BioMedInformatics

MDPI

Article

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

Nand Kumar Yadav [1,*], Rodrigue Rizk [1], William C.W. Chen [2] and KC Santosh [1]

[1] AI Research Lab, Department of Computer Science, University of South Dakota, Vermillion, SD 57069, USA; kc.santosh@usd.edu (K.S.)
[2] Biomedical & Translational Sciences, Sanford School of Medicine, University of South Dakota, Vermillion, SD 57069, USA; william.chen@usd.edu
* Correspondence: nand.yadav@usd.edu

Abstract

Background: U-Net-style encoder–decoder architectures are widely used for 3D medical image segmentation, but their conventional skip connections usually transfer encoder features to the decoder through static concatenation. Such direct skip fusion may inadequately address the semantic gap between low-level encoder features and high-level decoder representations, especially in heterogeneous volumetric medical images. This study aims to improve skip-connection fusion by making encoder–decoder feature transfer adaptive, spatially expressive, and scale-aligned. **Methods:** We propose Capsule-Expert Routing UNet (CER-UNet), a skip-enhanced encoder–decoder architecture for 3D medical image segmentation. CER-UNet replaces static skip fusion with a Capsule-based Mixture-of-Experts (CapMoE) routing mechanism that dynamically selects and combines capsule-inspired spatial experts for skip-feature refinement. The routed features are further reorganized using ALIGNER-based multi-scale alignment before being injected into the decoder. The model also incorporates 2.5D Inception-style factorized convolutions and parameter-free Statistical Attention CBAM (SCBAM) to support efficient volumetric representation learning. Experiments were conducted on three public benchmarks: Synapse multi-organ CT, BTCV abdominal CT, and ACDC cardiac MRI. **Results:** CER-UNet achieved average Dice scores of 86.70% on Synapse, 84.94% on BTCV, and 92.52% on ACDC, using approximately 32M parameters in the proposed 2.5D configuration. These results show competitive segmentation performance compared with CNN-based, Transformer-based, and recent hybrid methods while maintaining a compact parameter budget. **Conclusions:** The findings suggest that adaptive skip-connection enhancement through capsule-expert routing and scale-aligned feature fusion can improve the effectiveness of encoder–decoder feature transfer for volumetric medical image segmentation.

Keywords: 3D image segmentation; inception-style 2.5D convolutions; capsule routing; mixture of experts

Check for updates

Academic Editors: Kwang-Sig Lee and Hyuntae Park

Received: 25 June 2026
Revised: 26 August 2026
Accepted: 1 September 2026
Published: 19 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Three-dimensional medical image segmentation [1,2] is central to quantitative analysis of volumetric CT and MRI, but accurate delineation remains difficult because anatomical structures vary widely in size, shape, contrast, and through-plane continuity [3]. Convolutional architectures such as U-Net [4,5] are effective at modeling local image struc-

ture, whereas Transformer-based and hybrid CNN–attention models introduce broader contextual interactions [6–8]. These gains often come with substantial computational cost, particularly for high-resolution 3D inputs [9]. Efficient sequence models such as Mamba [10], factorized convolutions [11,12], multi-scale aggregation [13], and deformable operations [14] reflect a broader effort to preserve useful context without making volumetric models prohibitively expensive.

Skip connections are another important part of this trade-off. Standard U-Net skip pathways pass encoder features directly to the decoder, even though the two branches operate at different semantic levels. Gated fusion [15], cross-attention [16], and more recent mixture-of-experts formulations have therefore been used to make encoder–decoder transfer more selective. UTANet [17], for example, formulates skip fusion as a Task-Adaptive Mixture of Skip Connections (TA-MoSC) that redistributes multi-level encoder information according to the decoding stage. These approaches motivate a practical question: can skip routing remain adaptive and spatially expressive while retaining a compact volumetric model? We address this question with CER-UNet, an encoder–decoder architecture that combines efficient 2.5D representation learning with adaptive capsule-expert skip routing. The main contributions are summarized as follows:

- Efficient volumetric representation: 2.5D factorized multi-scale convolutions are paired with a SimAM-inspired, parameter-free Statistical Attention CBAM (SCBAM) module to retain local and cross-slice information with limited attention overhead.
- Adaptive capsule-expert skip fusion: Spatial capsule-inspired experts refine a fused multi-level encoder representation, while sparse multi-gate routing and ALIGNER-based scale adaptation produce skip features matched to successive decoder stages.
- Routing-aware evaluation: CER-UNet is evaluated on Synapse, BTCV, and ACDC, together with a parameter-matched routing audit and a pathology-skew stress test that measures both segmentation quality and expert-utilization stability.

This design focuses the contribution on how efficient representation, spatial expert transformations, routing, and scale alignment work together within the skip pathway.

2. Materials and Methods

CER-UNet is designed to preserve useful volumetric context while making encoder–decoder feature transfer adaptive rather than fixed. The architecture integrates a 2.5D factorized encoder–decoder with parameter-free statistical attention, capsule-inspired spatial experts, sparse routing, and decoder-stage feature alignment. Within the encoder, the 2.5D Inception blocks decompose volumetric filtering into pointwise channel mixing, in-plane spatial filtering, and through-plane depthwise filtering, while SCBAM modulates the resulting features using activation statistics without introducing additional learned attention weights. The four hierarchical encoder representations are then projected to a common intermediate scale and fused to form a shared multi-scale representation for CapMoE. CapMoE adaptively routes this representation through the capsule-inspired experts, after which the ALIGNER modules transform the routed outputs into decoder-specific skip features at the required spatial scales. These aligned representations are progressively integrated into the decoder to support multi-scale volumetric segmentation. The overall architecture of CER-UNet is shown in Figure 1.

