Capsule-Expert Routing UNet: A Hybrid 2.5D Convolution-Attention Architecture with Mixture-of-Experts for 3D Medical Segmentation

Nand Kumar Yadav *[1] NAND.YADAV@USD.EDU
Rodrigue Rizk*[1] RODRIGUE.RIZK@USD.EDU
William CW Chen*[2] WILLIAM.CHEN@USD.EDU
KC Santosh*[1] KC.SANTOSH@USD.EDU
[1] *AI Research Lab, Department of Computer Science, University of South Dakota, USA*
[2] *Biomedical & Translational Sciences, Sanford School of Medicine, University of South Dakota, USA*

Abstract

Recent advances in 3D medical image segmentation have been driven by hybrid CNN-Transformer architectures that capture long-range dependencies at the cost of heavy parameters. This paper introduces Capsule-Expert Routing UNet (CER-UNet), a novel encoder–decoder model that achieves strong global context modeling with substantially lower computational parameters. CER-UNet integrates two complementary contributions: (1) a statistical attention module that performs computationally efficient long-range interaction via low-rank covariance pooling and channel-wise statistics, coupled with a 2.5D hybrid convolutional design featuring Inception-style multi-scale depthwise-separable kernels. (2) a Capsule-Expert Mixture-of-Experts (CapMoE) routing mechanism that introduces dynamic feature routing across hierarchical scales, enabling lightweight multi-scale fusion and expert specialization while avoiding the instability of full attention-based routing mechanism. CER-UNet preserves the strong context modeling of recent UNet-like CNN-Transformer hybrids but surpasses them in accuracy-efficiency trade-off.CER-UNet achieves an average Dice of 92.52% on ACDC, 84.94% on BTCV, and 86.64% on Synapse, while using nearly only 32M parameters. Across all three benchmarks, it consistently outperforms competitive Transformer-based methods and conventional 2D/2.5D and 3D segmentation networks, highlighting a strong accuracy-efficiency trade-off. Extensive experiments across multiple 3D medical segmentation benchmarks demonstrate that CER-UNet delivers robust state-of-the-art performance with significantly lower computational overhead. The implementation of CER-UNet is available at https://anonymous.4open.science/r/CER-Unet-BC53.

Keywords: 3D Image Segmentation, Inception-style 2.5D convolutions, Capsule Routing, Mixture-of-experts.

1. Introduction

3D medical image segmentation is fundamental to clinical decision-making, yet remains challenging due to the high resolution and anatomical complexity of volumetric CT and MRI data.

* Contributed equally

While CNN-based architectures such as U-Net (Ronneberger et al., 2015a) effectively capture local spatial patterns, they struggle with long-range dependencies essential for accurate delineation. Transformer-based and hybrid CNN–attention models address this limitation by introducing global context modeling (Azad et al., 2023; Hatamizadeh et al., 2021), but their quadratic attention cost often limits scalability in real-world deployments (Hatamizadeh et al., 2022). Recent efforts to improve efficiency, such as factorized convolutions (Roy et al., 2022), multi-scale aggregation (Zhou et al., 2018), and deformable operations (Dai et al., 2017) highlight the need for architectures that balance accuracy, capacity, and computational practicality. This work addresses this gap by introducing an efficient, scalable approach for high-resolution 3D medical segmentation.

Despite these advances, achieving a balance between model expressiveness and computational cost remains a persistent challenge in 3D medical image analysis. Skip connections, play a crucial role in maintaining spatial detail during the reconstruction phase of segmentation. Traditional U-Net implementations use direct concatenation between encoder and decoder features, but these naive skip connections may inadequately bridge the semantic gap between low-level and high-level feature representations. To address this limitation, recent studies have proposed more advanced skip connection strategies. For instance, gated mechanisms have been employed between encoder and decoder stages (Rahman et al., 2024), while others have incorporated cross-attention mechanisms across the network (Xu et al., 2023). More recently, UTANet (Luo et al., 2025) introduces a Task-Adaptive Mixture of Skip Connections (TA-MoSC), reframing skip fusion as an MoE-style routing problem that dynamically redistributes multi-level encoder features to decoder stages to reduce semantic disparity across tasks and datasets. Nonetheless, the semantic misalignment between encoder and decoder features becomes particularly problematic when dealing with heterogeneous datasets, where variations in data distribution and structure are common. As noted by Wang et al. (Wang et al., 2022), directly transferring features through basic skip connections can lead to suboptimal generalization, especially when semantic information differs significantly between encoding and decoding stages. This mismatch can degrade the model's performance, highlights the need for more semantically aware feature fusion strategies in medical image segmentation frameworks.

To address these limitations, we propose **CER-UNet**: a Capsule-Enhanced Skip connections for encoder-decoder architecture. Our contributions are as follows:

- **We introduce SCBAM:** a statistical, parameter-free CBAM variant that replaces MLP/conv attention using second-order statistics, improving feature focus with negligible overhead.
- **We propose a capsule-inspired MoE** a skip mechanism for 3D medical segmentation, combining structured expert design, balanced multi-gate routing, and Docker-based multi-scale alignment
- **We introduce a Unified architecture:** we combine 2.5D multi-scale encoding, SCBAM, CapMoE, and Docker-based scale alignment into a compact encoder–decoder that improves accuracy while maintaining a favorable parameter–compute budget.

2. Proposed Work

We propose CER-UNet, a parameter efficient yet expressive 3D segmentation architecture that jointly optimizes efficiency, representation quality, and spatiotemporal modeling. To mitigate the high computational cost of 3D convolutions, we introduce 2.5D Inception residual blocks that factorize 3D kernels into a sequence of pointwise, spatial, and temporal depthwise operations, preserving full 3D receptive fields while substantially reducing parameters. On top of this backbone, we design SCBAM, a Statistical attention module that replaces learned MLP/convolutional attention with SimAM-style, parameter-free channel and spatial energies, enabling the network to emphasize salient responses without additional trainable weights. A lightweight transformer-based encoder coupled with residual paths and hierarchical trilinear fusion produces multi-scale tokens (t_1–t_4), which are further refined through Docker-based temporal alignment and CapMoE skip bridges that adaptively route encoder features via capsule-style experts instead of static concatenation.

