

SAM+D: Parameter-Efficient Dimensional Lifting of SAM-Family Models via Depth-Routed LoRA and Depth Shifting

Yu Song^{1,*}, Hao Sun^{1,*}, Shiyu Teng¹, Ikuko Nishikawa¹, and Yen-wei Chen^{1,†}

College of Information Science and Engineering, Ritsumeikan University
2-150 Iwakura-cho, Ibaraki, Osaka 567-8570, Japan
yusong@fc.ritsumei.ac.jp, chen@is.ritsumei.ac.jp

* Equal contribution. † Corresponding author.

Abstract. Existing methods for adapting 2D foundation models such as SAM to 3D volumes either process slices independently—ignoring inter-slice context—or require substantial architectural changes and retraining. In this paper, we present **SAM+D**, a parameter-efficient framework that lifts SAM-family models by one spatial dimension—enabling 3D volumetric segmentation from 2D SAM and, for the first time via parameter-efficient fine-tuning, end-to-end 4D (3D+T) spatiotemporal segmentation from video-based SAM2—while keeping the vast majority of pre-trained parameters frozen. SAM+D introduces two lightweight, model-agnostic modules into frozen transformer blocks: (1) **Depth-Routed LoRA (DRLoRA)** experts with learned routing for spatially adaptive low-rank updates, and (2) **Depth Shift Modules (DSM)** for cross-slice feature exchange at zero additional parameter cost. Together, they provide volume-level context while tuning only $\sim 2.8\%$ of parameters for SAM and $\sim 3.7\%$ for SAM2. We evaluate SAM+D in two distinct settings, each lifting the base model by one spatial dimension: 3D segmentation, where SAM (2D $\rightarrow$ 3D) is evaluated on four CT benchmarks (KiTS, Pancreas, LiTS, Colon), and 4D segmentation, where SAM2 (2D+T $\rightarrow$ 3D+T) is evaluated on a cell tracking challenge (CTC) dataset (Fluo-N3DH-SIM+). In both settings SAM+D achieves competitive or superior results under the single-point prompt setting while using fewer trainable parameters than existing methods, demonstrating that SAM+D generalizes across SAM-family architectures, target dimensionalities (3D, 4D), and domains spanning medical imaging and bio-scene understanding. Code is publicly available at <https://github.com/JerrySongCST/SAM-Plus-D>.

Keywords: Segment Anything Model · Parameter-efficient adaptation · Dimensional lifting · 3D/4D segmentation

1 Introduction

In recent years, a large number of vision foundation models have focused on 2D images, from self-supervised feature extractors such as VAEs [23] and DINO [33],

to diffusion-based image generators like stable diffusion [35], to segmentation foundation models such as the Segment Anything Model (SAM) family [6, 24, 34]. However, in a wide range of real-world applications—from CT tumor segmentation for medical diagnosis to cell tracking on time-lapse microscopic images—current 2D models substantially lack the depth context that distinguishes overlapping structures. Take SAM [24] as an example: it is trained on over one billion masks on 2D images, while SAM2 [34] extends it to 2D video via tracklet-based memory propagation, but neither provides volumetric understanding. This gap has fuelled urgent interest in spatial intelligence—equipping foundation models with genuine volumetric understanding in 3D and 4D (3D+T).

Volumetric segmentation has traditionally relied on fully 3D architectures trained from scratch [10, 20], which model inter-slice context natively but require large annotated 3D datasets that are hard to obtain in many domains, especially in medical and biomedical domains, where dataset sizes are relatively small. Recent 3D segmentation foundation models [13, 32, 42] scale this paradigm by rebuilding or fully fine-tuning SAM/SAM2 with 3D components on massive datasets, where 2D priors are discarded in the process. This motivates a second line of work that directly adapts pre-trained 2D model weights for volumetric input, ranging from injecting 3D adapters or attention layers into the frozen backbone [7, 45, 55], to reinterpreting volumes as spatial videos for SAM2 [48, 54], to inserting minimal spatial adapters with very few trainable parameters [15].

Despite the recent progress, existing methods share several limitations. First, existing adapters like Low-Rank Adaptation (LoRA) [18] apply the same transformation everywhere in the volume, ignoring that features such as sharp boundaries and homogeneous interiors need different adaptations—position-aware routing is missing. Second, modelling cross-slice dependencies still requires 3D convolutions or attention that add extra parameters and memory; no method achieves inter-slice communication at zero cost. Third, each method targets a single base model—either SAM or SAM2—so every new foundation model needs a new adapter design; no unified architecture lifts both SAM to 3D and SAM2 to 4D with the same design.

To address the limitations mentioned above, we introduce "SAM+D" in this paper, which introduces two low-cost, parameter-efficient modules into each frozen pre-trained transformer encoder block to lift SAM-family models by one spatial dimension. The first, Depth-Routed LoRA (DRLoRA), maintains multiple LoRA experts per block and routes among them based on the slice's depth position, so that boundary regions and organ interiors can receive distinct low-rank updates. The second, Depth Shift Module (DSM) repurposes the Temporal Shift Module (TSM) [29], originally designed for video CNNs, to shift a fraction of channels between neighbouring slices before self-attention—giving each token access to adjacent-slice features at zero extra parameters or FLOPs. By resampling the entire volume to a fixed depth and processing all slices jointly in a single forward pass, SAM+D operates as a patch-free method [21, 43], preserving global spatial context without sliding window inference. Neither module assumes anything about the base architecture beyond standard transformer blocks; the

same pair therefore lifts SAM ($2D \rightarrow 3D$) and SAM2 ($2D+T \rightarrow 3D+T$) without redesign.

Our contributions can be summarized as follows:

- We propose SAM+D, a unified parameter-efficient framework that lifts SAM-family models by one spatial dimension—enabling 3D volumetric segmentation from SAM and 3D+T spatiotemporal segmentation from SAM2—using the same lightweight components while keeping the vast majority of pre-trained parameters frozen.
- We introduce Depth-Routed LoRA (DRLoRA), which learns to blend multiple low-rank experts based on spatial context, and Depth Shift Modules (DSM) that shift a fraction of feature channels between adjacent slices before self-attention, accumulating cross-slice interaction across layers at zero cost (zero added parameters, zero multiply-accumulate (MAC) operations).
- We evaluate SAM+D on both SAM and SAM2, covering 3D and 3D+T settings across medical imaging and cell tracking tasks, achieving competitive or superior results on five benchmarks—four CT organ segmentation tasks (KiTS [16], Pancreas [1], LiTS [3], Colon [1]) and one Cell Tracking Challenge (CTC) data (Fluo-N3DH-SIM+) [41]—while training only $\sim 2.8\%$ and $\sim 3.7\%$ of total parameters.