2.1. 2.5D Factorized Convolutions and Inception-Style Representation Learning

Volumetric convolution is effective for capturing spatial continuity in 3D medical images, but applying a full 3D kernel jointly across depth, height, and width substantially increases the computational and parameter cost. For an input with C_{in} channels and C_{out} output channels, a conventional 3D convolution with a kernel size of $k_d \times k_h \times k_w$

requires $P_{3D} = C_{in}C_{out}k_dk_hk_w$ parameters. CER-UNet reduces this cost by decomposing volumetric filtering into complementary channel, in-plane, and through-plane operations. Specifically, a pointwise $1 \times 1 \times 1$ convolution first performs channel mixing, followed by a depthwise $1 \times k_h \times k_w$ operation that captures spatial structure within each slice and a depthwise $k_d \times 1 \times 1$ operation that propagates information across neighboring slices: $\mathbf{X} \xrightarrow{1\times1\times1} \mathbf{X}' \xrightarrow{1\times k_h\times k_w} \mathbf{X}_s \xrightarrow{k_d\times1\times1} \mathbf{X}_t$. Ignoring bias terms, the resulting parameter count is $P_{2.5D} = C_{in}C_{out} + C_{out}(k_hk_w + k_d)$, which is lower than the corresponding full 3D formulation while retaining interactions in both the in-plane and through-plane directions.

Rather than using this factorized operation at a single receptive-field scale, CER-UNet incorporates it within an Inception-style representation block. The purpose of this design is to capture anatomical patterns that appear at different spatial scales while avoiding the cost of repeatedly applying dense 3D convolutions. The factorized paths provide complementary local and broader contextual responses, which are combined to form the feature representation passed to the subsequent attention and encoder stages. In this way, the term *2.5D* describes how volumetric filtering is factorized across spatial dimensions, whereas the Inception-style design describes how multiple receptive-field representations are aggregated within the same stage. These blocks form the basic convolutional representation units used throughout the CER-UNet encoder and decoder.

2.2. SCBAM: Statistical Attention CBAM

Following the feature extraction performed by the 2.5D Inception-style blocks, CER-UNet uses Statistical Attention CBAM (SCBAM) to refine the resulting representations without introducing additional trainable parameters in the attention mechanism. SCBAM is inspired by the energy-based formulation of SimAM [18] but organizes the statistical modulation sequentially along the channel and spatial dimensions. In contrast to conventional CBAM [19], which learns channel and spatial attention through dedicated MLP and convolutional branches, SCBAM derives its attention responses directly from the statistics of the incoming activations. This formulation allows the network to reweight informative responses while avoiding additional learned attention filters. Before SCBAM, a volumetric feature tensor is flattened into a token sequence ($\mathbf{X}_{tok} \in \mathbb{R}^{B\times N\times C}$, where $N = H_vW_vD_v$ denotes the number of volumetric tokens). For statistical attention, these tokens are arranged on an auxiliary 2D grid, with zero padding along the token dimension when required, to obtain $\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$. After SCBAM, the grid is flattened back to the token sequence and subsequently restored to volumetric form. Here, H and W denote the auxiliary SCBAM grid dimensions rather than the original in-plane image dimensions. For the resulting feature map ($\mathbf{X} \in \mathbb{R}^{B\times C\times H\times W}$), SCBAM first estimates the distribution of activations independently for each channel. For channel c, the mean μ_c and variance σ_c^2 are computed as

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

where $\varepsilon_{att} = 10^{-4}$ is included for numerical stability. These statistics measure how strongly an activation deviates from the characteristic response of its channel and are used to define the channel-wise energy term, i.e.,

$$E_{cij}^{ch} = \frac{(X_{cij} - \mu_c)^2}{4\sigma_c^2} + \frac{1}{2}, \qquad X_{cij}^{ch} = X_{cij}\,\sigma\left(E_{cij}^{ch}\right),$$

where $\sigma(\cdot)$ denotes the sigmoid function. The resulting response therefore modulates each activation according to its relative statistical deviation within the corresponding feature channel.

SCBAM subsequently performs spatial modulation on the channel-refined representation $\mathbf{X}^{\mathrm{ch}}$. At each spatial location (i, j), the mean μ_{ij} and variance σ_{ij}^2 are evaluated across the C channels, allowing the module to characterize how strongly that location differs from the surrounding channel responses:

$$\mu_{ij} = \frac{1}{C}\sum_{c=1}^{C} X_{cij}^{\mathrm{ch}}, \qquad \sigma_{ij}^2 = \frac{1}{C-1}\sum_{c=1}^{C}\left(X_{cij}^{\mathrm{ch}} - \mu_{ij}\right)^2 + \varepsilon_{\mathrm{att}}. \tag{2}$$

$$E_{ij}^{\mathrm{sp}} = \frac{\sum_{c=1}^{C}\left(X_{cij}^{\mathrm{ch}} - \mu_{ij}\right)^2}{4\sigma_{ij}^2} + \frac{1}{2}, \qquad \hat{X}_{cij} = X_{cij}^{\mathrm{ch}}\,\sigma\left(E_{ij}^{\mathrm{sp}}\right). \tag{3}$$

Because E_{ij}^{sp} is computed for each spatial position and shared across channels, the spatial stage then adjusts the response at each location based on the information across channels, without requiring an additional learned convolutional attention map.

SCBAM therefore complements the preceding 2.5D Inception-style representation learning: the factorized multi-scale convolutions capture local, contextual, and cross-slice patterns, whereas SCBAM adaptively reweights those responses using their channel-wise and spatial activation statistics. The refined feature map ($\hat{\mathbf{X}}$) is then forwarded to the subsequent encoder or decoder operation.

2.3. Multi-Scale Residual Encoder–Decoder

The statistically refined features produced by the 2.5D Inception–SCBAM stages are organized through a hierarchical encoder [20], which generates four representations at progressively different resolutions, denoted by $\{enc_1, enc_2, enc_3, enc_4\}$. These feature levels preserve information at different levels of the hierarchy, with earlier stages retaining fine anatomical details and deeper stages capturing broader contextual structure. Rather than forwarding each encoder feature independently to the decoder, CER-UNet first brings them to a common intermediate scale so that information from different depths can be combined before adaptive routing.