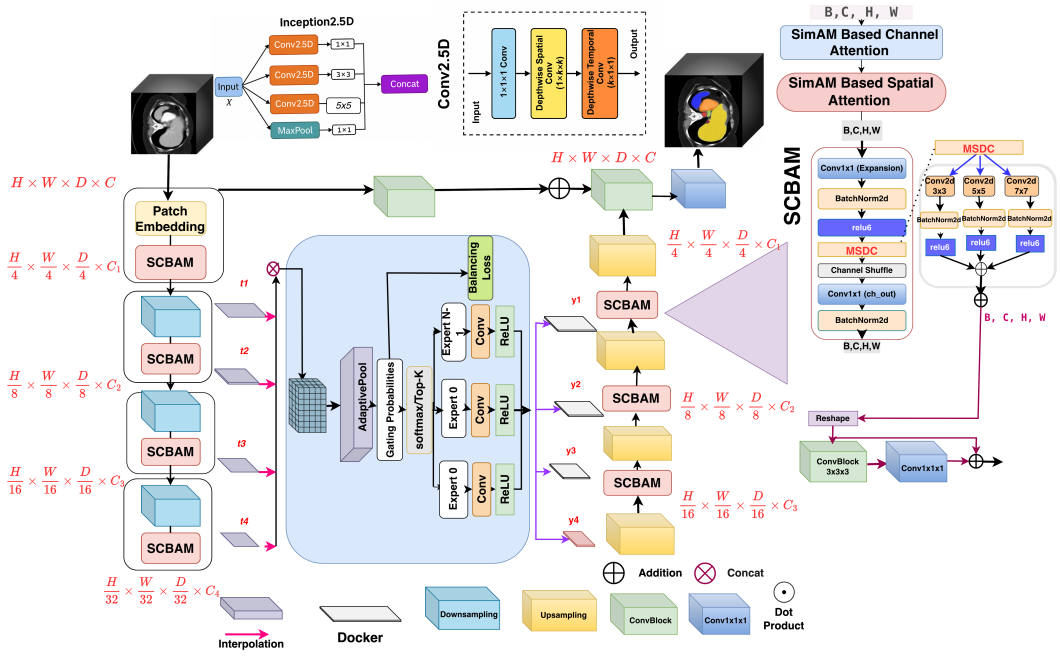

Figure 1: Architecture of the proposed CER-UNet. CER-UNet integrates an Inception-2.5D and statistical attention enhanced (SCBAM) UNET with Docker-based spatio-temporal alignment and a CapMoE, followed by a decoder with multi-scale depthwise convolutions for 3D medical image segmentation.

2.1. 2.5D Factorized Convolutions & Inception Modules

Standard 3D convolutions process all three spatial axes simultaneously, incurring high parameter counts and FLOPs. A single 3D layer with kernel size $k_d \times k_h \times k_w$ has $P_{\text{3D}} = C_{\text{in}} \cdot C_{\text{out}} \cdot k_d \cdot k_h \cdot k_w$.. We approximate a 3D conv by a pointwise $1\times1\times1$ (channel mixing), a spatial depthwise $1\times k_h \times k_w$, and a temporal depthwise $k_d\times1\times1$: $\mathbf{X} \xrightarrow{1\times1\times1} \mathbf{X}' \xrightarrow{1\times k_h\times k_w} \mathbf{X}_s \xrightarrow{k_d\times1\times1} \mathbf{X}_t$. The parameter count becomes $P_{\text{2.5D}} = C_{\text{in}}C_{\text{out}} + C_{\text{out}}\big(k_h k_w + k_d\big)$, instead of $P_{\text{3D}} = C_{\text{in}}C_{\text{out}}k_d k_h k_w$. which helps to retaining a 3D receptive field while substantially reducing computation and memory. These Conv2.5D blocks form the backbone of each encoder and decoder stage.

2.2. SCBAM: Statistical Attention CBAM

Further each encoder and decoder stage equipped with SCBAM attention module. Earlier proposed attention modules like CBAM (Woo et al., 2018) improve focus but typically rely on MLPs (channel) and convolutions (spatial), adding parameters and compute. In contrast, SimAM (Yang et al., 2022) formulates attention as an energy minimization using **second-order statistics**. Building on this idea, we propose SCBAM, which applies SimAM-style energies to both channel and spatial dimensions without learnable parameters as illustrated below. We apply SCBAM to 2D feature maps (per 2.5D stack), $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$.

Channel attention via SimAM energy. For each channel c, the channel attention is computed as

$$\mu_c = \tfrac{1}{HW}\textstyle\sum_{i=1}^{H}\sum_{j=1}^{W} X_{cij}, \quad \sigma_c^2 = \tfrac{1}{HW-1}\textstyle\sum_{i=1}^{H}\sum_{j=1}^{W}(X_{cij}-\mu_c)^2 + \lambda \tag{1}$$

and the energy response as $E_{cij}^{\text{ch}} = \frac{(X_{cij}-\mu_c)^2}{4\sigma_c^2} + \frac{1}{2}$, $X_{cij}^{\text{ch}} = X_{cij} \cdot \sigma(E_{cij}^{\text{ch}})$, where $\sigma(\cdot)$ is the sigmoid and λ is a small constant (10^{-4}) for stability. Higher-variance activations receive larger emphasis, implicitly modeling saliency within each channel.

Spatial attention via Statistical Attention. At each spatial location (i, j), compute

$$\mu_{ij} = \tfrac{1}{C}\textstyle\sum_{c=1}^{C} X_{cij}, \quad \sigma_{ij}^2 = \tfrac{1}{C-1}\textstyle\sum_{c=1}^{C}(X_{cij}-\mu_{ij})^2 + \lambda, \quad E_{ij}^{\text{sp}} = \frac{\sum_c (X_{cij}^{\text{ch}}-\mu_{ij})^2}{4\sigma_{ij}^2} + \frac{1}{2}, \quad \hat{X}_{cij} = X_{cij}^{\text{ch}} \cdot \sigma\big(E_{ij}^{\text{sp}}\big),$$

this yields a shared spatial attention map E_{ij} that modulates all channels without any convolutional filters, this helps the each encoder and decoder blocks to more attentive towards global features without extra trainable params. Further we use the Multi depthwise convolutional block (Rahman et al., 2024) (MSDC) to capture the multiscale features.

