When CNN Features Help and When They Don't: Task-Aware Parameter-Efficient Adaptation of Frozen Medical Vision Transformers

Dina A. Elkholy[1], Mohamed S. Shehata[1], and John W. Braun[1]

Department of Computer Science
The University of British Columbia
Kelowna, BC, Canada
delkholy@student.ubc.ca

Abstract. Deploying deep learning for medical image analysis in resource-constrained settings motivates adaptation strategies that are accurate, parameter-efficient to train, and compact to store. We investigate how to inject convolutional inductive bias into a frozen, Low-Rank Adaptation (LoRA)-adapted MedMAE Vision Transformer (ViT) for both classification and segmentation. A controlled study reveals a task-dependent asymmetry: replacing the ViT's linear patch embedding with a trainable Convolutional Neural Network (CNN) stem improves classification on chest X-ray benchmarks but consistently degrades segmentation on a skin lesion benchmark. Through targeted ablations of the CNN-stem configuration, we find that none of the tested alignment, distribution-matching, or bypass strategies recovers the no-stem baseline. The pattern is consistent with a token distribution and spatial-calibration mismatch between CNN-generated tokens and a backbone pretrained on linearly projected patches, although a definitive causal claim would require further analysis. We further evaluate several architectural alternatives for segmentation, including a plain ViT + LoRA baseline, multi-scale ViT feature extraction, and a dual encoder that bypasses the frozen backbone. Under multi-seed validation, we find that they converge to comparable Dice scores, suggesting an empirical plateau for this backbone, dataset, and parameter-efficient fine-tuning (PEFT) setting rather than a universal ceiling. The resulting task-conditional recommendation is: for classification, replace the tokenisation; for segmentation, prefer plain or multi-scale ViT + LoRA rather than adding CNN complexity; for both, share a single frozen backbone.

Keywords: Parameter-efficient fine-tuning · Medical image analysis · Vision Transformer · Low-Rank Adaptation · Frozen backbones.

1 Introduction

Medical imaging models deployed in clinical settings face a persistent tension between predictive accuracy and computational efficiency. Large pretrained Vision Transformers (ViTs) [1] offer strong learned representations, but modern

ViT backbones often contain tens to hundreds of millions of parameters, with ViT-Base alone requiring approximately 86 M parameters. Fully fine-tuning a separate copy for each downstream task increases training cost, risks overfitting on small medical datasets, and creates task-specific checkpoints that become inefficient as the number of tasks grows. Parameter-efficient adaptation therefore offers a practical way to reuse a shared foundation model while keeping task-specific overhead small.

Low-Rank Adaptation (LoRA) [3] is one such approach: it freezes the pretrained backbone and trains only lightweight adapter matrices. However, LoRA does not change the ViT input representation. The model still relies on the standard linear patch embedding, which projects each 16×16 image patch independently. While this tokenisation is effective for classification, where information is aggregated into a global CLS representation, it may be less suitable for segmentation, where the decoder depends on spatially coherent patch-level features.

This raises a natural question: can convolutional inductive bias be added at the input while preserving the efficiency of a frozen LoRA-adapted backbone? We study this by replacing the linear patch embedding with a trainable CNN stem and evaluating the resulting model on both classification and segmentation tasks. The outcome is task-dependent: the CNN stem improves classification but underperforms for segmentation. This asymmetry motivates our central analysis: whether CNN-generated tokens are compatible with a frozen ViT backbone pretrained on linearly projected patches.

To answer this, we first probe the CNN-stem configuration through ablations targeting positional calibration, token alignment, optimisation, and decoder-level bypassing. We then compare alternative segmentation designs that avoid feeding CNN-generated tokens through the frozen transformer, including a plain ViT + LoRA baseline, a multi-scale ViT decoder that reuses intermediate transformer features, and a dual encoder that introduces CNN features only at the decoder. Together, these experiments characterise when convolutional features help, when they do not, and which parameter-efficient configuration is preferable for segmentation under the tested setting.

Contributions.

1. **Task-dependent effect of CNN tokenisation.** We show that replacing the linear patch embedding with a CNN stem has opposite effects across tasks: it improves classification by up to +0.07 AUC on NIH ChestX-ray14, but reduces segmentation performance by approximately −0.02 Dice on ISIC 2017. This establishes a task-dependent asymmetry in how frozen LoRA-adapted medical ViTs respond to convolutional input tokenisation.
2. **Ablation-based analysis of the segmentation drop.** Through thirteen ablations of the CNN-stem configuration, we test whether the no-stem baseline can be recovered through positional calibration, token alignment, optimisation changes, or decoder-level bypassing. The results are consistent with a structural mismatch between CNN-generated tokens and a frozen backbone

pretrained on linearly projected patches, although the mechanism remains interpretive rather than fully causal.