To achieve this, a lightweight projection and trilinear interpolation are applied according to the resolution of each encoder output. The first-level feature (enc_1) is projected through a $1 \times 1 \times 1$ convolution to obtain t_1, while enc_2 is upsampled and projected to form t_2. Similarly, enc_3 is projected before upsampling to produce t_3. The deepest representation, enc_4, is first projected and reshaped back into volumetric form and is then trilinearly resized to obtain t_4. By mapping all four representations to a compatible resolution, the network can combine fine structural information with deeper semantic context within a single feature space.

The aligned representations are concatenated along the channel dimension as

$$\mathbf{F}_{\mathrm{fused}} = \mathrm{Concat}(t_1, t_2, t_3, t_4),$$

forming a shared multi-scale representation that serves as the input to CapMoE. This fused feature is not transferred directly to the decoder. Instead, CapMoE adaptively transforms it through expert routing and produces four routed outputs corresponding to different decoding stages. Each routed representation is then processed by an ALIGNER block to obtain the scale-matched skip features ($\{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4\}$), which are progressively introduced into the decoder as the spatial resolution is recovered. In this way, the encoder, routing module, and decoder form a continuous feature-processing pathway in which multi-scale information is first consolidated, selectively refined, and finally redistributed according to the requirements of each decoding stage. Residual connections are retained throughout the

encoder and decoder to preserve feature information during these transformations and to support stable optimization.

2.4. Capsule Experts for Adaptive Skip Routing

The fused multi-scale representation ($\mathbf{F}_{\text{fused}}$) is passed to CapMoE, where direct encoder–decoder skip transfer is replaced by content-dependent expert routing. Rather than sending the same encoder representation to every decoder stage, CapMoE uses multiple gating heads to select the capsule-inspired experts that are most relevant to the current input. Each gate activates only the top-k experts, allowing the skip features to be adapted before they are passed to the decoder.

For a given routed feature tensor ($\mathbf{X} \in \mathbb{R}^{N\times C\times H\times W}$), each capsule-inspired expert applies K parallel 3×3 convolutional branches. Their outputs are concatenated, passed through a capsule-style squashing function, and projected back to the original channel dimension using a 1×1 convolution:

$$\begin{aligned} \mathbf{U}_k &= \mathrm{Conv}^k_{3\times 3}(\mathbf{X}), \quad k = 1, \ldots, K, \\ \mathbf{U} &= \mathrm{Concat}(\mathbf{U}_1, \ldots, \mathbf{U}_K), \\ \mathbf{E}(\mathbf{X}) &= \mathrm{Conv}_{1\times 1}\big(\mathrm{Squash}(\mathbf{U})\big) \in \mathbb{R}^{N\times C\times H\times W}. \end{aligned} \tag{4}$$

The 3×3 branches allow each expert to capture information from neighboring locations while preserving the spatial size of the feature map. After the branch outputs are combined, the capsule-style squashing function controls their magnitude before the final 1×1 projection. The resulting expert features are then passed to the routing mechanism, where each gating head uses globally pooled information from the fused encoder representation to score the available experts. The top-k experts are selected, their weights are normalized, and their outputs are combined to form the routed feature. The resulting representation therefore combines the overall encoder context with the local spatial information captured by the selected experts. CapMoE produces four routed outputs for the different decoder stages. These outputs are passed to the ALIGNER modules, which adjust their spatial resolution and channel dimensions to match the corresponding stages of the decoder. The aligned outputs are then used as adaptive skip features instead of being transferred directly from the encoder through fixed skip connections.

The capsule-inspired experts introduce local spatial interactions within the routing pathway by allowing each response to incorporate information from neighboring locations before projection. Theorem 1 formalizes this local dependence, while Theorem 2 shows that the capsule-squash function remains bounded and stable across its input range. Together, these results help explain how the expert transformation behaves, while the routing mechanism, itself, is assessed separately through the routing audit and pathology-skew experiments.

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. The CapMoE routing procedure and the complete network forward pass are summarized in Algorithms 1 and 2, respectively. After the cross-scale fused volume is rearranged slice-wise, as specified in Algorithm 2, let $N = BT$ denote the number of routing tokens, and let $\mathbf{X} \in \mathbb{R}^{N\times C\times H\times W}$ denote the resulting fused routing tensor. The capsule experts are denoted by $\{\mathcal{E}_e\}_{e=1}^{E}$. 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 every routing token using average pooling, i.e., $\mathbf{z} = \mathrm{GAP}(\mathbf{X}) \in \mathbb{R}^{N\times C}$, and compute expert scores as $\mathbf{P}_j = \mathrm{softmax}(\mathbf{z}\mathbf{G}_j) \in \mathbb{R}^{N\times E}$. For every routing token, we select the top-k experts according to $\mathbf{P}_j$, renormalize the selected scores, and route the corresponding feature only

to those experts. The expert outputs are then combined using the normalized weights to produce a routed feature map ($\mathbf{Y}_j \in \mathbb{R}^{N\times C\times H\times W}$). To discourage routing collapse and maintain expert availability, the router is trained with the soft-usage balance term defined in Equations (8) and (9).