2.3. Efficient Multi-Scale Residual Encoder–Decoder

We encode the volumetric input using an SCBAM and Inception-guided UNETR++ (Shaker et al., 2024) inspired encoder. The encoder yields four hierarchical feature representations, $enc_1, enc_2, enc_3, enc_4$. To facilitate efficient cross-scale integration before expert routing, we align these features to a shared spatiotemporal resolution through trilinear interpolation and lightweight channel projections. Concretely, t_1 is obtained from enc_1 using a $1\times1\times1$ 3D projection. We

upsample enc_2 to the target resolution and apply a $1\times1\times1$ projection to form t_2. For t_3, we first perform a $1\times1\times1$ projection and then trilinearly upsample. The deepest transformer token representation enc_4 is projected with a 1×1 1D convolution, expanded into volumetric form, and trilinearly resized to produce t_4. We concatenate t_1, t_2, t_3, t_4 along the channel dimension to construct a unified multi-scale tensor called, $fused(F)$ which serves as a rich context representation for routing and subsequent decoding. In our design, the CapMoE outputs are further organized into scale-specific pathways and compressed by **Docker** blocks that uses 2D convolutions, adaptive pooling, to generate aligned volumetric features y_1, y_2, y_3, y_4 for the decoder. To promote stable optimization and support deeper hierarchies, we incorporate residual learning across all encoder and decoder stages. Each block using 2.5D Inception modules, enabling efficient cross-slice aggregation. This yields an end-to-end multi-scale residual encoder-decoder, well suited for high-resolution volumetric medical segmentation.

2.4. Docker for spatiotemporal features.

As we discussed above, the Docker module transforms each CapMoE output into a spatiotemporal compressed representation that can be aligned with the decoder hierarchy. Given an expert feature map, we reshape it into $\mathbf{X} \in \mathbb{R}^{B\times C\times T\times H\times W}$, where T denotes the number of slices treated as a pseudo-temporal axis. In our implementation, Docker performs lightweight slice-wise refinement using a 1×1 2D projection on each slice, followed by spatiotemporal compression via adaptive *average* pooling. To obtain scale-specific tokens, Docker applies adaptive pooling with temporal and spatial downscaling factors (s_t, s_s): $T' = \max(1, \lfloor T/s_t \rfloor)\,, \quad H' = \max(1, \lfloor H/s_s \rfloor)\,, \quad W' = \max(1, \lfloor W/s_s \rfloor)\,$, yielding a compressed feature volume $\mathbf{Y} \in \mathbb{R}^{B\times C'\times T'\times H'\times W'}$. We employ four Docker blocks with progressively larger (s_t, s_s) to form $\{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4\}$. This hierarchical downscaling constitutes the main multi-scale mechanism in Docker, producing coarse-to-fine spatiotemporal tokens that match successive decoder stages.

2.5. Capsule-experts for skip connections.

Building on the multi-scale features aligned by Docker, we use CapMoE to make skip connections adaptive and spatially aware. In standard U-Net style models, skip connections are static concatenations, which can pass along low-level details that are not always helpful for the decoder at a given stage. CapMoE instead allows the network to selectively refine and re-encode skip information based on the input content, improving semantic alignment between encoder cues and decoder needs. To keep the module lightweight while increasing expressiveness (see Appendix for proofs), each expert is implemented as a simple capsule-inspired convolutional block. Given an input feature map $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$, we compute K parallel capsule responses: $\mathbf{U}_k = \mathrm{Conv}^k_{3\times3}(\mathbf{X}), k = 1, \ldots, K$, concatenate them as, $\mathbf{U} = \mathrm{Concat}(\mathbf{U}_1, \ldots, \mathbf{U}_K)$, apply a capsule-style squashing nonlinearity, and project back to the original channel dimension using a 1×1 convolution:

$$\mathbf{E}(\mathbf{X}) = \mathrm{Conv}_{1\times1}\big(\mathrm{Squash}(\mathbf{U})\big) \in \mathbb{R}^{B\times C\times H\times W}.$$

This design preserves spatial resolution but gives each expert a richer, capsule-like way to reshape channel interactions. The CapMoE outputs are then used as adaptive skip features, which can be further organized by Docker depending on the configuration and injected into the decoder.

Algorithm 1: Capsule Mixture of Experts (CapMoE)