3. **Segmentation alternatives under a shared frozen backbone.** We compare the plain ViT + LoRA baseline with a dual encoder and a multi-scale ViT decoder under multi-seed validation. These configurations reach comparable Dice (≈ 0.83), suggesting an empirical plateau under the tested backbone, dataset, decoder, and PEFT setting rather than a universal ceiling.
4. **Parameter-efficient task-conditional recommendation.** Among the tested segmentation configurations, multi-scale ViT + LoRA matches the plain baseline with $\sim$3 M trainable parameters. The resulting recommendation is task-conditional: use CNN-stem tokenisation for classification, but prefer plain or multi-scale ViT + LoRA for segmentation rather than adding CNN complexity.

2 Related Work

2.1 Medical Imaging Foundation Models

Recent medical imaging foundation models have increasingly used large-scale pretraining to provide reusable backbones for downstream tasks. Domain-pretrained models such as MedMAE [2], Rad-DINO [4], and BiomedCLIP [5] have been explored for classification, segmentation, and multimodal medical-image analysis. At the same time, larger backbones such as CheXFound [6] and general-purpose self-supervised models such as DINOv2 show that scale can improve representation quality, but scale alone does not guarantee transfer to every medical task. In resource-constrained settings, efficiently adapting an existing pretrained backbone can therefore be more practical than deploying or fine-tuning a larger model for each downstream task.

2.2 Efficient Adaptation of Medical Foundation Models

LoRA [3] adapts a pretrained model by representing weight updates with low-rank matrices, $\mathbf{W} = \mathbf{W}_0 + \mathbf{BA}$, where $r \ll \min(d, k)$. This reduces the number of trainable parameters while preserving the original backbone. In medical imaging, LoRA-style adaptation has been used in segmentation frameworks such as SAMed [7], with more recent variants such as ARENA [8] adapting the rank to the downstream task. Visual prompt tuning [9] provides another lightweight alternative by learning task-specific prompts rather than updating the full backbone. These methods reduce adaptation cost, but they generally leave the input tokenisation unchanged. How tokenisation changes interact with parameter-efficient adaptation in frozen medical ViTs remains less explored.

2.3 Hybrid Tokenisation and CNN-Transformer Designs

Hybrid CNN-transformer architectures such as CvT [10], CoAtNet [11], and MedViT [12] integrate convolutions with attention to improve visual representation learning, while TransUNet [13] combines a CNN encoder with a ViT for segmentation. CNN-stem and progressive-tokenisation methods have also been studied for standard ViTs: Xiao et al. [14] showed that early convolutions improve ViT training stability on natural images, and Yuan et al. [15] proposed overlapping tokenisation through Tokens-to-Token ViT. These works show the promise of convolutional tokenisation in fully trained Vision Transformers. However, whether the same inductive bias remains useful when adapting frozen medical foundation models with PEFT remains unclear.

3 Method

3.1 Shared Backbone and Adaptation

We use MedMAE [2] (ViT-Base, 86 M parameters) as the shared medical vision backbone. LoRA adapters with rank $r = 16$, $\alpha = 16$, and dropout 0.1 are applied to the QKV projection in each of the twelve transformer blocks. In LoRA-based experiments, the original MedMAE backbone remains frozen and only the LoRA adapters and task-specific modules are trained. In full fine-tuning baselines, all backbone parameters are trainable.

All ViT-based configurations operate on 224×224 inputs and use a 16×16 patch grid, producing 196 patch tokens. For classification, the CLS representation is passed to a trainable linear head. For segmentation, the CLS token is discarded and the patch tokens are reshaped into a 14×14 spatial feature map before decoding.

3.2 Baseline ViT Configurations

The plain ViT classification and segmentation baselines use MedMAE's original linear patch embedding. For classification, the final CLS token is passed to a trainable linear classifier. For segmentation, the final patch tokens are reshaped into a $14 \times 14 \times 768$ feature map and passed to a trainable four-stage transposed-convolution decoder. The decoder upsamples the feature map to 224×224 using channel widths 256, 128, 64, and 32, followed by a final 1×1 convolution to produce the binary segmentation logit map.

These baselines are evaluated under both full fine-tuning and LoRA adaptation, allowing us to separate the effect of adaptation strategy from the effect of input tokenisation.