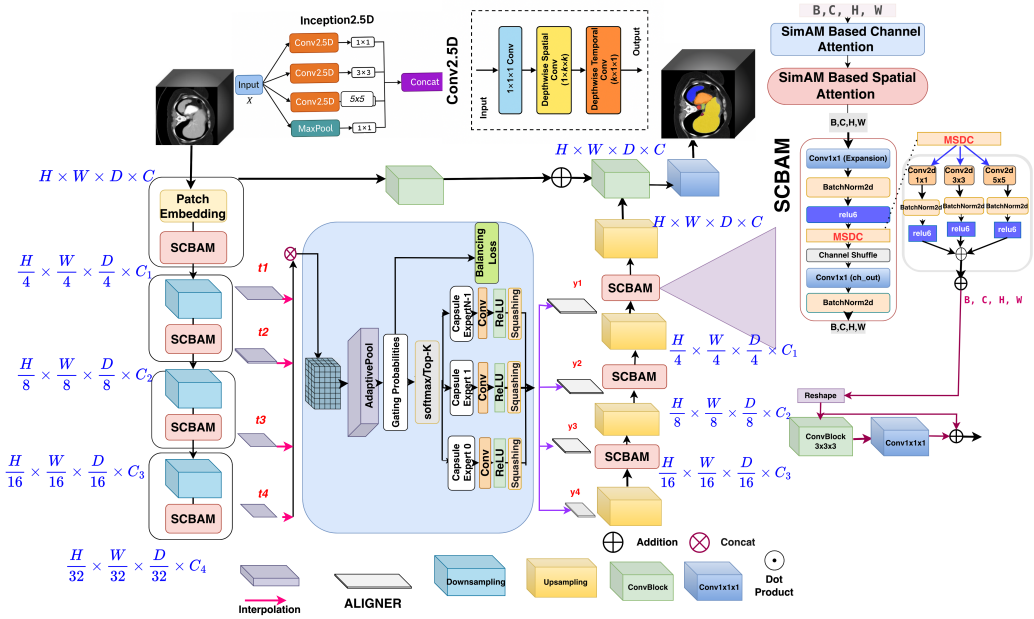

Figure 1. Overview of CER-UNet. The network combines 2.5D Inception-style factorized convolutions, SCBAM statistical attention, CapMoE skip routing, ALIGNER-based scale adaptation, and a multi-scale decoder for volumetric medical image segmentation.

2.5. ALIGNER for Decoder-Stage Feature Alignment

The routed features from CapMoE share a common intermediate scale, while the decoder operates across multiple spatial resolutions. ALIGNER bridges this difference by converting each routed output into a volumetric feature representation that matches the corresponding decoder stage. It first adjusts the channel dimension on a slice-by-slice basis, then restores the volumetric structure before applying scale-specific pooling. For a given slice-wise feature tensor ($\mathbf{X} \in \mathbb{R}^{(B\cdot T)\times C\times H\times W}$, where T denotes the number of slice tokens at the shared routing resolution), ALIGNER applies a 1×1 Conv2D followed by BN and ReLU to each slice independently. The projected features are then reshaped into a 5D volume ($\mathbb{R}^{B\times C'\times T\times H\times W}$) and resized using adaptive average pooling according to the through-plane and in-plane scaling factors (s_d, s_s):

$$\mathrm{ALIGNER}(\mathbf{X}) = \mathrm{AvgPool3D}\big(\mathrm{Reshape}\big(\mathrm{Conv}_{1\times 1}(\mathbf{X})\big)\big). \tag{5}$$

The resulting output sizes are defined as $T' = \max(1, \lfloor T/s_d \rfloor)$, $H' = \max(1, \lfloor H/s_s \rfloor)$, and $W' = \max(1, \lfloor W/s_s \rfloor)$. Four ALIGNER blocks are used with progressively different scaling factors to produce the decoder-specific skip features ($\{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4\}$). In this way, the routed information from CapMoE is reorganized into coarse-to-fine volumetric representations that match the spatial and channel requirements of the successive decoder stages.

2.6. Decoder Integration

The decoder progressively upsamples the deepest representation while incorporating the four CapMoE-enhanced, scale-aligned skip features. The fused encoder tensor is routed in slice-wise form, the four routed outputs are converted back to volumetric features by the ALIGNER modules, and these features are introduced at the corresponding decoding

stages. Deep supervision is retained during training. The complete pathway is summarized in Algorithm 2 and Figure 1.

Algorithm 1: Capsule Mixture of Experts (CapMoE)

Input: $\mathbf{X} \in \mathbb{R}^{N \times C \times H \times W}$, $N = BT$; $E = 4$ experts, $G = 4$ gates, top-k with $k = 2$, and $K = 16$ capsules
Output: $\{\mathbf{Y}_j\}_{j=1}^{G}$ and $\mathcal{L}_{\text{CapMoE}}$
Initialize experts $\{\mathcal{E}_i\}_{i=1}^{E}$ and gates $\{\mathbf{G}_j \in \mathbb{R}^{C \times E}\}_{j=1}^{G}$
$\mathbf{z} \leftarrow \text{GAP}(\mathbf{X}) \in \mathbb{R}^{N \times C}$
for $j \leftarrow 1$ **to** G **do**
  $\mathbf{P}_j \leftarrow \text{softmax}(\mathbf{z}\mathbf{G}_j)$, $\mathbf{u}_j \leftarrow \sum_{n=1}^{N} \mathbf{P}_j[n,:]$
  $(\mathbf{w}_j, \mathbf{I}_j) \leftarrow \text{TopK}(\mathbf{P}_j, k)$, $\mathbf{w}_j \leftarrow \text{softmax}(\mathbf{w}_j)$
  Repeat each token k times to obtain $\mathbf{X}_{\text{exp}} \in \mathbb{R}^{Nk \times C \times H \times W}$; flatten $\mathbf{I}_j$ in the same order
  $\mathbf{Y}_{\text{exp}} \leftarrow \text{ZerosLike}(\mathbf{X}_{\text{exp}})$
  for $i \leftarrow 1$ **to** E **do**
    $\mathcal{M}_i \leftarrow \{m \mid \mathbf{I}_j[m] = i\}$
    if $\mathcal{M}_i \neq \emptyset$ **then** $\mathbf{Y}_{\text{exp}}[\mathcal{M}_i] \leftarrow \mathcal{E}_i(\mathbf{X}_{\text{exp}}[\mathcal{M}_i])$
  Reshape $\mathbf{Y}_{\text{exp}}$ to $N \times k \times C \times H \times W$ and $\mathbf{w}_j$ to $N \times k \times 1 \times 1 \times 1$
  $\mathbf{Y}_j \leftarrow \sum_{t=1}^{k} \mathbf{w}_j[:,t] \odot \mathbf{Y}_{\text{exp}}[:,t]$
  $\ell_j \leftarrow \text{Var}(\mathbf{u}_j)/(\text{Mean}(\mathbf{u}_j)^2 + \varepsilon)$
$\mathcal{L}_{\text{CapMoE}} \leftarrow G^{-1} \sum_{j=1}^{G} \ell_j$
return $\{\mathbf{Y}_j\}_{j=1}^{G}, \mathcal{L}_{\text{CapMoE}}$