Input: $\mathbf{x} \in \mathbb{R}^{B\times C\times H\times W}$, number of experts E, top-k k
Output: $\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4, \mathcal{L}_{\text{CapMoE}}$
Initialize capsule experts $\{\mathcal{E}_i\}_{i=1}^{E}$
Initialize gating matrices $G_1, G_2, G_3, G_4 \in \mathbb{R}^{C\times E}$ (Xavier)
for $j \leftarrow 1$ **to** 4 **do**
  $\mathbf{z} \leftarrow \text{GAP}(\mathbf{x}) \in \mathbb{R}^{B\times C}$
  $\mathbf{P}_j \leftarrow \text{softmax}(\mathbf{z}G_j) \in \mathbb{R}^{B\times E}$ // gate probs
  $\mathbf{u}_j \leftarrow \sum_{b=1}^{B} \mathbf{P}_j[b,:] \in \mathbb{R}^{E}$ // usage
  $(\mathbf{w}_j, \mathbf{I}_j) \leftarrow \text{TopK}(\mathbf{P}_j, k)$ // $\mathbf{w}_j \in \mathbb{R}^{B\times k}$
  $\mathbf{w}_j \leftarrow \text{softmax}(\mathbf{w}_j)$ // normalize over top-k
  Expand $\mathbf{x}$ to $\mathbf{x}_{\text{exp}} \in \mathbb{R}^{Bk\times C\times H\times W}$
  Initialize $\mathbf{y}_{\text{exp}} \leftarrow 0$ (same shape)
  for $i \leftarrow 1$ **to** E **do**
    $\mathcal{M}_i \leftarrow \{m \mid \mathbf{I}_j[m] = i\}$ // routed slots
    if $\mathcal{M}_i \neq \emptyset$ **then**
      $\mathbf{x}_i \leftarrow \mathbf{x}_{\text{exp}}[\mathcal{M}_i]$
      $\mathbf{y}_{\text{exp}}[\mathcal{M}_i] \leftarrow \mathcal{E}_i(\mathbf{x}_i)$
    end
  end
  Reshape $\mathbf{y}_{\text{exp}} \rightarrow \mathbb{R}^{B\times k\times C\times H\times W}$
  Reshape $\mathbf{w}_j \rightarrow \mathbb{R}^{B\times k\times 1\times 1\times 1}$
  $\mathbf{y}_j \leftarrow \sum_{t=1}^{k} \mathbf{w}_j[:,t] \odot \mathbf{y}_{\text{exp}}[:,t]$
  $\ell_j \leftarrow \dfrac{\text{Var}(\mathbf{u}_j)}{\text{Mean}(\mathbf{u}_j)^2 + \varepsilon}$
end
$\mathcal{L}_{\text{CapMoE}} \leftarrow \sum_{j=1}^{4} \ell_j$
return $\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4, \mathcal{L}_{\text{CapMoE}}$

Algorithm 2: Concise Forward Pass with CapMoE + Docker

Input: $\mathbf{x}_{\text{in}} \in \mathbb{R}^{B\times C_{in}\times D\times H\times W}$
Output: *logits*, $\mathcal{L}_{\text{CapMoE}}$
$[enc_1, enc_2, enc_3, enc_4] \leftarrow encoder(\mathbf{x}_{\text{in}})$
$convBlock \leftarrow stem(\mathbf{x}_{\text{in}})$
Cross-scale alignment to a shared resolution
$t_1 \leftarrow \text{Interpolate}(enc_1)$
$t_2 \leftarrow \text{Interpolate}(enc_2)$
$t_3 \leftarrow \text{Interpolate}(enc_3)$
$t_4 \leftarrow \text{Interp}\big(\text{unsq}(enc_4.\text{permute}(0,2,1))\big)$
Fused $\leftarrow \text{cat}(t_1, t_2, t_3, t_4;\ \text{dim} = 1)$
$(B, C, T, H_s, W_s) \leftarrow \text{shape}(\textbf{Fused})$ // pseudo-temporal T (code uses T=16)
$\mathbf{F}_{2D} \leftarrow \mathbf{F}.\text{permute}(0,2,1,3,4).\text{reshape}(B\cdot T,\ C,\ H_s,\ W_s)$
$(o_1, o_2, o_3, o_4, \mathcal{L}_{\text{CapMoE}}) \leftarrow \text{CapMoE}(\mathbf{F}_{2D})$ // Alg. 1
Docker stage alignment (replacing skips)
$y_1 \leftarrow docker_1(o_1)$
$y_2 \leftarrow docker_2(o_2)$
$y_3 \leftarrow docker_3(o_3)$
$y_4 \leftarrow docker_4(o_4)$
$y_4 \leftarrow \text{TokenProj}(y_4)$ // flatten + 1D conv + linear resize
Decoder with Docker skips
$dec_4 \leftarrow proj\_feat(y_4)$
$dec_3 \leftarrow decoder5(dec_4, y_3)$
$dec_2 \leftarrow decoder4(dec_3, y_2)$
$dec_1 \leftarrow decoder3(dec_2, y_1)$
$out \leftarrow decoder2(dec_1, convBlock)$
if *deep_supervision* **then**
  $logits \leftarrow [\,out1(out),\ out2(dec_1),\ out3(dec_2)\,]$
end
else
  $logits \leftarrow out1(out)$
end
return $(logits,\ \mathcal{L}_{\text{CapMoE}})$

Capsule-MoE routing and load balancing. Following the capsule-expert design above, we use a multi-gate CapMoE to adaptively route the fused skip tokens before scale-wise compression. Let $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$ denote the fused token tensor and let $\{\mathcal{E}_e\}_{e=1}^{E}$ be the capsule experts. We use four independent gates $\{\mathbf{G}_j\}_{j=1}^{4}$, with each $\mathbf{G}_j \in \mathbb{R}^{C\times E}$. For each gate, we first summarize the global context using average pooling, $\mathbf{z} = \text{GAP}(\mathbf{X}) \in \mathbb{R}^{B\times C}$, and compute expert scores as $\mathbf{P}_j = \text{softmax}(\mathbf{z}\mathbf{G}_j) \in \mathbb{R}^{B\times E}$. For every sample, we select the top-k experts according to $\mathbf{P}_j$, renormalize their scores, and route the corresponding features only to these experts. The expert outputs are then combined using the normalized weights to produce a routed feature map $\mathbf{Y}_j \in \mathbb{R}^{B\times C\times H\times W}$. To avoid collapsed routing and encourage diverse expert usage, we include a simple load-balancing term. Let $\mathbf{u}_j \in \mathbb{R}^{E}$ be the batch-wise expert usage for gate j (sum of selection probabilities over the batch). We define $\ell_j = \text{CV}^2(\mathbf{u}_j) = \dfrac{\text{Var}(\mathbf{u}_j)}{\big(\text{Mean}(\mathbf{u}_j)\big)^2 + \varepsilon}$, and aggregate across gates to obtain $\mathcal{L}_{\text{CapMoE}} = \sum_{j=1}^{4} \ell_j$, which is optimized jointly with the segmentation loss. In our implementation, the four routed outputs $\{\mathbf{Y}_j\}_{j=1}^{4}$ are passed to the Docker modules with stage-specific downscaling to form a multi-scale hierarchy for the decoder.