3.3 CNN-Stem Tokenisation

To test whether convolutional input tokenisation improves frozen medical ViTs, we replace the standard linear patch embedding with a trainable four-layer CNN

stem. Each layer applies a 3×3 convolution with stride 2, batch normalisation, and GELU activation, progressively downsampling the input from 224×224 to a $14 \times 14 \times 768$ feature map. As illustrated in Figure 1, this feature map is flattened into 196 tokens, combined with the CLS token and positional embeddings, and passed through the MedMAE backbone:

$$\text{Input} \xrightarrow[\text{trainable}]{\text{CNN Stem}} \text{Tokens} \xrightarrow[\text{frozen + LoRA}]{\text{ViT}} \text{Features} \xrightarrow[\text{trainable}]{\text{Head / Decoder}} \text{Prediction.} \quad (1)$$

The same CNN-stem tokenisation is evaluated for both classification and segmentation, under both full fine-tuning and LoRA. This gives a controlled comparison between two tokenisation choices (linear patch embedding vs. CNN stem) and two adaptation strategies (full fine-tuning vs. LoRA). The dual encoder and multi-scale ViT are evaluated separately as segmentation-only alternatives.

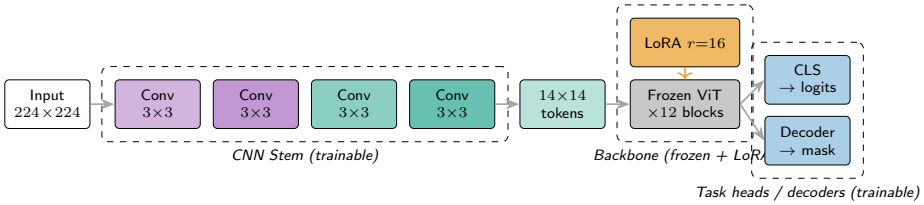

Fig. 1: CNN-stem tokenisation with a LoRA-adapted MedMAE backbone. The linear patch embedding is replaced by a trainable four-layer CNN stem; LoRA updates only QKV projections; task heads or decoders are trainable. Grey = frozen, coloured = trainable.

3.4 CNN-Stem Segmentation Ablations

To analyse the segmentation behaviour of the CNN-stem model, we perform thirteen ablations of the CNN Stem + LoRA configuration. These ablations test whether the no-stem ViT + LoRA baseline can be recovered by changing positional calibration, token alignment, optimisation, or decoder-level feature routing. Specifically, we evaluate trainable positional embeddings, stem warmup, a reduced stem learning rate, pretrained stem initialisation, token distillation to the original linear embedding, LayerNorm after the stem, skip connections to the decoder, and selected combinations of these strategies. All ablations keep the MedMAE backbone frozen except for LoRA adapters and modify only the CNN-stem segmentation configuration.

3.5 Multi-Scale ViT Decoder

We evaluate a CNN-free multi-scale ViT decoder that keeps the original linear patch embedding unchanged and reuses intermediate transformer features for segmentation. During the forward pass, patch-token features are collected from blocks 3, 6, 9, and 12. For each selected block, the CLS token is removed and the remaining tokens are reshaped into a 14×14 spatial feature map.

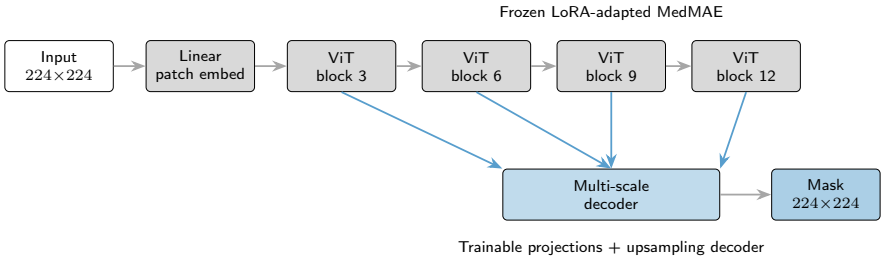

Fig. 2: Multi-scale ViT decoder. The original linear patch embedding is kept unchanged, and intermediate ViT block outputs are reused as decoder features. No CNN branch is introduced.

As shown in Fig. 2, these feature maps are projected with trainable 1×1 convolutions and fused in a lightweight U-Net-style decoder, which progressively upsamples the fused representation to a 224×224 segmentation mask. The trainable components are the LoRA adapters and the multi-scale decoder; no CNN branch is introduced.

3.6 Dual Encoder and Fusion Decoder

As a second segmentation alternative, we evaluate a dual-encoder model that keeps the ViT input unchanged and introduces CNN features only in the decoder:

$$\begin{aligned} &\text{Input} \xrightarrow[\text{frozen + LoRA}]{\text{ViT}} \text{global features} \searrow \\ &\text{Input} \xrightarrow[\text{trainable}]{\text{CNN branch}} \text{local features} \rightarrow \text{Fusion Decoder} \rightarrow \text{Mask}. \end{aligned} \tag{2}$$

The ViT branch uses the original linear patch embedding, avoiding any change to the token distribution entering the frozen transformer. In parallel, a lightweight CNN branch extracts features at 112×112, 56×56, 28×28, and 14×14 using four 3×3 convolution blocks with channel widths $[64, 128, 256, 512]$. Each block uses stride 2, batch normalisation, and GELU activation, and the branch is trained from scratch with Kaiming initialisation.