Algorithm 2: Forward Pass with CapMoE and ALIGNER Blocks

Input: $\mathbf{x}_{\text{in}} \in \mathbb{R}^{B \times C_{\text{in}} \times D \times H \times W}$
Output: Segmentation logits and auxiliary loss $\mathcal{L}_{\text{CapMoE}}$
$[enc_1, enc_2, enc_3, enc_4] \leftarrow \text{Encoder}(\mathbf{x}_{\text{in}})$; $\mathbf{s} \leftarrow \text{Stem}(\mathbf{x}_{\text{in}})$
for $i = 1,2,3$ **do** $t_i \leftarrow \text{Interpolate}(enc_i)$
$t_4 \leftarrow \text{Interpolate}(\text{Unsqueeze}(enc_4.\text{permute}(0,2,1)))$
$\mathbf{F}_{\text{fused}} \leftarrow \text{Concat}(t_1, t_2, t_3, t_4; \text{dim} = 1)$
$(B, C, T, H_s, W_s) \leftarrow \text{Shape}(\mathbf{F}_{\text{fused}})$
$\mathbf{F}_{2D} \leftarrow \mathbf{F}_{\text{fused}}.\text{permute}(0,2,1,3,4).\text{reshape}(BT, C, H_s, W_s)$
$(o_1, o_2, o_3, o_4, \mathcal{L}_{\text{CapMoE}}) \leftarrow \text{CapMoE}(\mathbf{F}_{2D})$
for $i = 1,2,3,4$ **do** $y_i \leftarrow \text{ALIGNER}_i(o_i)$
$dec_4 \leftarrow \text{ProjFeat}(y_4)$
for $i = 3,2,1$ *(descending)* **do** $dec_i \leftarrow \text{Decoder}_{i+2}(dec_{i+1}, y_i)$
$out \leftarrow \text{Decoder}_2(dec_1, \mathbf{s})$
if deep_supervision **then** $logits \leftarrow [\text{Out}_1(out), \text{Out}_2(dec_1), \text{Out}_3(dec_2)]$
else $logits \leftarrow \text{Out}_1(out)$
return $logits, \mathcal{L}_{\text{CapMoE}}$

2.7. Training Objective and Routing Diagnostics

The network is trained with a segmentation objective and a routing-balance regularizer, i.e.,

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg}} + \lambda_{\text{bal}} \mathcal{L}_{\text{CapMoE}}, \tag{6}$$

where $\lambda_{\text{bal}} = 0.01$ controls the contribution of the routing term.

For one-hot labels (Y) and predicted class probabilities (P), the segmentation term combines soft Dice and voxel-wise cross-entropy:

$$\begin{aligned} \mathcal{L}_{\text{Dice}} &= 1 - \frac{1}{I} \sum_{i=1}^{I} \frac{2 \sum_{v=1}^{V} Y_{v,i} P_{v,i} + \epsilon}{\sum_{v=1}^{V} Y_{v,i}^2 + \sum_{v=1}^{V} P_{v,i}^2 + \epsilon}, \quad \mathcal{L}_{\text{CE}} = -\frac{1}{V} \sum_{v=1}^{V} \sum_{i=1}^{I} Y_{v,i} \log(P_{v,i}), \\ \mathcal{L}_{\text{seg}} &= \mathcal{L}_{\text{Dice}} + \mathcal{L}_{\text{CE}}. \end{aligned} \tag{7}$$

Here, I is the number of classes and V is the number of voxels.

The balance loss is computed from the *soft* gate probabilities before top-k expert selection. For gate g, let $\mathbf{P}^{(g)} \in \mathbb{R}^{N \times E}$ denote the routing probabilities, and let $\mathbf{u}^{(g)} = \sum_{n=1}^{N} \mathbf{P}_{n,:}^{(g)}$ be the corresponding soft expert-usage vector. The per-gate penalty is

$$\ell_g = \mathrm{CV}^2_{\mathrm{soft}}\left(\mathbf{u}^{(g)}\right) = \frac{\mathrm{Var}\left(\mathbf{u}^{(g)}\right)}{\mathrm{Mean}\left(\mathbf{u}^{(g)}\right)^2 + \varepsilon}, \tag{8}$$

and the CapMoE balance loss averages this quantity across the $G = 4$ gating heads:

$$\mathcal{L}_{\mathrm{CapMoE}} = \frac{1}{G} \sum_{g=1}^{G} \ell_g. \tag{9}$$

Therefore, gradients reach the selected routing weights as well as the soft probabilities used to compute the balance loss.

Routing quality is also measured after routing has been realized. Let $\mathbf{h} \in \mathbb{R}^E$ be the normalized expert-load vector over the evaluation set. We report the post hoc diagnostic, i.e.,

$$\mathrm{CV}^2_{\mathrm{load}} = \frac{\mathrm{Var}(\mathbf{h})}{\mathrm{Mean}(\mathbf{h})^2 + \varepsilon},$$

together with normalized entropy, minimum expert-load fraction, maximum-to-minimum load ratio, and starvation counts. Thus, $\mathrm{CV}^2_{\mathrm{soft}}$ is optimized during training, whereas $\mathrm{CV}^2_{\mathrm{load}}$ describes realized expert utilization.

2.8. Theoretical Properties

The following results isolate two properties of the capsule-inspired expert: local spatial dependence and bounded nonlinear response.