2.6. Decoder Design and Integration with UNET.

The decoder progressively upsamples feature maps while fusing CapMoE-enhanced, scale-aligned skip information, with deep supervision to stabilize training. Built on the latest UNET based UNETR++ backbone, our design replaces static skip fusion with an adaptive pipeline that couples CapMoE routing and Docker-based compression. Specifically, the encoder produces hierarchical outputs $\{enc_1, enc_2, enc_3, enc_4\}$, which are projected and trilinearly resized to a shared intermediate resolution to form $\{t_1, t_2, t_3, t_4\}$. For the deepest tokenized representation enc_4, we apply a 1×1 1D projection and reshape it into a volumetric form before alignment. We then concatenate $\{t_1, t_2, t_3, t_4\}$, rearrange the fused tensor into per-slice 2D features, and route it through CapMoE to obtain four expert-enhanced outputs. These outputs are compressed by Docker into multi-scale representations that are injected into the decoder during progressive upsampling. The overall forward pathway is summarized in Algo. 2 and illustrated in Fig. 1.

3. Experimental Settings, Results & Discussion

We implement CER-UNet in PyTorch with the MONAI (Cardoso et al., 2022) framework. For fair comparison, we use the same input resolutions, preprocessing pipeline, and no additional training data across all baselines (Zhou et al., 2023a), (Isensee et al., 2021), (Xie et al., 2021), (Zhou et al., 2023b). All experiments are conducted on a single NVIDIA A100 GPU for 1,000 epochs with an AdamW optimizer, an initial learning rate of 0.01, and weight decay of 3×10^{-5}. We evaluate on three benchmarks. The Synapse multi-organ CT dataset (Landman et al., 2015) contains 30 scans with 8 organs, using 18 cases for training and 12 for testing; models are trained with 3D patches of size $128 \times 128 \times 64$. Using patch size $96\times96\times96$ on BTCV (Landman et al., 2015), dataset includes 30 CT volumes across 13 abdominal organs. The ACDC (Bernard et al., 2018) cardiac MRI dataset consists of 100 subjects split into 70/10/20 for train/val/test, and we use inputs of size $160 \times 160 \times 16$ to match the 2.5D slice setting. All other training configurations follows baseline those described in (Zhou et al., 2023a).

3.1. Loss Function

The total loss function is composed of segmentation loss $\mathcal{L}_{\text{seg}}$ and the balancing loss from CapMoE ($\mathcal{L}_{\text{CapMoE}}$) as described in Algo. 2. Thus $\mathcal{L}_{total} = \mathcal{L}_{seg} + \lambda\mathcal{L}_{CapMoE}$. The $\mathcal{L}_{\text{seg}}$ combines soft Dice loss and cross-entropy loss:

$$\mathcal{L}_{\text{seg}}(Y, P) = 1 - \sum_{i=1}^{I} \frac{2\sum_{v=1}^{V} Y_{v,i} P_{v,i}}{\sum_{v=1}^{V} (Y_{v,i}^2 + P_{v,i}^2)} - \sum_{v=1}^{V} \sum_{i=1}^{I} Y_{v,i} \log(P_{v,i}). \tag{2}$$

where I is the number of classes, V is the total number of voxels, $Y_{v,i}$ and $P_{v,i}$ are the ground truth and predicted probabilities at voxel v for class i, respectively.

Table 1: Quantitative comparison of segmentation performance using 2D, 2.5D and 3D segmentation models on the Synapse multi-organ dataset. We report Dice Similarity Coefficient (DSC, %) for each organ and HD95. Best results are depicted in **BOLD**.

Method	Spleen	Kidney (R)	Kidney (L)	Liver	Gallbladder	Aorta	Pancreas	Stomach	DSC ↑	HD95 ↓	Params ↓
2D Models											
U-Net (Ronneberger et al., 2015b)	86.67	68.60	77.77	93.43	69.72	89.07	53.98	75.58	76.85	39.70	**14.8M**
TransUNet (Chen et al., 2021)	85.08	77.02	81.87	94.08	63.16	87.23	55.86	75.62	77.49	31.69	96.07M
SwinUNet (Cao et al., 2021)	90.66	79.61	83.28	94.29	66.53	85.47	56.58	76.60	79.13	21.55	27.17M
MISSFormer (Huang et al., 2022)	91.92	82.00	85.21	94.41	68.65	86.99	65.67	80.81	81.96	18.20	42.46M
DAE-Former (Azad et al., 2023)	91.82	82.39	87.66	95.08	71.65	87.84	63.93	80.77	82.63	16.39	48.01M
Cascaded MERIT (Rahman, 2024)	92.01	84.85	87.79	95.26	74.40	87.71	71.81	85.38	84.90	13.22	147.86 M
3D Models											
UNETR (Hatamizadeh et al., 2022)	85.00	84.52	85.60	94.57	56.30	89.80	60.47	70.46	78.35	18.59	92.58M
CoTr (Xie et al., 2021)	94.93	86.80	87.67	96.37	62.90	92.43	78.84	80.46	85.05	**9.04**	46.51M
nnUNet (Isensee et al., 2021)	91.16	86.21	86.92	96.49	69.77	91.78	**83.23**	85.92	86.44	10.91	19.07M
nnFormer (Zhou et al., 2023a)	90.51	86.25	86.57	96.84	70.17	92.04	82.41	86.83	86.57	10.63	149.12M
CER-UNet Using 3D	90.21	**87.41**	87.13	**96.86**	66.28	92.06	82.20	85.70	85.98	11.02	53.37 M
2.5D Models											
TransUNet-2.5D (Zhou et al., 2023b)	**95.61**	83.99	73.28	89.97	71.90	81.33	70.39	**94.59**	84.78	19.24	96.00 M
MOSFormer (Huang et al., 2025)	92.29	83.58	**90.32**	95.96	71.90	88.95	74.14	87.87	85.63	13.40	77.00M
CER-UNet(with Mixture Of Experts (Luo et al., 2025))	87.80	87.37	87.52	96.38	70.81	91.97	76.62	80.90	84.92	11.10	49.12 M
CER-UNet(Ours)	91.37	87.37	87.61	96.71	**73.49**	**92.93**	79.64	84.44	**86.64**	9.57	31.83M

Table 2: Quantitative comparison on BTCV. We report Dice Similarity Coefficient (DSC, %) for each organ and the mean across all organs. Bold indicates the best scores. RAG and LAG refers to the Right Adrenal Gland and Left Adrenal Gland, respectively.