2 Related Work

2.1 SAM-Family Models

The Segment Anything Model (SAM) [24] introduced a promptable segmentation paradigm trained on over 11 million images and one billion masks. Its ViT-based image encoder [12], combined with a lightweight prompt encoder and mask decoder, delivers strong zero-shot segmentation across diverse 2D domains. SAM2 [34] extends this paradigm to video by replacing the ViT encoder with a hierarchical Hiera backbone [36] and introducing a streaming memory architecture—comprising a memory encoder, memory bank, and memory attention module—that propagates object identity across frames via tracklet-based matching. More recently, SAM3 [6] further extends the SAM family with concept-based segmentation, introducing a text encoder and visual detector that enable segmentation of objects specified by text queries or visual exemplars within video tracklets.

Despite their architectural differences, all SAM-family models share a fundamental limitation: they operate natively in two spatial dimensions ($2D$ for SAM, $2D+T$ for SAM2). Volumetric data— $3D$ medical scans, $3D+T$ time-lapse microscopy—cannot be processed without either discarding inter-slice context (slice-by-slice application) or modifying the architecture. SAM+D addresses this limitation by lifting any SAM-family models by one spatial dimension through lightweight, model-agnostic insertions.

2.2 Parameter-Efficient Fine-Tuning

Parameter-efficient fine-tuning (PEFT) adapts large pretrained models by updating only a small fraction of parameters. Early strategies include prompt tuning [26] and prefix tuning [28], which prepend learnable tokens to the input sequence, and adapter modules [8, 17], which insert lightweight bottleneck layers into transformer blocks. Visual Prompt Tuning (VPT) [22] extends prompt tuning to vision transformers by injecting learnable tokens into each layer’s input space. Low-Rank Adaptation (LoRA) [18] has since become the dominant paradigm, injecting trainable low-rank matrices into frozen linear projections with minimal inference overhead, and has been widely adopted for both language and vision models [19, 50]. Subsequent works refine LoRA along several axes: QLoRA [11] combines 4-bit quantisation with LoRA for memory-efficient fine-tuning, DoRA [30] decomposes weight updates into magnitude and direction components, and AdaLoRA [51] adaptively allocates rank budgets across layers based on importance scores. Several works further extend LoRA with mixture-of-experts (MoE) routing—MoLoRA [49], MoLE [46], and MixLoRA [27]—training multiple LoRA experts with learned gating to improve multi-task generalisation.

However, all existing MoE-LoRA methods route on task identity or token content and require auxiliary load-balancing or importance losses to avoid expert collapse. DRLoRA is categorically different: it routes on the normalised depth position z , which makes balancing losses unnecessary and shrinks the router to $\sim 0.14\text{K}$ parameters (vs. $\sim 3\text{--}6\text{K}$ for content routers).

2.3 Adapting Foundation Models for Volumetric Segmentation

Existing approaches to volumetric segmentation span a wide spectrum. At one end, fully 3D architectures such as nnU-Net [20] and dedicated 3D foundation models (SAM-Med3D [42], SegVol [13], MedSAM2 [32]) model inter-slice context natively but forfeit 2D pretrained priors or require massive volumetric corpora. At the other end, adapter-based methods retain a frozen 2D backbone: 3DSAM-adapter [15] inserts depthwise 3D convolutions, MA-SAM [7] uses factorised 3D adapters, and Med-SA [45] bifurcates attention into spatial and depth branches (SD-Trans)—the closest prior work to our proposed DSM, though SD-Trans doubles the attention computation whereas DSM incurs zero additional cost. More recently, SAM2-based methods [48, 54] treat 3D volumes as spatial videos, but remain SAM2-specific and address 3D segmentation only.

All existing approaches target a single base model and a single dimensionality; none provides a unified mechanism that lifts both SAM ($2\text{D} \rightarrow 3\text{D}$) and SAM2 ($2\text{D}+\text{T} \rightarrow 3\text{D}+\text{T}$) with same design.

3 Proposed Method

3.1 Overview

SAM+D lifts frozen SAM-family models by one spatial dimension through two lightweight modules inserted into every transformer block, as shown in Fig. 1.

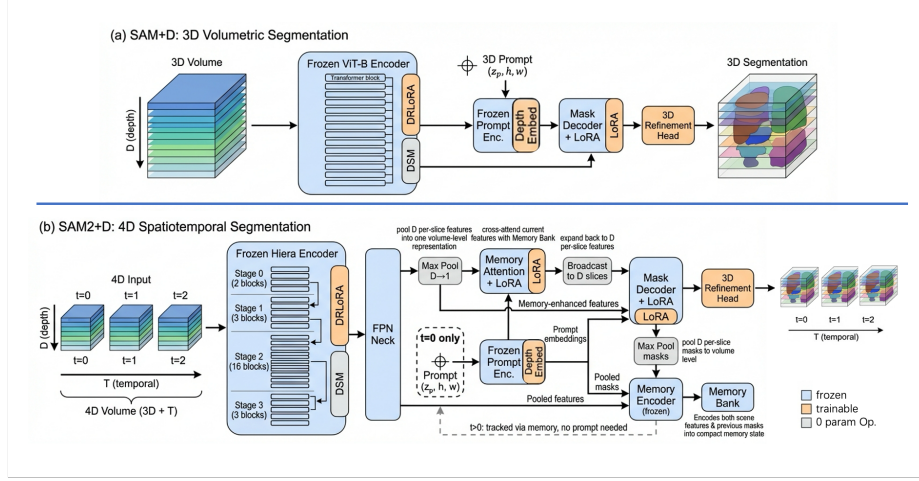


Fig. 1: Overview of SAM+D. (a) 3D setting: DRLoRA and DSM are inserted into SAM’s frozen ViT-B encoder; a LoRA-adapted decoder produces masks refined by a 3D head. (b) 4D setting: the same modules are inserted into SAM2’s frozen Hiera encoder; depth-pooled features interface with LoRA-adapted memory attention for prompt-free tracking at $t > 0$. Blue: frozen; orange: trainable; gray: zero-parameter operation.

Depth-Routed LoRA Experts (DRLoRA) replaces the standard single-LoRA adaptation with multiple low-rank experts whose contributions are blended by a learned router conditioned on depth position, capturing the structural variation that arises along the depth axis of volumetric data (§3.2). Depth Shift Modules (DSM) exchange a fraction of feature channels between adjacent slices before each attention computation, providing cross-slice context at zero additional parameters or FLOPs (§3.3).

We instantiate SAM+D in two settings. In the 3D setting (§3.4), DRLoRA and DSM are inserted into SAM’s frozen ViT-B encoder, lifting 2D image segmentation to 3D volumetric segmentation of CT scans. In the 4D setting (§3.5), the same modules are inserted into SAM2’s hierarchical Hiera encoder, lifting 2D+T video segmentation to 3D+T spatiotemporal segmentation, with SAM2’s native memory mechanism handling temporal propagation. In both cases, only around $\sim 2.8\%$ and $\sim 3.7\%$ of total parameters are trainable; all pretrained encoder weights remain frozen.