The fusion decoder combines ViT and CNN features in a U-Net-like cascade. At 14×14, the ViT feature map is projected from 768 to 256 channels and concatenated with the CNN branch's deepest feature map, also projected to 256 channels. The fused representation is progressively upsampled, with CNN features concatenated at 28×28, 56×56, and 112×112 before the final 224×224 prediction. The 1×1 projection layers make the decoder compatible with the CNN branch channel dimensions used in the reported experiments.

4 Experimental Setup

4.1 Datasets

NIH ChestX-ray14 *[16]* contains 112,120 frontal chest radiographs from 30,805 unique patients, labelled across fourteen thoracic pathologies including pneumo-

nia, cardiomegaly, atelectasis, effusion, and pneumothorax. We treat the task as multi-label classification and exclude the "No Finding" label. We use the official 85/15 train/validation split documented in the original release. This official partition is constructed at the patient level, ensuring that no patient contributes images to both the train and validation splits.

***CheXpert** [17]* contains chest radiographs labelled for multiple thoracic observations. We evaluate the five standard competition labels on AP frontal views, with uncertain labels mapped to zero. We rely on the official CheXpert training/validation partition, which is separated at the patient level so that no patient appears in both splits.

***ISIC 2017** [18]* provides 2,000 training, 150 validation, and 600 test dermoscopic images with pixel-level binary lesion annotations. We use it for binary skin-lesion segmentation.

All images are resized to 224×224 and normalised using ImageNet statistics. Classification augmentation includes random resized cropping and horizontal flipping. Segmentation augmentation includes random horizontal flips, random vertical flips, and rotations of $\pm 15°$, with identical geometric transforms applied to each image and mask using a shared random seed.

Relationship between pretraining and evaluation data. MedMAE [2] was pretrained on a large collection of public medical-imaging datasets covering CT, chest radiography, MR, and other related modalities. Since NIH ChestX-ray14 is part of this corpus, its evaluation results may benefit from pretraining overlap. In contrast, CheXpert and ISIC 2017 are not included in the pretraining data, making them a more direct assessment of transfer to unseen datasets.

4.2 Configurations

Table 1 summarises the experimental configurations. The main 2×2 comparison evaluates two tokenisation choices (linear patch embedding vs. CNN stem) under two adaptation strategies (full fine-tuning vs. LoRA) for both classification and segmentation. The dual encoder and multi-scale ViT are evaluated as segmentation-only alternatives to test whether local or intermediate features improve dense prediction without replacing the ViT input tokens.

4.3 Training Details

All experiments use AdamW ($\mathrm{lr} = 10^{-4}$, weight decay 0.05) on a single NVIDIA A6000 GPU. Classification models are trained for 50 epochs with batch size 32 using BCEWithLogitsLoss, with model selection based on best validation AUC. Segmentation models are trained for 50 epochs with batch size 16 using a combined Dice + BCE loss, with model selection based on best validation Dice. For segmentation, training images use random horizontal flips, random vertical flips, and random rotations up to 15°; validation and test images use deterministic resizing and normalisation.

Table 1: Experimental configurations. All ViT-based configurations use MedMAE (ViT-B). Task heads and decoders are fully trainable.

Configuration	Tokenisation	Adaptation	Task
ViT + Full FT	Linear	Full	Cls + Seg
CNN Stem + Full FT	CNN Stem	Full	Cls + Seg
ViT + LoRA	Linear	LoRA	Cls + Seg
CNN Stem + LoRA	CNN Stem	LoRA	Cls + Seg
Dual Encoder	Linear + CNN	LoRA	Seg
Multi-scale ViT	Linear	LoRA	Seg

Table 2: Classification results. Best result **bolded**. "Params" = trainable parameters.

		NIH ChestX-ray14		CheXpert	
Config.	**Params**	**Acc.**	**AUC**	**Acc.**	**AUC**
ViT + Full FT	86 M	87.79	0.66	75.26	0.82
CNN Stem + Full FT	87 M	88.25	0.73	79.45	0.87
ViT + LoRA	2.9 M	87.77	0.67	76.29	0.84
CNN Stem + LoRA	4.5 M	**88.33**	**0.74**	**80.17**	**0.88**

4.4 Metrics

Classification performance is reported using accuracy and macro-averaged AUC across the multi-label outputs. Accuracy is computed after applying a sigmoid activation and thresholding each output at 0.5. Segmentation performance is reported using Dice coefficient and Jaccard index (IoU), computed after thresholding sigmoid outputs at 0.5 and averaged over the evaluation set.