Method	Spleen	RKid	LKid	Gall	Esoph	Liver	Stom	Aorta	IVC	PSV	Pan	RAG	LAG	Avg
nnUNet (Isensee et al., 2021)	95.95	88.35	93.02	**70.13**	**76.72**	96.51	86.79	88.93	82.89	78.51	79.60	73.26	68.35	83.16
CoTr (Xie et al., 2021)	95.80	92.10	93.60	70.00	76.40	96.30	85.40	**92.00**	83.80	78.70	77.50	69.40	66.50	82.88
UNETR (Hatamizadeh et al., 2022)	90.48	82.51	86.05	58.23	71.21	94.64	72.06	86.57	76.51	70.37	66.06	66.25	63.04	76.00
SwinUNETR (Hatamizadeh et al., 2021)	94.59	88.97	92.39	65.37	75.43	95.61	75.57	88.28	81.61	76.30	74.52	68.23	66.02	80.44
nnFormer (Zhou et al., 2023a)	94.58	88.62	93.68	65.29	76.22	96.17	83.59	89.09	80.80	75.97	77.87	70.20	66.05	81.62
TransUNet (Chen et al., 2021)	95.20	92.70	92.20	66.20	75.70	96.90	88.90	**92.00**	83.30	79.10	77.50	69.60	66.60	82.76
CER-UNet (Ours)	**96.06**	**95.24**	**95.23**	65.68	75.93	**97.19**	**90.00**	88.28	**88.02**	**79.70**	**86.48**	**73.63**	**72.83**	**84.94**

3.2. Quantitative Results

We evaluate CER-UNet on three widely used benchmarks: Synapse, BTCV multi-organ CT, and ACDC cardiac MRI, and compare against recent CNN, Transformer, and hybrid segmentation models. Overall, CER-UNet consistently achieves the best or most competitive accuracy with a compact footprint, highlighting an effective balance between representation and efficiency. On Synapse, CER-UNet achieves an average Dice of 86.64% with only 31.83M parameters, outperforming strong baselines such as TransUNet, UNETR, SwinUNet, and nnFormer while remaining substantially lighter than large Transformer counterparts (Table 1). On BTCV, CER-UNet attains the highest mean Dice of 84.94%, delivering clear gains across multiple organs and confirming the robustness of our capsule-expert routing and aligned multi-scale features for abdominal structures

Table 3: ACDC Dataset: Quantitative Comparison using CER-UNet (Dice %).

Method	RV	Myo	LV	Avg
TransUNet (Chen et al., 2021)	88.86	84.54	95.73	89.71
Swin-UNet (Cao et al., 2021)	88.55	85.62	95.83	90.00
UNETR (Hatamizadeh et al., 2022)	85.29	86.52	94.02	88.61
MISSFormer (Huang et al., 2022)	86.36	85.75	91.59	87.90
CoTr (Xie et al., 2021)	89.13	88.64	95.16	91.04
nnUNet (Isensee et al., 2021)	**90.96**	90.34	95.92	92.41
nnFormer (Zhou et al., 2023a)	90.94	89.58	95.65	92.06
EMCAD (Rahman et al., 2024)	90.65	89.68	96.02	92.12
Parallel MERIT (Rahman, 2024)	90.87	90.00	96.08	92.32
CER-UNet (Ours)	90.66	**90.75**	**96.15**	**92.52**

Table 4: Ablation of CER-UNet variants on ACDC (Dice %).

CER-UNet Variant	Params.	FLOPs	RV	Myo	LV	Avg
CER-UNet (skip connections)	**27.22**	**16.87**	90.37	90.66	95.99	92.34
CER-UNet (EPA as in UNETR++)	53.89	163.88	90.44	90.30	96.13	92.51
CER-UNet (MoE (Luo et al., 2025))	30.79	128.16	89.70	90.20	95.99	91.96
CER-UNet (2-Experts)	30.00	154.48	90.63	90.68	96.03	92.44
CER-UNet (Ours)	31.31	154.58	**90.66**	**90.75**	**96.15**	**92.52**

(Table 2). On ACDC, CER-UNet reaches an average Dice of 92.52%, surpassing competitive methods including Parallel MERIT, with consistent improvements across RV, myocardium, and LV (Table 3). These results demonstrate that CER-UNet's SCBAM-enhanced encoder-decoder, and CapMoE skip routing collectively translate into superior accuracy across both multi-organ CT and cardiac MRI settings.

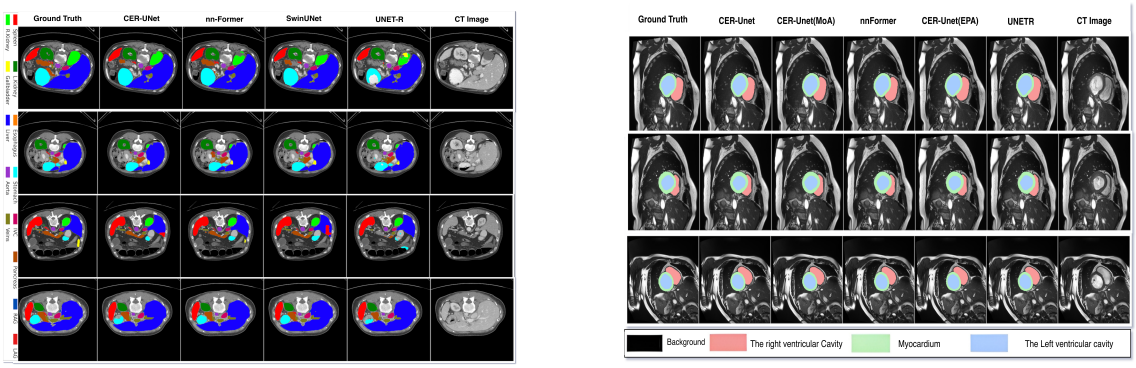

Figure 2: Qualitative results on (left) Synapse and (right) ACDC. CER-UNet produces sharper boundaries on Synapse and more accurately delineates ventricular cavities on ACDC.