3.2 Depth-Routed LoRA Experts (DRLoRA)

LoRA [18] injects trainable low-rank matrices into frozen layers, providing an efficient alternative to full fine-tuning. However, a single LoRA applies the same update to every slice, ignoring the fact that adaptation needs vary systematically along the depth axis—e.g., anatomy near the liver dome differs markedly from pelvic slices. DRLoRA addresses this with a mixture-of-experts formulation that conditions the adaptation on depth position, as shown in Fig. 2.

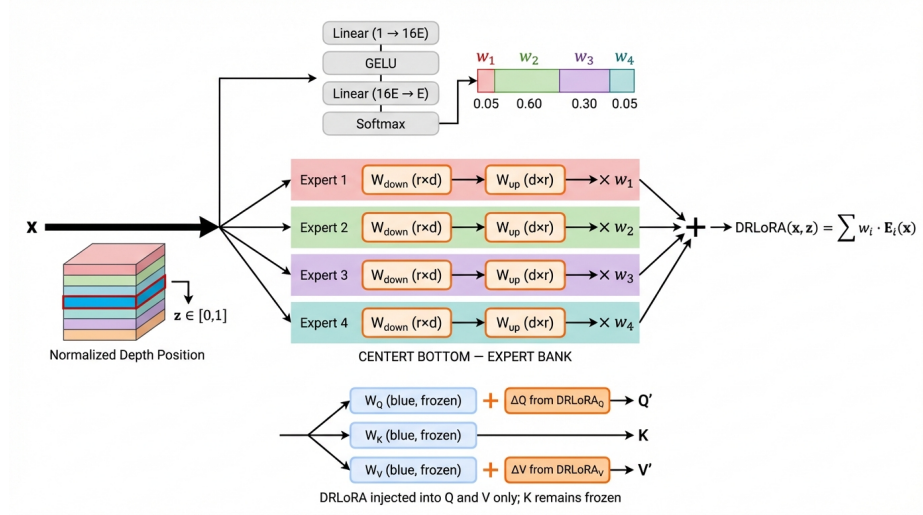


Fig. 2: Depth-Routed LoRA (DRLoRA): a lightweight router maps the normalised depth position z to a softmax distribution over E LoRA experts. The weighted sum of expert outputs is added to the frozen Q and V projections, producing depth-dependent adaptations while leaving K unchanged.

Concretely, DRLoRA maintains E parallel LoRA experts $\{\mathcal{E}_1, \dots, \mathcal{E}_E\}$, each a standard low-rank factorisation:

$$\mathcal{E}_i(\mathbf{x}) = \mathbf{W}_{\text{up}}^{(i)} \mathbf{W}_{\text{down}}^{(i)} \mathbf{x}, \quad \mathbf{W}_{\text{down}}^{(i)} \in \mathbb{R}^{r \times d}, \quad \mathbf{W}_{\text{up}}^{(i)} \in \mathbb{R}^{d \times r}, \quad (1)$$

where d is the embedding dimension and $r \ll d$ is the rank. $\mathbf{W}_{\text{up}}^{(i)}$ is initialised to zero so that training begins from the identity. To select among experts, a lightweight MLP routes based on the normalised depth position $z = d/D_s \in [0, 1]$:

$$\mathbf{w} = \text{softmax}(\mathcal{R}(z)), \quad \mathcal{R}: \mathbb{R}^1 \xrightarrow{\text{Lin}} \mathbb{R}^h \xrightarrow{\text{GELU}} \mathbb{R}^h \xrightarrow{\text{Lin}} \mathbb{R}^E, \quad (2)$$

where h is the hidden dimension. The final output is the soft-weighted combination of all experts:

$$\text{DRLoRA}(\mathbf{x}, z) = \sum_{i=1}^E w_i \cdot \mathcal{E}_i(\mathbf{x}). \quad (3)$$

We inject DRLoRA into the *query* and *value* projections of each self-attention layer, leaving keys frozen:

$$\mathbf{Q}' = \mathbf{W}_Q \mathbf{x} + \text{DRLoRA}_Q(\mathbf{x}, z), \quad \mathbf{V}' = \mathbf{W}_V \mathbf{x} + \text{DRLoRA}_V(\mathbf{x}, z), \quad \mathbf{K} = \mathbf{W}_K \mathbf{x}. \quad (4)$$

Following the finding of Hu et al. [18] that adapting both W_Q and W_V outperforms other combinations—including higher-rank adaptation of a single matrix—under a fixed parameter budget, we apply DRLoRA to these two projections.

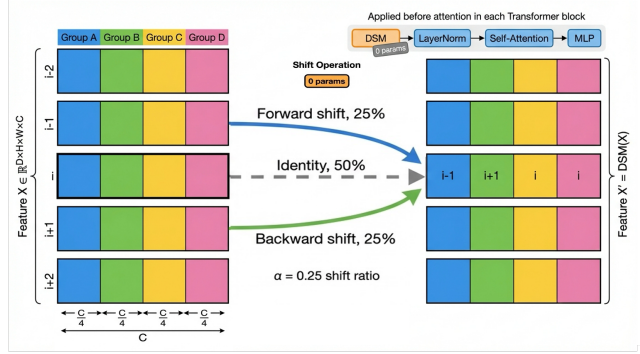


Fig. 3: Depth Shift Module (DSM) applied along the depth axis. Before self-attention, a fraction α of channels are shifted forward from slice $i-1$ and backward from slice $i+1$, while the remaining $(1-2\alpha)$ channels stay in place. This gives each slice access to neighbouring features at zero extra parameters or FLOPs.

Intuitively, adapting queries modifies the attention pattern—which tokens each position attends to—while adapting values modifies the information content that is propagated through attention. Together they give DRLoRA control over both where and what the model attends to in a depth-dependent manner, while the frozen keys preserve the pre-trained feature space as a stable matching reference.

3.3 Depth Shift Module (DSM)

Processing a volume slice by slice through a 2D encoder isolates each slice from its neighbours, preventing the model from capturing inter-slice continuity. Adding 3D convolutions or cross-slice attention would introduce substantial parameters and compute. We instead introduce the Depth Shift Module (DSM) [29]—inspired by TSM originally proposed for efficient video understanding—repurposed here for cross-slice feature exchange along the depth axis, as shown in Fig. 3. DSM adds no learnable parameters and performs no multiply-accumulate (MAC) operations; its only overhead is the memory traffic of a slice-strided copy, which we measure at 0.34 ms on an RTX5090 GPU (1.7% of one encoder block’s forward latency). Our contribution is not the operator itself but the lifting strategy: we are the first to show that a temporal-shift operator, applied along the spatial depth axis, both outperforms no-exchange and parameter-rich cross-slice alternatives (supplementary material, Table 8) and transfers unchanged from 2D→3D (SAM) to 2D+T→3D+T (SAM2).