5 Results and Discussion

5.1 Classification: The CNN Stem Helps

Table 2 presents classification results. CNN Stem + LoRA achieves the highest accuracy and AUC among all ViT-based configurations on both datasets, despite training only 4.5 M parameters, roughly 19× fewer than fully fine-tuned ViT. The stem provides a consistent improvement regardless of adaptation strategy: +0.07 AUC on NIH and +0.05/ + 0.04 on CheXpert under full fine-tuning and LoRA, respectively. This pattern suggests that the improvement may reflect better input-token quality rather than a LoRA-specific interaction.

ViT + LoRA slightly outperforms ViT + Full FT on both datasets (0.67 vs. 0.66 AUC on NIH; 0.84 vs. 0.82 on CheXpert), suggesting that LoRA provides near-lossless adaptation in this setting with fewer trainable parameters. For classification, the results support replacing the patch embedding with a CNN stem, particularly when the goal is to preserve a shared frozen backbone and keep the task-specific trainable footprint small.

Table 3: ISIC 2017 segmentation. Best result **bolded**.

Config.	Params	Dice	Jaccard
ViT + Full FT	88 M	0.8140	0.6975
CNN Stem + Full FT	89 M	0.7954	0.6705
ViT + LoRA	3 M	**0.8243**	**0.7088**
CNN Stem + LoRA	4.5 M	0.8038	0.6830

A possible explanation is that classification relies on the CLS token as a global aggregator. The CLS representation may tolerate changes in the input token distribution while still benefiting from the CNN stem's richer local features. Under this interpretation, the CNN stem improves the information available to the backbone without disrupting the final global pooling mechanism.

5.2 Segmentation: The Same Stem Underperforms

Table 3 reveals the opposite pattern. The CNN stem degrades segmentation under both adaptation strategies: -0.02 Dice under LoRA (0.80 vs. 0.82) and -0.02 under full fine-tuning (0.80 vs. 0.81). This stands in contrast to classification, where the stem helps under both regimes.

An additional observation is that ViT + LoRA outperforms ViT + Full FT (0.8243 vs. 0.8140 Dice), despite training roughly 30× fewer parameters. On the small ISIC 2017 training set, freezing the backbone may reduce overfitting relative to full fine-tuning, while the LoRA adapters provide sufficient task-specific capacity. This pattern is consistent with LoRA acting as an implicit regulariser in this dense prediction setting.

The segmentation result also shows that the benefit of CNN tokenisation is task-dependent. Unlike classification, segmentation requires patch-level features to remain spatially coherent after the frozen transformer. CNN-stem tokens may depart from the linear-projection statistics that the pretrained backbone was calibrated for, and the decoder may not be able to recover the resulting loss of spatial calibration within the tested training regime. We treat this as an interpretive reading of the observed results, not as a fully isolated causal mechanism.

5.3 Ablation: Probing the Stem Underperformance

To probe why the CNN stem degrades segmentation, we conducted thirteen ablations of the CNN Stem + LoRA configuration, organised in three thematic rounds: alignment strategies, distribution matching, and bypass strategies. All ablations modify the naive-stem baseline; the ViT + LoRA configuration without any stem (Dice 0.8268) is shown as the target to match or exceed.

Table 4 and Figure 3(a) present results across the three rounds. The most effective single intervention is making the positional embeddings trainable (+0.016 Dice), which is consistent with spatial calibration contributing meaningfully to the gap: the frozen positional encodings are calibrated for linear-projected tokens, not CNN features. Combining trainable positional embeddings with stem

Table 4: Ablation experiments on ISIC 2017 segmentation. Thirteen ablations of the CNN Stem + LoRA configuration (rows below the naive-stem baseline). The ViT + LoRA result (no stem) is included as the target to match.

Approach	**Dice**	Δ
Round 1: Alignment strategies		
Naive stem (baseline)	0.7951	—
+ Trainable positional embeddings	0.8114	+0.016
+ Stem warmup (10 epochs)	0.7873	−0.008
+ Differential LR (stem 10^{-5})	0.8024	+0.007
+ Pretrained stem init	0.7882	−0.007
Round 2: Distribution matching		
+ Token distillation ($\lambda = 0.5$)	0.8014	+0.006
+ Token distillation ($\lambda = 0.1$)	0.7987	+0.004
+ LayerNorm after stem	0.8020	+0.007
+ Pos. embed. + warmup	0.8148	+0.020
+ LayerNorm + pos. embed.	0.8050	+0.010
+ Pos. embed. + diff. LR	0.7986	+0.004
Round 3: Bypass strategies		
+ Skip connections to decoder	0.7969	+0.002
+ Skip + pos. embed.	0.8052	+0.010
+ Skip + pos. embed. + warmup	0.8043	+0.009
ViT + LoRA (no stem)	**0.8268**	—

warmup yields the best stem result (0.8148, +0.020), but still falls 0.012 Dice short of simply removing the stem.

