L^3Seg: Lean Linear Layers for Language-Guided Vision Transformer in Medical Image Segmentation

Rahul Bhardwaj[1], Utkarsh Yashwant Tambe[2], Debanga Raj Neog[1]

[1]Mehta Family School of Data Science & Artificial Intelligence, Indian Institute of Technology Guwahati, India

[2]Department of Data Science & Business Systems, SRM Institute of Science & Technology, Kattankulathur, India

NEURAL INFORMATION PROCESSING SYSTEMS

Introduction

Problem Statement:

- Vision-Language models are heavy (params, FLOPs).
- Need cross-modality generalization.

Motivation:

- Compute is concentrated in dense linear projections.
- Fine-tuning/PEFT change few weights while base matrix multiplications still run, keeping compute high and adaptation limited.

Key Contributions:

- Replace all dense projections with L^3
- Trainables: $O(d_{in}d_{out}) \rightarrow O(r(d_{in}+d_{out}))$

Algorithm 1 Lean Linear Layer ($\mathbf{L}^3$)

Input: $x \in \mathbb{R}^{B\times N\times d_{\text{in}}}$
Frozen base: $W_0 \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$, $b_0 \in \mathbb{R}^{d_{\text{out}}}$
B: batch size, N: number of tokens, d: feature dimension
class LeanLinearLayer(*Module*):
  def __init__(self, d_{in}, d_{out}, r, W_0, b_0):
    super().__init__()
    self.$W_0 \leftarrow W_0$ $[d_{\text{out}}, d_{\text{in}}]$, frozen
    self.$b_0 \leftarrow b_0$ $[d_{\text{out}}]$, frozen
    # trainable low-rank factors
    Notation: $\mathcal{N}(0,10^{-3})_{m\times n} = \texttt{randn}(m,n) \times 10^{-3}$
    Notation: $\mathbf{0}_{m\times n} = \texttt{zeros}(m,n)$
    self.$A_g \leftarrow \mathcal{N}(0,10^{-3})^{d_{\text{in}}\times r}$
    self.$B_g \leftarrow \mathbf{0}_{r\times d_{\text{out}}}$
    self.$A_b \leftarrow \mathcal{N}(0,10^{-3})^{d_{\text{in}}\times r}$
    self.$B_b \leftarrow \mathbf{0}_{r\times d_{\text{out}}}$
  def forward(self, x):
    # 1. frozen baseline
    $y_0 \leftarrow xW_0^{\top} + b_0$ $[B, N, d_{\text{out}}]$
    # 2. low-rank scale and shift
    $\gamma \leftarrow (xA_g)B_g$ $[B, N, d_{\text{out}}]$
    $\beta \leftarrow (xA_b)B_b$ $[B, N, d_{\text{out}}]$
    # 3. scaled–offset fusion
    return $(1+\gamma) \odot y_0 + \beta$