Given a feature tensor $\mathbf{X} \in \mathbb{R}^{D \times H \times W \times C}$, DSM partitions the channels into three groups controlled by a shift ratio α :

$$\mathbf{X}' = \text{DSM}(\mathbf{X}) = \text{concat}[\mathbf{X}^{\text{fwd}}, \mathbf{X}^{\text{bwd}}, \mathbf{X}^{\text{id}}], \quad (5)$$

where $\mathbf{X}^{\text{fwd}}[i]$ copies αC channels from slice $i-1$ (forward shift), $\mathbf{X}^{\text{bwd}}[i]$ copies αC channels from slice $i+1$ (backward shift), and the remaining $(1-2\alpha)C$ chan-

nels are unchanged. Boundary slices retain their own features to avoid artificial discontinuities.

DSM is applied before self-attention in each transformer block, so that query, key, and value projections already incorporate neighbouring-slice information. A key design choice is that DSM operates exclusively along the depth dimension—across spatial slices within a single volume—rather than across temporal frames. In the 4D setting, this ensures a clean separation of concerns: depth-local context is handled by DSM, while temporal dynamics are delegated to SAM2’s memory mechanism (§3.5).

3.4 SAM+D for 3D Volumetric Segmentation

In the 3D setting, we lift SAM’s frozen ViT-B from 2D to 3D (Fig. 1a). Given a 3D volume, we uniformly subsample D_s depth slices and resize each to match SAM’s input resolution. The slices are processed as a batch through the ViT-B encoder, where every transformer block is wrapped with: (i) DSM shift for cross-slice channel exchange with ratio α , (ii) attention with DRLoRA corrections $\Delta\mathbf{Q}$, $\Delta\mathbf{V}$ conditioned on normalised slice position z , followed by (iii) the standard residual connection and frozen feedforward network. Layer norms in each block are left trainable to allow distribution statistics to adapt, while all attention weights and MLP parameters stay frozen. After encoding, per-slice features are reassembled into a 3D feature volume through SAM’s neck projection.

For decoding, we study three designs: a Conv3D decoder that directly up-samples the 3D feature volume, a Multi-Layer Aggregation (MLA) decoder that fuses features from multiple encoder depths, and SAM’s native prompt-based decoder augmented with LoRA. All three achieve competitive performance; SAM’s prompt-based decoder is our default. Full decoder specifications are provided in the supplementary material.

When interactive segmentation is desired, we reuse SAM’s frozen prompt encoder and mask decoder. Since SAM has no notion of depth, we introduce a learnable depth embedding $\mathbf{z}_{\text{embed}}$ that encodes the relative distance between each slice and the prompt point along the z -axis via a two-layer MLP:

$$\mathbf{D}'_d = \mathbf{D}_{\text{prompt}} + \mathbf{z}_{\text{embed}}((z_p - d)/D_s), \quad d = 1, \dots, D_s. \quad (6)$$

This embedding is added to the dense prompt embeddings before mask decoding, so the decoder can modulate predictions based on each slice’s proximity to the annotated slice. LoRA is injected into the Q and V projections of the mask decoder’s attention layers. Per-slice mask logits are then assembled into a 3D volume and refined through a lightweight 3D convolution head for cross-slice smoothing and multi-class output.

3.5 SAM+D for 4D Spatiotemporal Segmentation

In the 4D setting, we lift SAM2 from 2D+T video to 3D+T spatiotemporal segmentation (Fig. 1b). The same DRLoRA and DSM modules are inserted into

SAM2’s Hiera backbone [36] in exactly the same way as the 3D setting—the only architectural difference is that Hiera uses Q-pooling at stage boundaries to progressively reduce spatial resolution, so DRLoRA outputs corrections matching the post-projection dimension at each stage. The depth embedding (Eq. 6), decoder LoRA injection, and 3D refinement head are likewise reused unchanged.

The key distinction from the 3D setting is how temporal dynamics are handled. DRLoRA routes by depth position only, not by time: $z_{b,t,d} = d/D_s, \forall b, t$. Temporal consistency is instead managed through SAM2’s memory architecture, which compresses each frame’s predictions into compact tokens, accumulates them in a memory bank, and cross-attends current-frame features with stored memories before decoding. The memory encoder and memory bank remain frozen, but we inject LoRA into the memory attention module so that the cross-attention between current-frame features and stored memories can be adapted to the volumetric setting. This is necessary because the original memory attention was trained for 2D video frames; adapting it allows the model to learn depth-aware temporal matching patterns specific to 3D+T data. The improved encoder features from DRLoRA + DSM, combined with the adapted memory attention, jointly enable more accurate spatiotemporal propagation.

Training uses a per-frame DiceCE loss without explicit tracking supervision—temporal consistency emerges from SAM2’s memory, as in its original video segmentation paradigm. The set of trainable parameters comprises: DRLoRA experts, layer norms, decoder LoRA, memory attention LoRA, the depth embedding, and the 3D refinement head. Implementation details are provided in supplementary material.

4 Experiments

4.1 Experimental Setup

3D Segmentation Datasets. We evaluate the 3D setting on four CT tumor segmentation benchmarks following the protocol of 3DSAM-adapter [15]. **KiTS** [16]: 209 / 30 / 61 (train/val/test) abdominal CT scans for kidney tumor segmentation. **Pancreas** [1]: 196 / 28 / 57 scans from the Medical Segmentation Decathlon for pancreas tumor segmentation. **LiTS** [3]: 83 / 11 / 24 contrast-enhanced abdominal CT scans for liver tumor segmentation. **Colon** [1]: 89 / 11 / 26 scans for colon cancer segmentation. All datasets use the same train/val/test splits as 3DSAM-adapter [15] for fair comparison.

4D Segmentation and Tracking Datasets. For the 4D (3D+T) setting we evaluate on Fluo-N3DH-SIM+ [41] from the Cell Tracking Challenge (CTC), which contains synthetically generated 3D time-lapse fluorescence microscopy sequences of HL60 cell nuclei produced by the CytoPacq/MitoGen simulator [39]. Each volume spans approximately $650 \times 650 \times 59$ voxels at 29-minute intervals, with cells exhibiting division, migration, and shape deformation. Ground-truth instance segmentation masks and lineage tracking annotations are provided. We train and validate on Sequence 01 and reserve Sequence 02 entirely for testing.

Evaluation Protocol. For 3D segmentation we report the Dice Similarity Coefficient (DSC, %) and the Normalized Surface Distance (NSD, %) at a tolerance of 2mm, following [15]. For 4D segmentation and tracking we adopt the standard CTC metrics [41]: DET (detection accuracy), SEG (segmentation accuracy), TRA (tracking accuracy), and the overall CTC benchmark score $OP_{CTB} = 0.5 (SEG + TRA)$.

During 4D inference, cells may divide or newly enter the field of view. Our tuned SAM2 handles these events by issuing a new point prompt for each newly divided or appeared cell; otherwise, a single prompt on the first frame suffices and the model proceeds end-to-end without post-processing. We compare under two modes following the tracking-by-detection paradigm: **(1) GT segmentation + SAM2 linking:** ground-truth masks are available at every frame; for each cell, SAM2’s memory produces ranked candidate matches to the next frame, and the known masks are used to validate the link, rejecting matches to non-existent cells and falling back to the next-highest-probability candidate. This isolates linking quality from segmentation. **(2) Cellpose detector + SAM2 propagation:** Cellpose [38] is used as the off-the-shelf 3D detector to discover new cells and divisions; SAM2 initialises each cell with a point prompt and propagates its mask forward using its memory bank. We use the decoder’s native predicted-IoU output as a per-mask confidence estimate to arbitrate between the two sources: high-confidence tracklet masks are trusted, while low-confidence frames fall back to Cellpose’s segmentation.