Theorem 1 (Local Spatial Expressivity)**.** *Consider an inference-time pointwise expert formed from spatially shared* 1×1 *convolutions, fixed normalization statistics, and pointwise nonlinearities. Its output at location* (h, w) *depends only on the feature vector at* (h, w)*. In contrast, there exists a CapsuleExpert with a* 3×3 *convolution, capsule squashing, and a final* 1×1 *projection whose output at* (h, w) *depends on at least one neighboring location.*

Proof. A composition of 1×1 convolutions, fixed affine normalization, and pointwise nonlinearities acts independently at each spatial position. A 3×3 convolution can instead assign a non-zero coefficient to an off-center location, so the pre-squash activation at (h, w) can depend on a neighboring feature. The subsequent squash preserves this dependence, and the final 1×1 projection can be chosen to retain a coordinate that varies with the neighboring feature. □

Theorem 2 (Stability of Capsule Squash)**.** *For* $x \in \mathbb{R}^d$*, with* $\|\cdot\|$ *denoting the Euclidean norm, we define*

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

Then, $\|s(x)\| < 1$ *for every* x*, and* s *is globally Lipschitz.*

Proof. Let $r = \|x\|$. For $x \neq 0$, $\|s(x)\| = r^2/(1+r^2) < 1$. Writing $s(x) = \alpha(r)x$ with $\alpha(r) = r/(1+r^2)$, the tangential and radial Jacobian eigenvalues are

$$\alpha(r) = \frac{r}{1+r^2}, \qquad \alpha(r) + r\alpha'(r) = \frac{2r}{(1+r^2)^2}.$$

At the origin, $Ds(0) = 0$, since $\|s(x)\|/\|x\| = \|x\|/(1+\|x\|^2) \to 0$ as $x \to 0$.

Their absolute values are uniformly bounded for $r \geq 0$ (in particular, by $9/(8\sqrt{3}) < 1$). Hence, the Jacobian operator norm is globally bounded, which gives a global Lipschitz constant. □

3. Experimental Settings, Results, and Discussion

Tables 1, 2, and 3 summarize the benchmark results on Synapse, BTCV, and ACDC, respectively. Table 4 presents the architectural ablation, and Table 5 reports the computational profile. Tables 6 and 7 present the routing audit and expert-utilization diagnostics, respectively. Table 8 reports the pathology-skew stress test.

CER-UNet is implemented in PyTorch version 1.11.0 using MONAI version 0.7.0 [21]. The implementation of CER-UNet is available at https://github.com/nandkyadavusa/CER-Unet.git. The model is trained for 1000 epochs with AdamW, an initial learning rate of 0.01, and weight decay of 3×10^{-5}. The Synapse experiments use $128 \times 128 \times 64$ patches, BTCV uses $96 \times 96 \times 96$ patches, and ACDC uses $160 \times 160 \times 16$ inputs. Synapse contains 30 CT scans with 18 training and 12 test cases [22]; BTCV contains 30 abdominal CT volumes with 13 target structures [22]; and ACDC contains 100 cardiac MRI subjects split 70/10/20 for training, validation, and testing [23,24]. CER-UNet experiments were run on a single NVIDIA Tesla V100 GPU. The benchmark tables and architectural ablation are based on the single run used for the main comparison. In contrast, the controlled routing audit and pathology-skew stress test are repeated experiments conducted over three matched seeds and are therefore reported as mean ± standard deviation. For this reason, the ACDC Dice score of 92.52% in Tables 3 and 4 should be interpreted as the main benchmark result rather than as the three-seed mean of any routing-audit configuration.

Table 1. Segmentation performance on the Synapse multi-organ dataset. Organ-wise Dice, mean Dice (DSC), HD95, and parameter counts are reported. Boldface marks the best displayed value in each column.

Method	Spleen	Kidney (R)	Kidney (L)	Liver	Gallbladder	Aorta	Pancreas	Stomach	DSC ↑	HD95 ↓	Params ↓
2D Models											
U-Net [4]	86.67	68.60	77.77	93.43	69.72	89.07	53.98	75.58	76.85	39.70	**14.8 M**
TransUNet [25]	85.08	77.02	81.87	94.08	63.16	87.23	55.86	75.62	77.49	31.69	96.07 M
Swin-UNet [26]	90.66	79.61	83.28	94.29	66.53	85.47	56.58	76.60	79.13	21.55	27.17 M
MISSFormer [27]	91.92	82.00	85.21	94.41	68.65	86.99	65.67	80.81	81.96	18.20	42.46 M
DAE-Former [6]	91.82	82.39	87.66	95.08	71.65	87.84	63.93	80.77	82.63	16.39	48.01 M
MERIT (Cascaded) [28]	92.01	84.85	87.79	95.26	**74.40**	87.71	71.81	85.38	84.90	13.22	147.86 M
Swin-UMamba [29]	89.95	81.60	83.66	95.23	70.77	86.32	69.36	81.14	82.26	19.51	28.52 M
MixFormer [30]	89.98	83.19	86.22	95.23	71.53	87.36	66.82	80.77	82.64	12.67	123.35 M
3D Models											
UNETR [9]	85.00	84.52	85.60	94.57	56.30	89.80	60.47	70.46	78.35	18.59	92.58 M
CoTr [31]	**94.93**	86.80	87.67	96.37	62.90	92.43	78.84	80.46	85.05	**9.04**	41.86 M
nnU-Net [32]	91.16	86.21	86.92	96.49	69.77	91.78	**83.23**	85.92	86.44	10.91	30.71 M
nnFormer [33]	90.51	86.25	86.57	96.84	70.17	92.04	82.41	86.83	86.57	10.63	150.2 M
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

Table 1. *Cont.*

Method	Spleen	Kidney (R)	Kidney (L)	Liver	Gallbladder	Aorta	Pancreas	Stomach	DSC ↑	HD95 ↓	Params ↓
2.5D Models											
TransUNet-2.5D [34]	94.59	81.33	83.99	95.61	73.28	89.97	70.39	84.78	84.24	19.24	96.01 M
MOSFormer [35]	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 TA-MoSC [17])	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	**86.70**	9.57	31.83 M

↑ indicates that higher values are better, whereas ↓ indicates that lower values are better.

Table 2. Segmentation performance on BTCV. Dice (%) is reported for 13 abdominal structures, together with the mean across structures. RAG and LAG denote the right and left adrenal glands, respectively. Boldface marks the best displayed value in each column.