Method

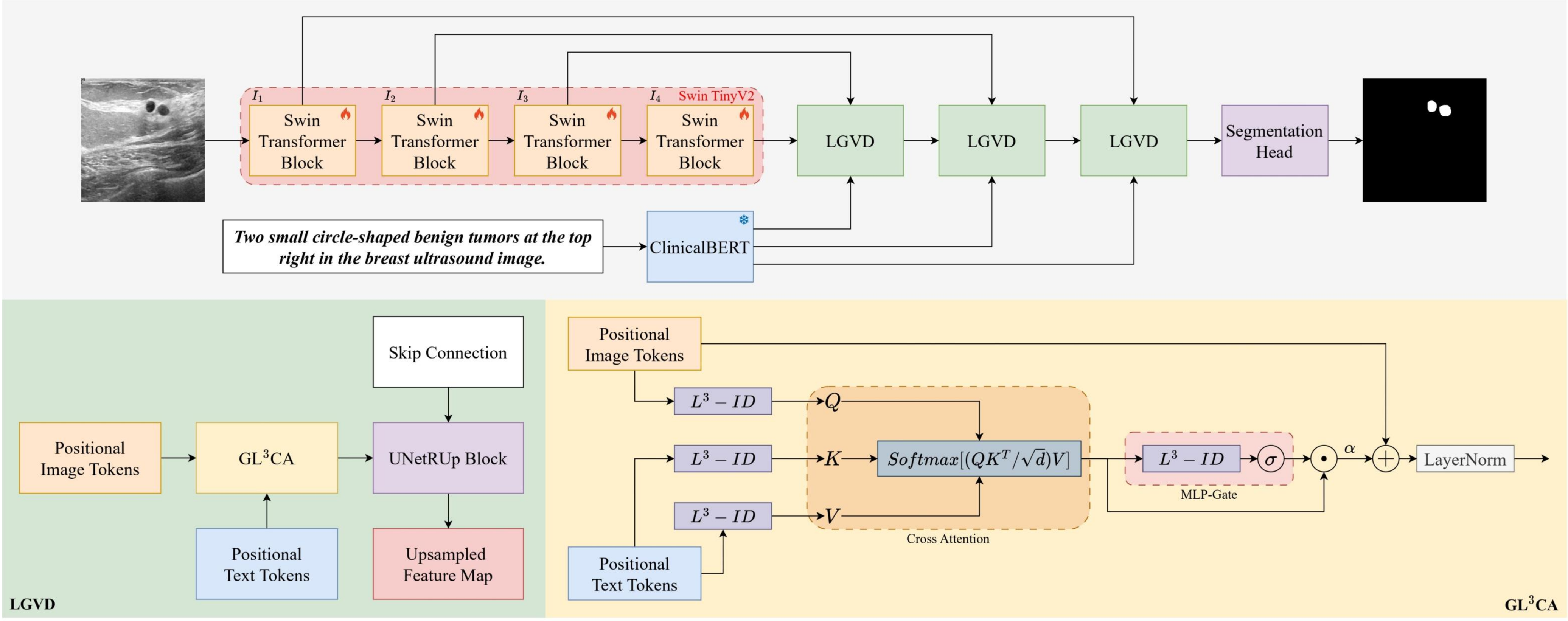

Fig. 1. Overview of L³Seg: Language-Guided Vision Decoder fuses image and text using Gated L³ Cross-Attention

Comparative Analysis

Table 1. Quantitative Comparison on QaTa-COV19 (X-ray), Kvasir-SEG (endoscopy) and BUSI (ultrasound) dataset. CNN-based (◇), SAM-based (¶), and hybrid CNN-Transformer (†).

Method	Venue	Text	Params (M) ↓	FLOPs (G) ↓	QaTa-COV19 (XRay)		Kvasir-SEG (Endoscopy)		BUSI (Ultrasound)	
					Dice(%) ↑	mIoU(%) ↑	Dice(%) ↑	mIoU(%) ↑	Dice(%) ↑	mIoU(%) ↑
UNet◇ [21]	MICCAI'15	×	<u>14.8</u>	50.3	79.02	69.46	81.83	74.60	57.28	49.19
UNet++◇ [27]	IEEE TMI'19	×	74.5	94.6	79.62	70.25	82.10	74.43	63.46	56.59
Swin-UNet† [3]	ECCV'22	×	82.3	67.3	78.07	68.34	85.90	77.56	63.67	55.54
H2Former† [9]	IEEE TMI'23	×	33.7	24.6	77.86	68.35	80.03	72.23	63.72	56.71
SAM¶ [13]	ICCV'23	×	93.6	50.9	71.85	56.06	77.83	70.72	49.93	33.27
SAM-Adapter¶ [4]	ICCV'23	×	104.3	55.2	84.76	73.55	83.42	71.55	77.47	63.22
CLIPSeg† [17]	CVPR'22	✓	150.0	23.0	78.92	71.55	83.71	76.02	62.06	57.91
TGANet◇ [22]	MICCAI'22	✓	19.8	41.9	79.87	70.75	<u>89.51</u>	<u>82.49</u>	69.33	62.32
Ariadne's Thread† [26]	MICCAI'23	✓	44.0	22.4	<u>89.78</u>	<u>81.45</u>	87.61	77.95	<u>79.36</u>	<u>65.78</u>
LViT† [14]	IEEE TMI'23	✓	29.7	54.1	83.66	75.11	88.62	81.90	65.51	58.73
RecLMIS† [11]	IEEE TMI'24	✓	23.7	24.1	85.22	77.00	85.78	78.76	63.66	55.96
SGSeg† [24]	MICCAI'24	✓	76.9	<u>19.3</u>	87.41	77.85	86.99	77.27	68.39	63.68
VLSM-Adapter† [7]	MICCAI'24	✓	136.9	<u>38.3</u>	79.98	76.69	82.34	74.91	65.02	57.20
L³Seg (Ours)†	ICCVW'25	✓	**8.2**	**5.1**	**90.98**	**83.46**	**90.10**	**82.67**	**85.53**	**74.72**