3.3. Qualitative Results

Qualitative results in Fig. 2 illustrated the quantitative improvements of CER-UNet. On Synapse, our method yields sharper boundaries and more consistent anatomy for small organs, whereas competing models tend to over-smooth edges. On ACDC, CER-UNet more precisely delineates the ventricular cavities and the thin myocardial wall, particularly in challenging basal slices. These gains align with the role of CapMoE in preserving semantic consistency across scales.

3.4. Ablation Studies on CER-UNet

We ablate CER-UNet on ACDC to isolate the impact of SCBAM and CapMoE (Table 4), comparing (i) standard skips (`NoExperts`), (ii) SCBAM replaced by EPA, (iii) a fixed lightweight expert design (`2-Experts`), and (iv) a conventional MoE (Luo et al., 2025). EPA improves over `NoExperts` but at a higher parameter cost. `2-Experts` further increases Dice, especially on thin structures such as the myocardium, while conventional MoE underperforms `2-Experts` and uses experts inefficiently in this compact regime. CER-UNet achieves the best accuracy with only modest overhead, supporting capsule-structured, balanced routing for adaptive skip fusion. **2.5D Inception vs. full 3D modeling.** On Synapse (Table 1), replacing our 2.5D Inception blocks with full 3D convolutions yields no consistent gains and can degrade performance on small or deformable organs, while incurring higher parameters. This confirms 2.5D Inception as the better accuracy–efficiency trade-off for inter-slice context modeling.

3.5. Limitations and Future Work

While CER-UNet offers a strong accuracy-efficiency trade-off, some limitations remain. First, our 2.5D design is compute-efficient but may not fully capture long-range 3D anatomical continuity; future work will explore hybrid 2.5D-3D designs for volumetric context to improve global reasoning. Second, CapMoE introduces routing overhead that can limit real-time deployment; hardware-aware routing, expert pruning could reduce latency. Third, we primarily evaluate on CT/MRI benchmarks; extending to broader modalities (e.g., ultrasound, PET) and multi-modal settings is important for clinical robustness. Finally, the current Docker pipeline may add system-level overhead; more optimized fused kernels could better realize the model's practical speedups.

4. Conclusion

In this work, we proposed Capsule-Expert Routing UNet (CER-UNet), a compact hybrid 2.5D encoder–decoder architecture for volumetric medical image segmentation that targets a stronger accuracy–efficiency trade-off than heavy CNN, Transformer designs. CER-UNet combines three core ideas: SCBAM, a statistical, parameter-free variant of CBAM to enhance feature selectivity with negligible overhead; a capsule-inspired mixture-of-experts skip strategy (CapMoE) to replace static skip connection with adaptive, semantically aligned routing. Extensive evaluations on Synapse, BTCV, and ACDC demonstrated that CER-UNet consistently achieves highly competitive performance with a streamlined parameter budget. In particular, CER-UNet reached 86.64% average Dice on Synapse with 92.52% on ACDC, and delivered the highest mean Dice of 84.94% on BTCV, confirming the benefit of capsule-expert routing and aligned multi-scale features for complex abdominal anatomy. Our findings highlight an emerging direction for volumetric segmentation:locality-aware multi-scale modeling and structured dynamic routing can surpass large Transformer-heavy designs, offering a principled and scalable backbone for future research in 3D medical image understanding.

References

Reza Azad, René Arimond, Ehsan Khodapanah Aghdam, Amirhossein Kazerouni, and Dorit Merhof. Dae-former: Dual attention-guided efficient transformer for medical image segmentation. In *International workshop on predictive intelligence in medicine*, pages 83–95. Springer, 2023.

Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018.

Hua Cao, Yutong Wang, Joy Chen, Dong Jiang, Xu Zhang, Qi Tian, and Li Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. *arXiv preprint arXiv:2105.05537*, 2021.

M. Jorge Cardoso, Wenqi Li, Tom Brown, Kohei Ito, Jose Montoya, Fredrik Johansson, Christoph Syben, Rahul Deshpande, Nils Gessert, Ali Abdi, et al. MONAI: An Open-Source Framework for Deep Learning in Healthcare. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2022*, pages 265–274. Springer, 2022. doi: 10.1007/978-3-031-18576-2_26.

Jieneng Chen, Yongyi Lu, Qihang Yu, Xuan Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021.

Jifeng Dai, Hao Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017.

Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger R Roth, and Daguang Xu. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *International MICCAI brainlesion workshop*, pages 272–284. Springer, 2021.

Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 574–584, 2022.

De-Xing Huang, Xiao-Hu Zhou, Mei-Jiang Gui, Xiao-Liang Xie, Shi-Qi Liu, Shuang-Yi Wang, Zhen-Qiu Feng, and Zeng-Guang Hou. Mosformer: Momentum encoder-based inter-slice fusion transformer for medical image segmentation, 2025. URL https://arxiv.org/abs/2401.11856.

Xiaohong Huang, Zhifang Deng, Dandan Li, Xueguang Yuan, and Ying Fu. Missformer: An effective transformer for 2d medical image segmentation. *IEEE transactions on medical imaging*, 42(5):1484–1494, 2022.

Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: A self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2):203–211, 2021.

Bennett Landman, Zhoubing Xu, Juan Igelsias, Martin Styner, Thomas Langerak, and Arno Klein. Miccai multi-atlas labeling beyond the cranial vault–workshop and challenge. In *Proc. MICCAI multi-atlas labeling beyond cranial vault—workshop challenge*, volume 5, page 12. Munich, Germany, 2015.

Zichen Luo, Xinshan Zhu, Lan Zhang, and Biao Sun. Rethinking u-net: Task-adaptive mixture of skip connections for enhanced medical image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 5874–5882, 2025.