Method	Spleen	RKid	LKid	Gall	Esoph	Liver	Stom	Aorta	IVC	PSV	Pan	RAG	LAG	Avg
nnU-Net [32]	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 [31]	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 [9]	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
Swin-UNETR [7]	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 [33]	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 [25]	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
UMamba [10]	93.50	81.12	85.81	66.32	73.37	95.86	80.65	90.70	86.06	70.96	65.49	61.97	62.08	77.98
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**

Table 3. Segmentation performance on ACDC. Dice (%) is reported for the right ventricle (RV), myocardium (Myo), and left ventricle (LV), together with the mean across the three classes. Boldface indicates the best value in each column.

Method	RV	Myo	LV	Avg
TransUNet [25]	88.86	84.54	95.73	89.71
Swin-UNet [26]	88.55	85.62	95.83	90.00
UNETR [9]	85.29	86.52	94.02	88.61
MISSFormer [27]	86.36	85.75	91.59	87.90
CoTr [31]	89.13	88.64	95.16	91.04
nnU-Net [32]	**90.96**	90.34	95.92	92.41
nnFormer [33]	90.94	89.58	95.65	92.06
EMCAD [15]	90.65	89.68	96.02	92.12
Swin-UMamba [29]	89.80	88.37	87.65	88.61
MERIT (Parallel) [28]	90.87	90.00	96.08	92.32
MixFormer [30]	89.02	88.46	95.55	91.01
CER-UNet (Ours)	90.66	**90.75**	**96.15**	**92.52**

Table 4. Ablation study on ACDC comparing alternative skip-fusion, attention, expert-routing, and alignment configurations. CER-UNet achieves the highest mean Dice in this experiment, although the differences among several configurations are small.

CER-UNet Variant	Params. (M)	FLOPs (G)	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	92.29
CER-UNet (TA-MoSC [17])	30.79	128.16	89.70	90.20	95.99	91.96
CER-UNet (strided Conv w/o ALIGNER)	33.20	158.80	90.61	90.54	96.07	92.41
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**

Boldface indicates the lowest parameter count and FLOPs, and the highest Dice score in each segmentation column.

Table 5. Computational profile on Synapse using a Tesla V100 (32 GB) and batch size of 1. FLOPs and latency are reported per 128 × 128 × 64 volume; Mem denotes peak GPU memory under the stated profiling setup. Parameter counts correspond to the implementations instantiated for computational profiling and may therefore differ slightly from literature-reported values in Table 1.

Model	Params (M)	FLOPs (G)	Mem (GB)	GPU Time (ms)
nnU-Net	30.71	476.57	7.65	80.5
UNETR	92.58	86.02	15.24	74.2
Swin-UNETR	62.8	384.2	13.90	176.2
nnFormer	150.2	213.4	9.74	128.2
CoTr	41.86	337.48	8.10	158.4
CER-UNet	31.83	216.4	2.72	78.8

Table 6. Parameter-matched routing audit on ACDC under the standard training distribution. All variants use the same backbone and an effectively matched parameter budget, so the table focuses on segmentation performance, while Table 7 reports the corresponding routing behavior. Values are mean ± standard deviation over three seeds. This audit is separate from both the main ACDC benchmark in Table 3 and the pathology-skew experiment in Table 8. Each audit row was independently retrained under the matched three-seed audit protocol; therefore, these means are not expected to numerically reproduce the single-run 92.52% benchmark value in Table 3.

Routing Variant	RV Dice ↑	Myo Dice ↑	LV Dice ↑	Mean Dice ↑	Mean HD95 ↓
Static uniform	89.27 ± 1.76	89.02 ± 0.86	93.69 ± 1.75	90.66 ± 1.45	4.68 ± 0.97
Dense learned	88.80 ± 0.88	89.02 ± 1.12	94.22 ± 1.13	90.68 ± 0.98	8.93 ± 5.25
Top-*k*, no balance loss	90.45 ± 1.26	88.49 ± 1.84	94.22 ± 2.42	91.05 ± 1.83	6.20 ± 0.78
Top-*k* + soft-CV^2 loss	90.18 ± 1.20	90.16 ± 0.27	95.26 ± 0.23	91.87 ± 0.48	5.68 ± 3.29
Robust CapMoE without volume context	**90.80 ± 1.21**	90.25 ± 0.45	**96.15 ± 0.62**	**92.40 ± 0.44**	**4.54 ± 2.94**
Robust CapMoE with volume context	90.10 ± 1.23	**90.44 ± 0.60**	95.12 ± 0.90	91.89 ± 0.69	7.37 ± 1.76

Boldface indicates the best value in each column. ↑ indicates that higher values are better, whereas ↓ indicates that lower values are better.

Table 7. Expert-utilization diagnostics for the three-seed ACDC routing audit. Diagnostic load (CV^2) is computed from realized expert assignments and is distinct from the soft-CV^2 balance loss optimized during training. Static uniform fusion is shown as a non-adaptive reference.

Routing Variant	Load CV^2 ↓	Normalized Entropy ↑	Min. Load Fraction ↑	Max/Min ↓	Latency (s) ↓
Static uniform	0.0000 ± 0.0000	1.0000 ± 0.0000	0.2500 ± 0.0000	1.000 ± 0.000	**0.326 ± 0.007**
Dense learned	0.4933 ± 0.0126	0.7538 ± 0.0033	0.00049 ± 0.00057	5623.19 ± 8044.07	0.412 ± 0.085
Top-*k*, no balance loss	0.1331 ± 0.1167	0.9513 ± 0.0437	0.1588 ± 0.0483	2.401 ± 0.988	0.405 ± 0.098
Top-*k* + soft-CV^2 loss	0.00540 ± 0.00285	0.99803 ± 0.00104	0.2247 ± 0.0082	1.206 ± 0.055	0.416 ± 0.093
Robust CapMoE without volume context	**0.00132 ± 0.00069**	**0.99953 ± 0.00024**	**0.2402 ± 0.0023**	**1.094 ± 0.021**	0.480 ± 0.100
Robust CapMoE with volume context	0.00222 ± 0.00160	0.99919 ± 0.00058	0.2338 ± 0.0101	1.128 ± 0.068	0.532 ± 0.030

Latency is specific to this ACDC audit and is not directly comparable with the Synapse profiling results in Table 5. The large dense-routing Max/Min reflects its near-zero minimum load. Boldface marks the best adaptive variant for the first four metrics and the lowest latency overall. ↑: higher is better; ↓: lower is better.