Token distillation, which explicitly encourages the stem to produce backbone-compatible tokens via an MSE loss against the original linear embedding, provides modest improvement (+0.006 at λ=0.5). One interpretation is that pushing the stem toward the linear projection sacrifices the local inductive bias that motivated using a CNN in the first place. Skip connections, which route stem features directly to the decoder, also do not recover the baseline. Under our setting, the backbone still processes the disrupted tokens through all twelve transformer blocks, and the decoder cannot compensate.

Takeaway. Under the tested settings, none of the thirteen ablations matched the no-stem baseline. The gap is consistent with a structural mismatch between CNN-generated tokens and a backbone pretrained on linear projections, rather than a simple hyperparameter issue. However, we cannot exclude that an alignment strategy outside our search space could close the gap.

5.4 Beyond the Stem: Multi-Seed Architectural Comparison

The ablation results suggest that the stem is poorly matched to a frozen ViT for segmentation under our setting. We therefore evaluate three architectural alter-

Table 5: Multi-seed segmentation results on ISIC 2017 (seeds 42, 123, 456). Mean ± std reported. Best mean **bolded**.

Config.	Trainable params	Dice	Jaccard
ViT + LoRA (baseline)	∼3.0 M	**0.831** ± 0.007	0.715 ± 0.009
Multi-scale ViT	∼2.5 M	**0.831** ± 0.005	**0.716** ± 0.007
Dual (lightweight CNN)	∼5.0 M	0.828 ± 0.008	0.711 ± 0.011
Dual + trainable pos. embed.	∼5.2 M	0.827 ± 0.011	0.710 ± 0.014

natives that route or extract local features differently, with multi-seed validation (seeds 42, 123, 456):

- **Dual encoder**: parallel CNN branch fused only in the decoder, bypassing the frozen backbone entirely.
- **Multi-scale ViT**: extract intermediate features from blocks 3, 6, 9, 12 of the ViT itself for U-Net-style decoding (no CNN at all).
- **ViT + LoRA (baseline)**: standard single-scale decoder for direct comparison.

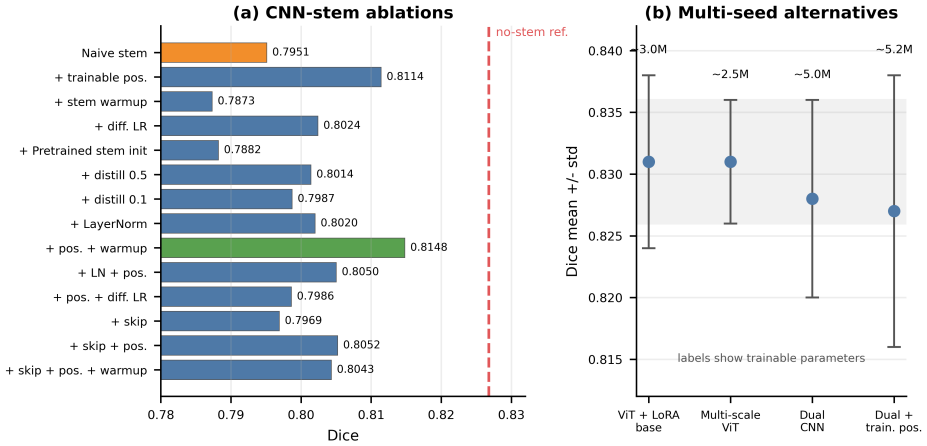

Fig. 3: Segmentation summary. **(a)** All thirteen CNN-stem ablations remain below the no-stem ViT + LoRA reference. **(b)** Plain ViT + LoRA, multi-scale ViT, and dual-encoder variants achieve comparable Dice within seed-to-seed variance.

Table 5 and Figure 3(b) show that all four configurations land within 0.005 Dice of each other, well within seed-to-seed variance. Multi-scale ViT achieves the same mean as the baseline (0.831) with the lowest variance (±0.005) and the fewest trainable parameters (∼2.5 M). The dual encoder, despite adding a complete parallel CNN branch, does not exceed the baseline; its single-seed best (0.8398, seed 42) does not generalise across seeds.