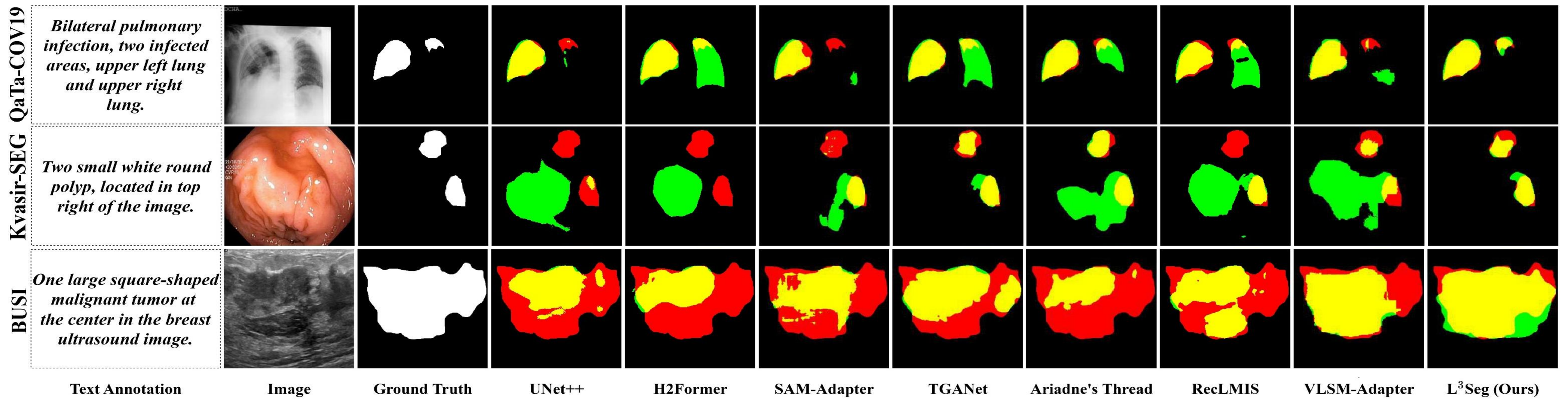

Fig. 2. Qualitative Comparison on QaTa-COV19, Kvasir-SEG and BUSI dataset. (TP, FN, FP)

Experimental Results

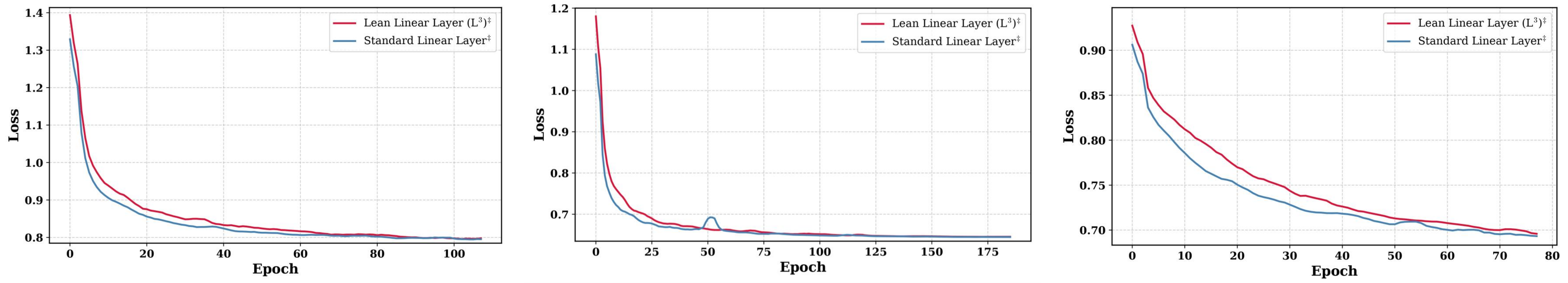

Fig. 3. Training loss curves for both the Standard Linear Layer and the Lean Linear Layer (L^3).