Comparing the two modes quantifies the contribution of SAM2’s learned temporal propagation versus per-frame detection. Detailed hyperparameter settings are provided in the supplementary material.

Implementation Details. We use SAM ViT-B as the frozen backbone for 3D and SAM2 Hiera-B+ for 4D, with DRLora ($E = 4$ experts, rank $r = 16$) and DSM ($\alpha = 25\%$) inserted into each encoder block. For the 3D setting, given a point prompt, we crop a $128 \times 128 \times 128$ volume centered on the prompt location and replicate it to three channels, yielding an input of size $3 \times 128 \times 128 \times 128$. We train for 500 epochs using AdamW with a learning rate of 4×10^{-4} and cosine annealing. The 3D model has ~ 83.78 M total parameters, of which only ~ 2.36 M ($\sim 2.8\%$) are trainable. For the 4D setting, we additionally inject LoRA (rank $r = 16$) into all 4 memory attention layers to adapt temporal propagation. Input volumes are cropped around the target cell with a $2.0\times$ padding factor, yielding per-timestep inputs of size $3 \times 32 \times 256 \times 256$, and each training clip spans $T = 16$ timesteps. We train for 500 epochs with a learning rate of 1×10^{-4} . The 4D model trains ~ 3.11 M ($\sim 3.7\%$) of ~ 83.90 M total parameters. All experiments use 3 NVIDIA RTX 6000 Pro GPUs with mixed-precision (BF16) training and gradient checkpointing. At inference, SAM+D processes a full 128^3 volume in ~ 228.5 ms on a single NVIDIA RTX 6000 Pro GPU. Full hyperparameters are provided in the supplementary material.

4.2 3D Volumetric Segmentation Results

Table 1 compares SAM+D with all baselines on the four CT tumor segmentation benchmarks. All SAM+D results use a single point prompt per volume. We compare against three categories of methods following the evaluation protocol of 3DSAM-adapter [15]: (i) Fully-supervised 3D methods: nnU-Net [20], TransBTS [44], nnFormer [52], Swin-UNETR [40], UNETR++ [37], and 3D UX-Net [25]; (ii) Interactive point-prompt methods: Gaussian Kernel [47], MIDeepSeg [31], GPCIS [53], and Visual Sampler [56]; (iii) SAM-based methods: SAM-B [24] applied per-slice without volumetric adaptation, 3DSAM-adapter [15], and the volumetric SAM adapters Med-SA [45] and MA-SAM [7]. Baseline numbers for categories (i), (ii), and SAM-B/3DSAM-adapter are taken directly from [15]; Med-SA and MA-SAM are fine-tuned per dataset under the same splits. As shown in the table, SAM+D achieves near state-of-the-art performance across virtually every evaluation metric on all four datasets, consistently matching or surpassing both fully-supervised 3D methods and existing SAM-based approaches despite using only a single point prompt. Among SAM-based methods, SAM+D surpasses the directly comparable Med-SA and MA-SAM on all four datasets; we omit SAM-Med3D [42], SegVol [13], and MedSAM2 [32] here because our test splits fall within their pretraining corpora (zero-shot numbers and a contamination-free private liver-cancer CT evaluation are in the supplementary material). We evaluate the three decoder variants described in (§3.4): a Conv3D decoder, a multi-layer aggregation (MLA) decoder, and the SAM prompt decoder with LoRA. Among the three variants, the SAM prompt decoder achieves the highest average Dice and NSD across all four datasets while requiring the fewest trainable parameters (2.57 M), and is therefore used as our default. Please refer to supplementary material for detail. We show qualitative comparisons on one representative case from each of the four datasets in Figure 4.

4.3 4D Spatiotemporal Segmentation and Tracking Results

Table 2 presents the 4D (3D+T) segmentation and tracking results on Fluo-N3DH-SIM+. As shown in Figure 5, our method yields more coherent segmentation than the slice-based SAM2 approach [9] without requiring dedicated tracking algorithms. Notably, under the Cellpose setting all baseline methods produce nearly identical SEG scores ($\sim 54\%$), since their segmentation quality is fully determined by the shared Cellpose detections. SAM2+D breaks this ceiling (56.79%), demonstrating that temporal propagation can refine per-frame segmentation beyond the detector’s quality. The identical DRLoRA and DSM modules improve SAM2 on 3D time-lapse tracking just as they improve SAM on medical CT volumes, validating that dimensional lifting is a general method.

4.4 Ablation Studies

We ablate each component on the LiTS benchmark (1 pt/volume) in Table 3. The component study (a) shows that each module contributes complementary

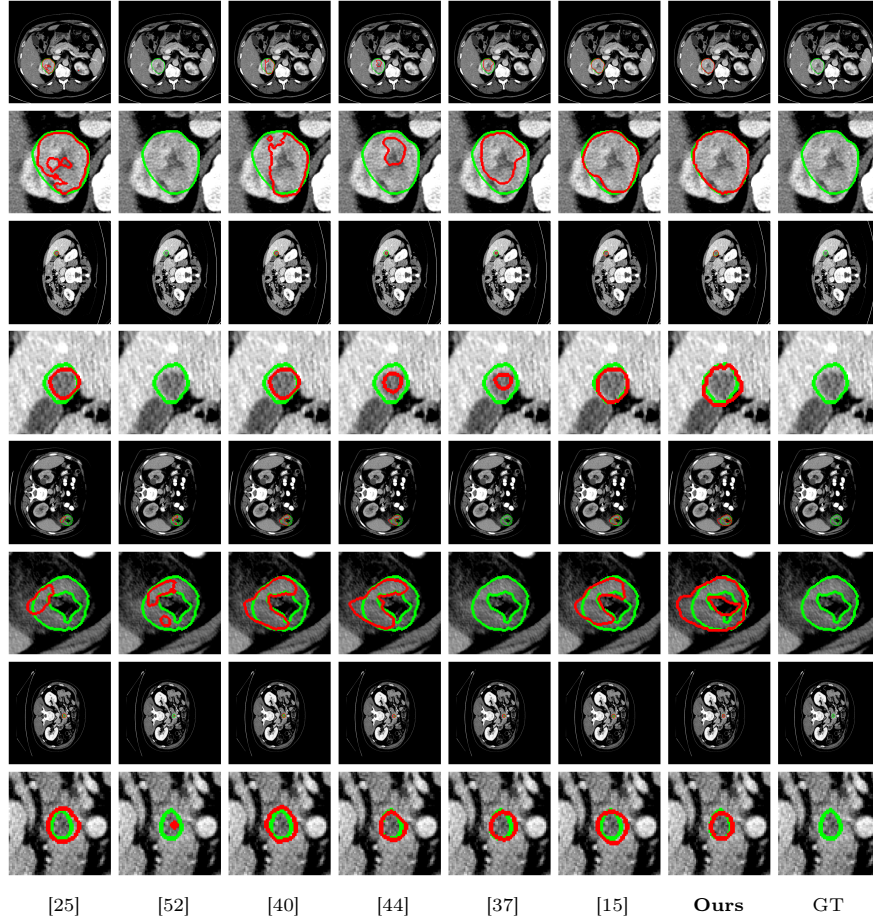


Fig. 4: Qualitative 3D segmentation results. Each pair shows a full slice (top) and an enlarged crop (bottom) for four representative cases. **Green:** GT; **Red:** prediction.

