	82.80	76.48	79.52	0.0005	81.81	76.08	78.84
No SharedProj	82.43	76.21	79.20	32	83.20	76.80	79.87	0.001	83.20	76.80	79.87
FCAA [1]	79.11	75.64	77.34	64	83.41	76.17	79.63	0.005	83.80	75.37	79.36
MMA	83.20	76.80	79.87	128	82.98	76.54	79.58	0.01	84.27	74.32	78.98

>> Ablation & Analysis

✓ Fine-tuning last few layers

- Tuning last CLIP layers boosts base but hurts novel accuracy
- More tuning leads to overfitting
- MMA offers a better balance with fewer updates

Layer	12	10→12	8→12	5→12	MMA
Base	80.77	83.02	83.77	83.21	83.20
Novel	74.08	74.55	73.77	70.95	76.80
HM	77.28	78.56	78.45	76.59	79.87

>> Conclusion

- **Limitation**
 - **Although MMA achieves state-of-the-art performance on average, it underperforms competing methods on certain tasks or datasets**
 - **Moreover, the evaluation is limited to classification tasks, excluding more complex downstream applications such as generation or multimodal reasoning**

Conclusion

- **Adapting large VLMs like CLIP to downstream tasks is challenging due to limited data and many trainable parameter**
- **The proposed MMA enhances cross-modal alignment by being inserted only into higher layers of the vision and language encoders**
- **MMA outperforms existing methods in generalization to novel classes, new datasets, and unseen domains**

감사합니다