This convergence suggests an *empirical plateau* specific to the evaluated MedMAE backbone, ISIC 2017 dataset, decoders, and PEFT configurations. However, we cannot claim a universal ceiling: a different backbone, a richer pretrain-

ing regime, a different downstream dataset, or a more expressive PEFT method might reach higher performance.

The frozen MedMAE features may limit segmentation performance under our PEFT setting. ViT + LoRA already appears to extract most of the task-relevant information available in the frozen features. Adding a CNN branch does not improve performance, possibly because the CNN features are introduced only at the decoder and do not alter the representation learned by the frozen backbone. Multi-scale extraction from intermediate ViT blocks reaches the same narrow performance range, which is consistent with the final LoRA-adapted representation already capturing much of the useful depth-wise variation.

Among the tested alternatives, multi-scale ViT is the most parameter-efficient segmentation configuration. At $\sim$2.5 M trainable parameters, 17% fewer than the plain baseline and about 50% fewer than the dual encoder, multi-scale ViT matches the baseline mean with lower seed-to-seed variance. For settings where trainable-parameter and storage budgets dominate, this is our recommended segmentation configuration.

The CNN stem is the only tested segmentation option consistently below the no-stem alternatives. The CNN-stem variants fall below the corresponding no-stem baselines in Tables 3 and 4, which is consistent with the stem disrupting the frozen backbone's spatial encoding rather than supplying useful complementary signal in this setting.

5.5 Qualitative Analysis

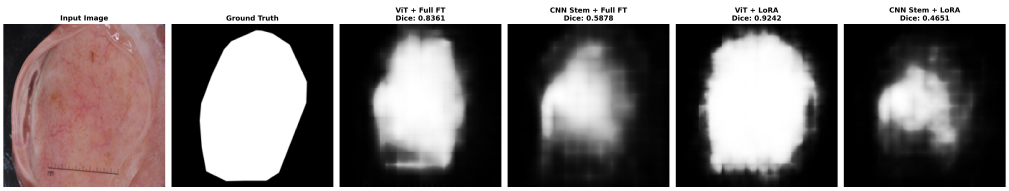

Fig. 4: Qualitative comparison on one ISIC 2017 test case. The example illustrates that CNN-stem variants produce less coherent masks than the no-stem ViT + LoRA baseline.

The qualitative example in Fig. 4 supports the quantitative trend observed in Tables 3 and 4: the CNN-stem variants produce less spatially coherent masks than the no-stem ViT + LoRA baseline. This test case is intended as an illustrative example of the observed behaviour.

5.6 Broader Interpretation

Taken together, the results point to a task-dependent asymmetry in how frozen LoRA-adapted medical ViTs respond to convolutional input tokenisation. Classification can benefit from the CNN stem, likely because the final decision depends

on a global CLS representation. Segmentation, however, depends on spatially coherent patch tokens, and the CNN-generated tokens appear less compatible with a backbone pretrained on linear patch projections.

The empirical plateau observed among ViT + LoRA, multi-scale ViT + LoRA, and the dual encoder should be interpreted narrowly. It is specific to the MedMAE backbone, ISIC 2017 dataset, decoder designs, and LoRA configuration (r=16, QKV adapters, rank fixed throughout training). A different backbone, pretraining regime, PEFT method, or decoder could reach higher performance. Within the tested setting, however, additional CNN complexity did not improve segmentation accuracy and increased the trainable parameter count.

Overall, the combined evidence supports a simple practical recommendation: use CNN-stem tokenisation for classification, prefer plain or multi-scale ViT + LoRA for segmentation, and share a single frozen backbone across tasks.

6 Conclusion and Future Work

We investigated how convolutional inductive bias interacts with parameter-efficient adaptation in a frozen medical Vision Transformer, revealing a clear task-dependent asymmetry. For classification, replacing the linear patch embedding with a trainable CNN stem consistently improved performance, with CNN Stem + LoRA achieving the best results on both NIH ChestX-ray14 and CheXpert. This suggests that CNN-based tokenisation can enrich local visual features when the final prediction relies on a globally pooled CLS representation. In contrast, for segmentation, the same modification consistently reduced performance under both full fine-tuning and LoRA. Across thirteen ablations, no CNN-stem variant recovered the no-stem ViT + LoRA baseline, indicating that CNN-generated tokens are less compatible with a frozen backbone pretrained on linear patch projections, particularly when dense prediction requires spatially coherent patch-level features.