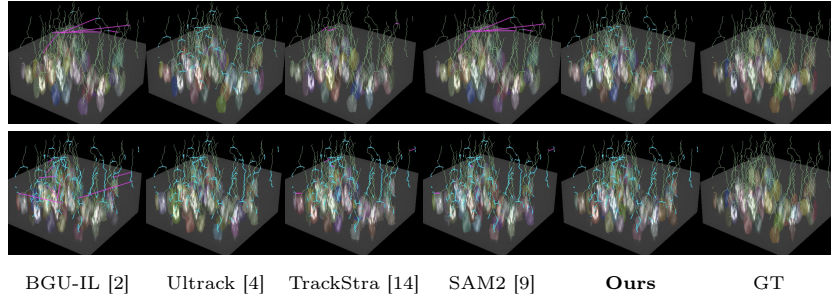


Fig. 5: Visualization on Fluo-N3DH-SIM+. Top: Mode 1 (GT segmentation + propagation). Bottom: Mode 2 (Cellpose [38] + propagation). **Green:** correctly tracked; **Purple:** false positive; **Blue:** false negative. Best viewed when zoomed in.

Table 1: Comparison with fully-supervised, interactive, and SAM-based methods on four tumor segmentation datasets. Dice (%) and NSD (%) are reported

Methods	Kidney Tumor		Pancreas Tumor		Liver Tumor		Colon Cancer		#Trainable Params
	Dice↑	NSD↑	Dice↑	NSD↑	Dice↑	NSD↑	Dice↑	NSD↑	
nnU-Net [20]	73.09	77.42	41.70	62.92	60.27	<u>75.60</u>	44.38	53.66	30.76 M
TransBTS [44]	41.70	38.37	32.00	46.91	35.10	50.08	17.62	22.03	32.33 M
nnFormer [52]	44.78	43.17	35.40	53.30	35.04	44.53	22.04	30.72	149.49 M
Swin-UNETR [40]	66.48	72.44	39.66	58.33	52.16	65.71	33.53	41.16	62.19 M
UNETR++ [37]	57.03	60.09	37.59	53.33	38.74	52.76	25.53	30.41	55.70 M
3D UX-Net [25]	58.98	59.60	34.92	52.39	47.41	62.90	28.62	35.20	53.01 M
Gaussian Kernel (1 pt/vol) [47]	22.32	37.27	36.60	56.79	39.27	53.84	22.54	36.41	31.20 M
Gaussian Kernel (3 pt/vol) [47]	70.56	75.77	53.90	73.34	54.63	71.48	58.99	74.22	31.20 M
MIDeepSeg (1 pt/vol) [31]	41.65	46.98	39.10	64.49	31.99	43.08	45.17	60.67	31.20 M
MIDeepSeg (3 pt/vol) [31]	60.54	64.48	48.58	77.09	39.58	56.58	59.36	76.45	31.20 M
GPCIS (1 pt/vol) [53]	61.54	71.03	41.18	66.12	41.21	58.11	53.11	72.45	31.29 M
GPCIS (3 pt/vol) [53]	68.15	79.68	48.33	76.99	48.40	70.55	57.74	77.16	31.29 M
Visual Sampler (1 pt/vol) [56]	72.47	78.25	49.48	69.49	52.51	64.29	53.49	68.05	33.67 M
Visual Sampler (3 pt/vol) [56]	73.77	79.87	51.13	72.67	54.85	66.73	57.52	71.64	33.67 M
SAM-B (1 pt/slice) [24]	36.30	29.86	24.01	26.74	6.71	7.63	28.83	33.63	–
3DSAM-adapter (1 pt/vol) [15]	80.16	87.40	53.69	76.44	58.02	71.55	59.29	76.10	25.46 M
SAM-B (3 pts/slice) [24]	39.66	34.85	29.80	33.24	7.87	6.76	35.26	39.31	–
3DSAM-adapter (3 pts/vol) [15]	81.50	88.18	<u>54.82</u>	<u>78.60</u>	<u>61.25</u>	77.09	60.93	77.56	25.46 M
Med-SA (1 pt/slice) [45]	70.73	82.41	43.59	73.35	51.61	72.04	53.56	74.85	13.00 M
MA-SAM (no prompt) [7]	69.25	65.84	34.72	47.72	56.28	62.87	50.32	56.37	63.04 M
Ours – Conv3D dec. (1 pt/vol)	82.42	88.89	50.02	67.92	54.06	62.02	66.39	79.85	<u>3.29 M</u>
Ours – MLA dec. (1 pt/vol)	<u>84.16</u>	<u>91.07</u>	51.34	70.52	58.54	67.68	<u>63.70</u>	<u>78.42</u>	10.27 M
Ours – SAM dec. (1 pt/vol)	84.74	92.11	59.68	79.05	63.33	73.52	<u>63.70</u>	77.69	2.57 M

Table 2: 4D segmentation and tracking results on Fluo-N3DH-SIM+ from the Cell Tracking Challenge. The CTC provides two annotated sequences; we train and validate on Seq01 and test on Seq02. SEG (%), DET (%), TRA (%) and OP_{CTB} (%) are reported. ‘–’ in #Trainable Params indicates training-free methods.

Method	Seg.	Source	SEG↑	DET↑	TRA↑	OP_{CTB} ↑	#Trainable Params
BGU-IL [2]	GT		–	–	99.96	99.98	127.70 M
	CellPose		53.80	85.20	84.20	69.00	
Ultrack [4, 5]	GT		–	–	99.23	99.36	–
	CellPose		54.51	85.43	85.17	69.84	
TrackStra [14]	GT		–	–	99.98	99.99	6.90 M
	CellPose		54.15	84.86	84.16	69.16	
SAM2 (slice-based) [9]	GT		–	–	98.18	98.20	–
	CellPose		54.19	84.84	84.14	69.16	
SAM2+D (Ours)	GT		–	–	99.31	99.47	3.11 M
	CellPose		56.79	87.32	87.15	71.97	

gains: DRLoRA adapts **what** features are extracted at each depth, DSM controls **how** they exchange information across slices, and the decoder components refine mask generation with depth-aware cues. The default hyperparameters ($\alpha=0.25$, $K=4$, $r=16$) consistently yield the best performance across all sweeps (b–d).