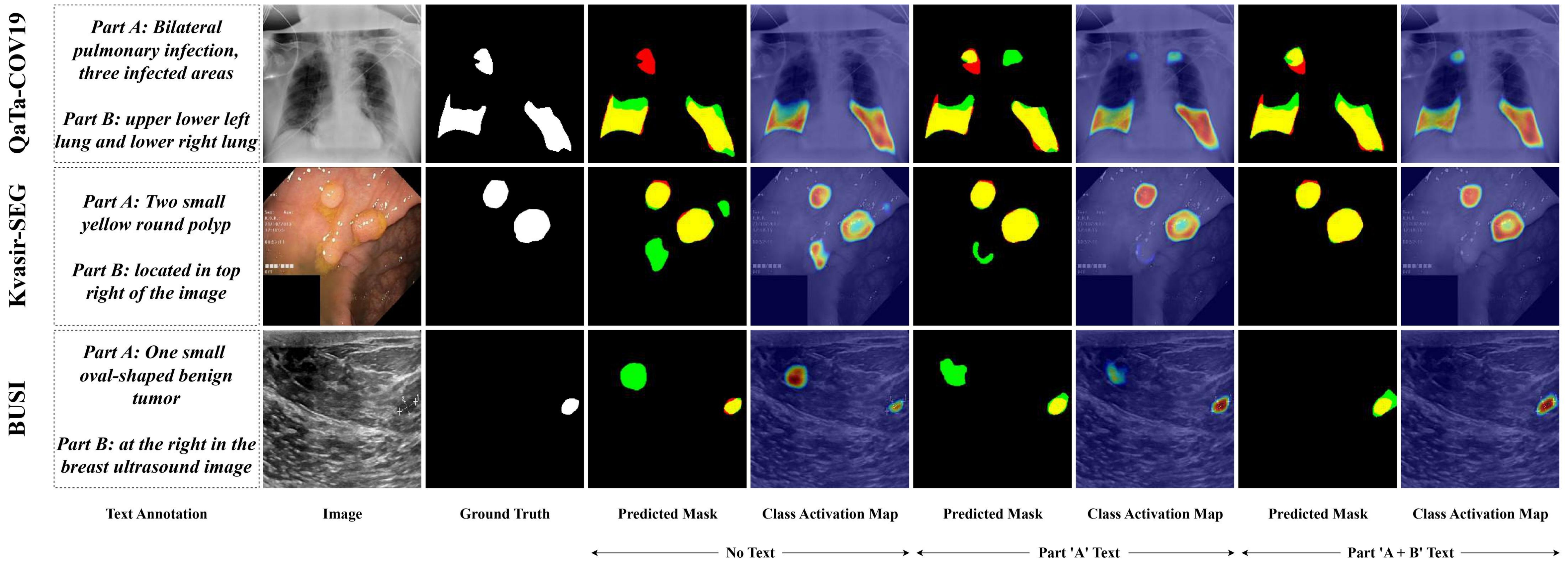

Fig. 4. Segmentation Visualizations with Varying Text Inputs. (TP, FN, FP)

Table 2. Impact of Training Data Size.

Data Usage	QaTa-COV19		Kvasir-SEG		BUSI	
	Dice(%) ↑	mIoU(%) ↑	Dice(%) ↑	mIoU(%) ↑	Dice(%) ↑	mIoU(%) ↑
SAM-Adapter [4] (100% Training)	84.76	73.55	83.42	71.55	77.47	63.22
VLSM-Adapter [7] (100% Training)	79.98	76.69	82.34	74.91	65.02	57.20
L³Seg (25% Training)	86.15	77.43	83.06	72.50	77.29	62.98
L³Seg (50% Training)	87.10	80.98	84.99	73.90	82.05	69.57
L³Seg (75% Training)	89.59	81.80	87.96	78.50	83.61	71.83
L³Seg (100% Training)	**90.98**	**83.46**	**90.10**	**82.67**	**85.53**	**74.72**

At a Glance

- Also accepted at **ICCV 2025 CVAMD Workshop.**
- For more information, please visit project webpage.