We further evaluated segmentation alternatives that preserve the original ViT tokenisation, including the plain ViT + LoRA baseline, a dual encoder, and a multi-scale ViT decoder. Under multi-seed validation, these approaches converged to a narrow performance range around Dice ≈ 0.83, suggesting an empirical plateau for the evaluated MedMAE backbone, ISIC 2017 dataset, decoder designs, and LoRA configuration. Among the tested alternatives, multi-scale ViT + LoRA matched the plain baseline while using fewer trainable parameters and showing lower seed-to-seed variance, making it the most parameter-efficient segmentation choice in this setting. Together, these results support a task-conditional design recommendation: use CNN-stem tokenisation for classification, but prefer plain or multi-scale ViT + LoRA for segmentation while maintaining a shared frozen backbone across tasks.

This study is intentionally designed as a controlled architectural analysis rather than a broad benchmarking study. By isolating the evaluated MedMAE backbone, specific datasets, decoder designs, and LoRA configuration, we reduce confounding factors and make the observed performance variations eas-

ier to interpret in relation to the tested architectural modifications. Consequently, our findings map the structural behaviour of this specific adaptation regime rather than establishing a universal ceiling on segmentation performance. Building on this controlled foundation, extending the framework across broader medical modalities, segmentation datasets, foundation models, decoder designs, and PEFT methods represents a natural progression. Furthermore, future work should move from ablation-based empirical observation toward deeper representational analysis by directly measuring token statistics, positional calibration, attention-map behaviour, and feature-space alignment. Such analysis would provide stronger mechanistic evidence for why convolutional tokenisation benefits global classification but can disrupt the spatial calibration required for dense prediction tasks.

References

1. Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: Proc. ICLR (2021)
2. Gupta, A., Osman, I., Shehata, M.S., Braun, J.W.: MedMAE: A Self-Supervised Backbone for Medical Imaging Tasks. arXiv preprint arXiv:2407.14784 (2024)
3. Hu, E.J., Shen, Y., Wallis, P., et al.: LoRA: Low-rank adaptation of large language models. In: Proc. ICLR (2022)
4. Pérez-García, F., et al.: Rad-DINO: Exploring scalable medical image encoders beyond text supervision. arXiv preprint arXiv:2401.10815 (2024)
5. Zhang, S., et al.: BiomedCLIP: A multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. arXiv preprint arXiv:2303.00915 (2023)
6. Yang, Y., et al.: CheXFound: A vision foundation model for chest radiographs. arXiv preprint arXiv:2402.01034 (2024)
7. Zhang, K., Liu, D.: Customized Segment Anything Model for medical image segmentation. arXiv preprint arXiv:2304.13785 (2023)
8. Baklouti, G., Silva-Rodríguez, J., Dolz, J., Bahig, H., Ben Ayed, I.: Regularized Low-Rank Adaptation for Few-Shot Organ Segmentation. In: Proc. MICCAI, LNCS, vol. 15966, pp. 517–527. Springer (2025)
9. Jia, M., Tang, L., Chen, B., et al.: Visual prompt tuning. In: Proc. ECCV, pp. 709–727 (2022)
10. Wu, H., Xiao, B., Codella, N., et al.: CvT: Introducing convolutions to Vision Transformers. In: Proc. ICCV, pp. 22–31 (2021)
11. Dai, Z., Liu, H., Le, Q.V., Tan, M.: CoAtNet: Marrying convolution and attention for all data sizes. In: Proc. NeurIPS, pp. 3965–3977 (2021)
12. Manzari, O.N., et al.: MedViT: A robust vision transformer for generalized medical image classification. Comput. Biol. Med. **157**, 106791 (2023)
13. Chen, J., Lu, Y., Yu, Q., et al.: TransUNet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 (2021)
14. Xiao, T., Singh, M., Mintun, E., et al.: Early convolutions help transformers see better. In: Proc. NeurIPS, pp. 30392–30400 (2021)
15. Yuan, L., Chen, Y., Wang, T., et al.: Tokens-to-Token ViT: Training vision transformers from scratch on ImageNet. In: Proc. ICCV, pp. 558–567 (2021)
16. Wang, X., Peng, Y., Lu, L., et al.: ChestX-ray8: Hospital-scale chest X-ray database and benchmarks. In: Proc. CVPR, pp. 2097–2106 (2017)

17. Irvin, J., Rajpurkar, P., Ko, M., et al.: CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Proc. AAAI, pp. 590–597 (2019)
18. Codella, N., Gutman, D., Celebi, M.E., et al.: Skin lesion analysis toward melanoma detection. In: Proc. ISBI, pp. 168–172 (2018)
19. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Proc. MICCAI, pp. 234–241 (2015)
20. Yuan, C., Zhao, D., Agaian, S.S.: MUCM-Net: A Mamba powered UCM-Net for skin lesion segmentation. Exploration of Medicine **5**, 694–708 (2024)