Table 3: Ablation study on the LiTS dataset. All models use SAM (ViT-B) and SAM’s original decoder. Default settings are marked with *.

Setting	Dice(%)	NSD(%)	Setting	Dice(%)	NSD(%)
<i>(a) Component contribution</i>			<i>(c) DRLoRA experts K</i>		
DRLoRA only	56.41	65.28	$K = 1$ (std. LoRA)	58.86	68.21
+ DSM	59.17	68.74	$K = 2$	61.47	71.30
+ z-embed	61.05	71.03	$K = 4^*$	63.33	73.52
+ Dec. LoRA *	63.33	73.52	$K = 8$	63.10	73.18
<i>(b) DSM shift ratio α</i>			<i>(d) LoRA rank r</i>		
$\alpha = 0.125$	61.22	71.46	$r = 4$	60.72	70.43
$\alpha = 0.25^*$	63.33	73.52	$r = 8$	62.85	73.01
$\alpha = 0.5$	62.08	72.15	$r = 16^*$	63.33	73.52
			$r = 32$	62.19	72.34

Depth routing also outperforms content-based MoE-LoRA routers (MoLoRA, MoLE, MixLoRA) by +7–9 Dice on LiTS under a matched budget (supplementary material, Table 10).

5 Discussion

We have not yet evaluated SAM+D on SAM3 [6], which introduces text-prompted multimodal segmentation; directly applying our modules may not suffice for cross-modal adaptation, and we leave this to future work. Additionally, our 3D pipeline treats a single anatomical axis as depth; averaging predictions across all three axes (sagittal, coronal, transverse) could further improve quality.

For the 4D setting, our method runs end-to-end from a single prompt when tracking a single object. In practice, however, frequent cell divisions and new entries require per-event prompting, increasing inference cost and making fully automatic tracking difficult without an external detector. Moreover, 4D training is computationally demanding (~ 876 GPU-hours per run), which constrained the breadth of our 4D experiments.

In this work, we lift a 2D model to 3D and a 2D+T model to 3D+T by injecting lightweight depth-aware modules along a new axis. The same principle should naturally extend to the temporal dimension: by treating time as the added axis, a pre-trained 2D image model could be lifted to handle video without architecture-level changes. We believe this paradigm of adding one dimension at a time through parameter-efficient adaptation offers a practical path toward extending foundation models to higher-dimensional tasks.

6 Conclusion

We presented SAM+D, a parameter-efficient framework that lifts pre-trained 2D foundation models to 3D and 2D+T models to 3D+T through two lightweight

modules: Depth-Routed LoRA (DRLoRA), which provides depth-conditioned feature adaptation by routing low-rank experts on slice depth, and Depth Shift Module (DSM), which enables zero-parameter inter-slice feature exchange. With fewer than 4% trainable parameters, SAM+D achieves competitive performance against existing methods on four 3D medical segmentation benchmarks and demonstrates effective spatiotemporal tracking on the Cell Tracking Challenge. We hope SAM+D provides a practical and generalisable recipe for extending pre-trained models to higher-dimensional tasks.

Acknowledgements

This work is supported in part by JST CREST, Japan, under Grant JPMJCR25T4, and in part by JSPS KAKENHI, Japan, under Grant 26H00455. We would like to thank Prof. Keiji Nakajima from the Nara Institute of Science and Technology for the valuable support throughout this research.

References

1. Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B.A., Litjens, G., Menze, B., Ronneberger, O., Summers, R.M., et al.: The medical segmentation decathlon. *Nature communications* **13**(1), 4128 (2022)
2. Ben-Haim, T., Raviv, T.R.: Graph neural network for cell tracking in microscopy videos. In: *European Conference on Computer Vision*. pp. 610–626. Springer (2022)
3. Bilic, P., Christ, P., Li, H.B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., Szeskin, A., Jacobs, C., Mamani, G.E.H., Chartrand, G., et al.: The liver tumor segmentation benchmark (lits). *Medical image analysis* **84**, 102680 (2023)
4. Bragantini, J., Theodoro, I., Zhao, X., Huijben, T.A., Hirata-Miyasaki, E., VijayKumar, S., Balasubramanian, A., Lao, T., Agrawal, R., Xiao, S., et al.: Ultrack: pushing the limits of cell tracking across biological scales. *bioRxiv* pp. 2024–09 (2024)
5. Bragantini, J., Lange, M., Royer, L.: Large-scale multi-hypotheses cell tracking using ultrametric contours maps. In: *European Conference on Computer Vision*. pp. 36–54. Springer (2024)
6. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719* (2025)
7. Chen, C., Miao, J., Wu, D., Zhong, A., Yan, Z., Kim, S., Hu, J., Liu, Z., Sun, L., Li, X., et al.: Ma-sam: Modality-agnostic sam adaptation for 3d medical image segmentation. *Medical Image Analysis* **98**, 103310 (2024)
8. Chen, S., Ge, C., Tong, Z., Wang, J., Song, Y., Wang, J., Luo, P.: Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems* **35**, 16664–16678 (2022)
9. Chen, Z., Edgü, M., Jin, E., Stegmaier, J.: Segment anything for cell tracking. In: *International Workshop on Foundation Models for General Medical AI*. pp. 12–22. Springer (2025)

10. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention. pp. 424–432. Springer (2016)
11. Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: Qlora: Efficient fine-tuning of quantized llms. *Advances in neural information processing systems* **36**, 10088–10115 (2023)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020)
13. Du, Y., Bai, F., Huang, T., Zhao, B.: Segvol: Universal and interactive volumetric medical image segmentation. *Advances in Neural Information Processing Systems* **37**, 110746–110783 (2024)
14. Gallusser, B., Weigert, M.: Trackastra: Transformer-based cell tracking for live-cell microscopy. In: European conference on computer vision. pp. 467–484. Springer (2024)
15. Gong, S., Zhong, Y., Ma, W., Li, J., Wang, Z., Zhang, J., Heng, P.A., Dou, Q.: 3dsam-adapter: Holistic adaptation of sam from 2d to 3d for promptable tumor segmentation. *Medical Image Analysis* **98**, 103324 (2024)
16. Heller, N., Isensee, F., Maier-Hein, K.H., Hou, X., Xie, C., Li, F., Nan, Y., Mu, G., Lin, Z., Han, M., et al.: The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge. *Medical image analysis* **67**, 101821 (2021)
17. Hounsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: International conference on machine learning. pp. 2790–2799. PMLR (2019)
18. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. *ICLR* **1**(2), 3 (2022)
19. Hu, J., Li, Y., Sun, H., Song, Y., Zhang, C., Lin, L., Chen, Y.W.: Lga: A language guide adapter for advancing the sam model’s capabilities in medical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 610–620. Springer (2024)
20. Isensee, F., Jaeger, P.F., Kohl, S.A.A., Petersen, J., Maier-Hein, K.H.: nnu-net: A self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods* **18**(2), 203–211 (2021)
21. Jeon, Y.S., Yang, H., Fu, H., Kway, Y., Feng, M.: No more sliding window: efficient 3d medical image segmentation with differentiable top-k patch sampling. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 376–386. Springer (2025)
22. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: European conference on computer vision. pp. 709–727. Springer (2022)
23. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013)
24. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4015–4026 (2023)
25. Lee, H.H., Bao, S., Huo, Y., Landman, B.A.: 3d ux-net: A large kernel volumetric convnet modernizing hierarchical transformer for medical image segmentation. *arXiv preprint arXiv:2209.15076* (2022)

26. Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. In: Proceedings of the 2021 conference on empirical methods in natural language processing. pp. 3045–3059 (2021)
27. Li, D., Ma, Y., Wang, N., Ye, Z., Cheng, Z., Tang, Y., Zhang, Y., Duan, L., Zuo, J., Yang, C., et al.: Mixlora: Enhancing large language models fine-tuning with lora-based mixture of experts. arXiv preprint arXiv:2404.15159 (2024)
28. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 4582–4597 (2021)
29. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 7083–7093 (2019)
30. Liu, S.Y., Wang, C.Y., Yin, H., Molchanov, P., Wang, Y.C.F., Cheng, K.T., Chen, M.H.: Dora: Weight-decomposed low-rank adaptation. In: Forty-first International Conference on Machine Learning (2024)
31. Luo, X., Wang, G., Song, T., Zhang, J., Aertsen, M., Deprest, J., Ourselin, S., Vercauteren, T., Zhang, S.: Mideepseg: Minimally interactive segmentation of unseen objects from medical images using deep learning. *Medical image analysis* **72**, 102102 (2021)
32. Ma, J., Yang, Z., Kim, S., Chen, B., Baharoon, M., Fallahpour, A., Asakereh, R., Lyu, H., Wang, B.: Medsam2: Segment anything in 3d medical images and videos. arXiv preprint arXiv:2504.03600 (2025)
33. Quab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
34. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024)
35. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 10684–10695 (2022)
36. Ryali, C., Hu, Y.T., Bolya, D., Wei, C., Fan, H., Huang, P.Y., Aggarwal, V., Chowdhury, A., Poursaeed, O., Hoffman, J., et al.: Hiera: A hierarchical vision transformer without the bells-and-whistles. In: International conference on machine learning. pp. 29441–29454. PMLR (2023)
37. Shaker, A., Maaz, M., Rasheed, H., Khan, S., Yang, M.H., Khan, F.S.: Unetr++: delving into efficient and accurate 3d medical image segmentation. *IEEE Transactions on Medical Imaging* **43**(9), 3377–3390 (2024)
38. Stringer, C., Wang, T., Michaelos, M., Pachitariu, M.: Cellpose: a generalist algorithm for cellular segmentation. *Nature methods* **18**(1), 100–106 (2021)
39. Svoboda, D., Ulman, V.: Mitogen: a framework for generating 3d synthetic time-lapse sequences of cell populations in fluorescence microscopy. *IEEE transactions on medical imaging* **36**(1), 310–321 (2016)
40. Tang, Y., Yang, D., Li, W., Roth, H.R., Landman, B., Xu, D., Nath, V., Hatamizadeh, A.: Self-supervised pre-training of swin transformers for 3d medical image analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20730–20740 (2022)
41. Ulman, V., Maška, M., Magnusson, K.E., Ronneberger, O., Haubold, C., Harder, N., Matula, P., Matula, P., Svoboda, D., Radojevic, M., et al.: An objective comparison of cell-tracking algorithms. *Nature methods* **14**(12), 1141–1152 (2017)

42. Wang, H., Guo, S., Ye, J., Deng, Z., Cheng, J., Li, T., Chen, J., Su, Y., Huang, Z., Shen, Y., et al.: Sam-med3d: a vision foundation model for general-purpose segmentation on volumetric medical images. *IEEE Transactions on Neural Networks and Learning Systems* (2025)
43. Wang, H., Lin, L., Hu, H., Chen, Q., Li, Y., Iwamoto, Y., Han, X.H., Chen, Y.W., Tong, R.: Patch-free 3d medical image segmentation driven by super-resolution technique and self-supervised guidance. In: *International conference on medical image computing and computer-assisted intervention*. pp. 131–141. Springer (2021)
44. Wang, W., Chen, C., Ding, M., Yu, H., Zha, S., Li, J.: Transbts: Multimodal brain tumor segmentation using transformer. In: *International conference on medical image computing and computer-assisted intervention*. pp. 109–119. Springer (2021)
45. Wu, J., Wang, Z., Hong, M., Ji, W., Fu, H., Xu, Y., Xu, M., Jin, Y.: Medical sam adapter: Adapting segment anything model for medical image segmentation. *Medical image analysis* **102**, 103547 (2025)
46. Wu, X., Huang, S., Wei, F.: Mixture of lora experts. *arXiv preprint arXiv:2404.13628* (2024)
47. Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.S.: Deep interactive object selection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 373–381 (2016)
48. Yang, Y., Xu, L., Tian, L.: Sam2-3dmed: Empowering sam2 for 3d medical image segmentation. *arXiv preprint arXiv:2510.08967* (2025)
49. Zadouri, T., Üstün, A., Ahmadian, A., Ermiş, B., Locatelli, A., Hooker, S.: Pushing mixture of experts to the limit: Extremely parameter efficient moe for instruction tuning. *arXiv preprint arXiv:2309.05444* (2023)
50. Zhang, K., Liu, D.: Customized segment anything model for medical image segmentation. *arXiv preprint arXiv:2304.13785* (2023)
51. Zhang, Q., Chen, M., Bukharin, A., Karampatziakis, N., He, P., Cheng, Y., Chen, W., Zhao, T.: Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512* (2023)
52. Zhou, H.Y., Guo, J., Zhang, Y., Yu, L., Wang, L., Yu, Y.: nnformer: Interleaved transformer for volumetric segmentation. *arXiv preprint arXiv:2109.03201* (2021)
53. Zhou, M., Wang, H., Zhao, Q., Li, Y., Huang, Y., Meng, D., Zheng, Y.: Interactive segmentation as gaussian process classification. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 19488–19497 (2023)
54. Zhu, J., Hamdi, A., Qi, Y., Jin, Y., Wu, J.: Medical sam 2: Segment medical images as video via segment anything model 2. *arXiv preprint arXiv:2408.00874* (2024)
55. Zhuang, J., Wu, L., Ni, X., Wang, X., Wang, L., Chen, H.: Bio2vol: Adapting 2d biomedical foundation models for volumetric medical image segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. pp. 24–34. Springer (2025)
56. Zou, X., Yang, J., Zhang, H., Li, F., Li, L., Wang, J., Wang, L., Gao, J., Lee, Y.J.: Segment everything everywhere all at once. *Advances in neural information processing systems* **36**, 19769–19782 (2023)