Rahman. Multi-scale hierarchical vision transformer with cascaded attention decoding for medical image segmentation. In *Medical Imaging with Deep Learning*, pages 1526–1544. PMLR, 2024.

Md Mostafijur Rahman, Mustafa Munir, and Radu Marculescu. Emcad: Efficient multi-scale convolutional attention decoding for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11769–11779, 2024.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015a.

Olaf Ronneberger et al. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015b.

Abhijit Guha Roy, Md Mahfuzur Rahman Siddiquee, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. Mednext: Transformers for medical image segmentation. *arXiv preprint arXiv:2203.05597*, 2022.

Abdelrahman M. Shaker, Muhammad Maaz, Hanoona Rasheed, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Unetr++: Delving into efficient and accurate 3d medical image segmentation. *IEEE Transactions on Medical Imaging*, 2024. doi: 10.1109/TMI.2024.3398728.

Haonan Wang, Peng Cao, Jiaqi Wang, and Osmar R Zaiane. Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 2441–2449, 2022.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

Yutong Xie, Jianpeng Zhang, and Chunhua Shen. Cotr: Efficiently bridging cnn and transformer for 3d medical image segmentation. *arXiv preprint arXiv:2103.03024*, 2021.

Ziqiang Xu, Yuying Zhang, Ming Li, Yifan Wang, and Yizhou Li. Emcad: Ensemble multi-scale cross attention for medical image segmentation. *arXiv preprint arXiv:2301.12345*, 2023.

Hao Yang, Jianwei Zhang, Kun Guo, Jian Zhang, and Yu Qiao. Simam: A simple, parameter-free attention module for convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, 2022.

Hong-Yu Zhou, Jiansen Guo, Yinghao Zhang, Xiaoguang Han, Lequan Yu, Liansheng Wang, and Yizhou Yu. nnformer: volumetric medical image segmentation via a 3d transformer. *IEEE transactions on image processing*, 32:4036–4045, 2023a.

Yucheng Zhou, Yunhao Wang, Xiangde Luo, Yutong Zhang, and Guotai Wang. Transunet 2.5d: Efficient volumetric medical image segmentation by slice-wise token aggregation. *IEEE Transactions on Image Processing*, 32:4742–4755, 2023b. doi: 10.1109/TIP.2023.3282182.

Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11. Springer, 2018.

Theorem and Proofs

Theorem 1 (Spatial Expressivity of Capsule Experts) *Let $\mathcal{F}_{1\times1}$ denote the class of mappings implemented by any finite composition of spatially shared 1×1 convolutions, batch-normalization, and pointwise nonlinearities (as in the baseline Expert (Luo et al., 2025)). Let $\mathcal{F}_{\text{cap}}$ denote the class of mappings implemented by a* CapsuleExpert *consisting of a 3×3 convolution, capsule squashing, and a final 1×1 projection. Then:*

1. *Every $f\in\mathcal{F}_{1\times1}$ is pointwise in space, i.e., $f(x)_{:,h,w}=g(x_{:,h,w})$ for some $g:\mathbb{R}^C\to\mathbb{R}^C$.*
2. *There exists $F\in\mathcal{F}_{\text{cap}}$ and a location (h,w) such that $F(x)_{:,h,w}$ depends on a neighbor $x_{:,h',w'}$ with $(h',w')\neq(h,w)$.*

Consequently, $\mathcal{F}_{1\times1}\subsetneq\mathcal{F}_{\text{cap}}$: CapsuleExperts are strictly more expressive for modeling local spatial dependencies.

Proof A 1×1 convolution followed by batch-normalization and a pointwise nonlinearity acts independently at each spatial location with shared parameters. Thus, any finite composition remains location-wise, yielding $f(x)_{:,h,w}=g(x_{:,h,w})$ for some g, and cannot incorporate information from neighboring pixels.

In contrast, a CapsuleExpert includes a $k\times k$ convolution where $k>1$. For any location (h,w), the pre-squash activation is a linear combination of $\{x_{:,h+i,w+j}\}_{(i,j)\in\{-1,0,1\}^2}$. Choosing a kernel with at least one nonzero off-center coefficient ensures that the output at (h,w) depends on a neighbor (e.g., $x_{:,h,w+1}$). The subsequent squashing and final 1×1 projection are pointwise transformations of this locally mixed signal and therefore do not eliminate the induced dependence. Hence, such an F cannot be represented within $\mathcal{F}_{1\times1}$, proving strict inclusion.

Theorem 2 (Stability of Capsule Squash) *Define $s:\mathbb{R}^d\to\mathbb{R}^d$ by*

$$s(x)=\begin{cases}\dfrac{\|x\|^2}{1+\|x\|^2}\,\dfrac{x}{\|x\|}, & x\neq0,\\ 0, & x=0.\end{cases}\qquad \textit{then}$$

1. $\|s(x)\|=\dfrac{\|x\|^2}{1+\|x\|^2}<1$ *for all x;*
2. *s is globally Lipschitz, i.e., there exists $L>0$ such that $\|s(x)-s(y)\|\le L\|x-y\|$ for all $x,y\in\mathbb{R}^d$.*

Proof Let $r=\|x\|$. For $x\neq0$, $\|s(x)\|=\frac{r^2}{1+r^2}\in[0,1)$, and $s(0)=0$, which proves boundedness. Write $s(x)=\alpha(r)x$ with $\alpha(r)=\frac{r}{1+r^2}$. Then $\alpha'(r)=\frac{1-r^2}{(1+r^2)^2}$, $\alpha(r)+r\alpha'(r)=\frac{2r}{(1+r^2)^2}$.

Both $\alpha(r)$ and $\alpha(r)+r\alpha'(r)$ are uniformly bounded for $r\ge0$. For radial scaling maps of this form, the operator norm of the Jacobian $\nabla s(x)$ is controlled by these two bounded quantities; hence $\sup_x\|\nabla s(x)\|<\infty$. By the mean-value theorem, this yields a global Lipschitz constant L for s.