3.1. Quantitative Results

Tables 1–3 summarize the benchmark results. On Synapse, CER-UNet reaches a mean Dice of 86.70%, the highest displayed mean in Table 1, with 31.83 M parameters. Its HD95 is 9.57, while CoTr reports the lowest HD95 in the table, at 9.04. The organ-wise results also show that the gain is not uniform across structures: CER-UNet performs strongly on the aorta and achieves the best displayed mean Dice, whereas other methods remain better on individual structures such as the spleen, left kidney, gallbladder, pancreas, and stomach.

Table 8. Pathology-skew stress test on ACDC. Only the training sampler was modified by varying the DCM sampling probability (p); validation and test distributions were unchanged. Values are mean ± standard deviation over three matched seeds. Soft-CV^2 in the routing label denotes the training loss, whereas diagnostic load (CV^2) is computed from realized expert assignments. "Starved seeds" counts runs in which at least one expert received < 1% of the realized routing load. Boldface marks the better value within each matched pair; ties are left unbolded.

DCM p	Routing	Dice (%) ↑	HD95 ↓	Load CV^2 ↓	Normalized Entropy ↑	Min. load Fraction ↑	Max/Min ↓	Starved Seeds ↓
0.20	Top-k, no balance	**91.15 ± 0.41**	**3.03 ± 0.49**	0.3039 ± 0.1154	0.8822 ± 0.0494	0.085 ± 0.055	7.01 ± 4.48	0/3
	Top-k + soft-CV^2	90.23 ± 0.59	3.08 ± 0.46	**0.0119 ± 0.0073**	**0.9957 ± 0.0026**	**0.221 ± 0.011**	**1.28 ± 0.12**	0/3
0.50	Top-k, no balance	**91.09 ± 0.44**	3.19 ± 0.80	0.3064 ± 0.0569	0.8840 ± 0.0274	0.099 ± 0.055	6.29 ± 4.58	0/3
	Top-k + soft-CV^2	90.55 ± 0.69	**2.76 ± 0.10**	**0.0085 ± 0.0060**	**0.9969 ± 0.0021**	**0.225 ± 0.014**	**1.25 ± 0.12**	0/3
0.80	Top-k, no balance	90.53 ± 0.21	3.54 ± 0.25	0.2642 ± 0.2647	0.8788 ± 0.1265	0.087 ± 0.097	46.88 ± 74.79	1/3
	Top-k + soft-CV^2	**90.82 ± 0.50**	**3.00 ± 0.65**	**0.0092 ± 0.0076**	**0.9967 ± 0.0028**	**0.223 ± 0.019**	**1.27 ± 0.17**	**0/3**
0.90	Top-k, no balance	**90.36 ± 0.73**	**3.17 ± 0.61**	0.3112 ± 0.2591	0.8587 ± 0.1228	0.086 ± 0.107	unstable †	1/3
	Top-k + soft-CV^2	89.61 ± 0.77	3.34 ± 0.07	**0.0039 ± 0.0014**	**0.9986 ± 0.0005**	**0.230 ± 0.001**	**1.17 ± 0.01**	**0/3**
0.95	Top-k, no balance	**89.96 ± 0.26**	**3.04 ± 0.23**	0.2103 ± 0.2667	0.9180 ± 0.1058	0.139 ± 0.078	3.97 ± 3.64	0/3
	Top-k + soft-CV^2	89.62 ± 0.38	3.65 ± 0.59	**0.0155 ± 0.0110**	**0.9943 ± 0.0043**	**0.210 ± 0.023**	**1.38 ± 0.18**	0/3

† One unregularized seed assigned zero load to an expert, so the max/min ratio is unbounded and reported as unstable. ↑: higher is better; ↓: lower is better.

On BTCV, CER-UNet obtains the highest displayed mean Dice of 84.94%. It has the better scores for several structures, including the spleen, both kidneys, liver, stomach, inferior vena cava, portal/splenic veins, pancreas, and both adrenal glands. Other methods remain stronger for the gallbladder, esophagus, and aorta. The BTCV result therefore supports strong multi-organ performance within this benchmark rather than uniform dominance on every structure.

On ACDC, CER-UNet achieves the highest displayed mean Dice of 92.52%, with the best scores for Myo (90.75%) and LV (96.15%). For RV, nnU-Net performs slightly better, reaching 90.96%, compared with 90.66% for CER-UNet. Overall, the results across Synapse, BTCV, and ACDC show that CER-UNet remains competitive across both abdominal CT and cardiac MRI while also indicating that some baseline methods perform better for specific structures. Rahman and Marculescu reported two different MERIT variants—namely, parallel and cascaded configurations. To keep the comparison clear, Table 1 refers to the cascaded version as MERIT (Cascaded), while Table 3 refers to the parallel version as MERIT (Parallel).

3.2. Qualitative Results

Figures 2 and 3 provide representative visual examples from Synapse and ACDC. In the qualitative evaluation, CER-UNet preserves several small-organ boundaries and thin myocardial regions with limited leakage into adjacent structures. These observations are consistent with the quantitative results. The Grad-CAM maps in Figure 4 are used only to visualize model-responsive regions; they do not establish causal explanations or clinical utility.

Clinical interpretation and failure-oriented analysis: Mean Dice is useful for benchmark comparison, but a small numerical difference should not be equated with a perceptible clinical improvement. Dice is sensitive to structure size and can hide localized boundary errors in small or thin anatomy [36]. This is visible in the organ-wise results: the Synapse gallbladder and pancreas and the BTCV adrenal glands remain more difficult than large structures such as the liver. HD95 and structure-wise Dice therefore provide important context for the reported averages. For example, the Synapse mean Dice of 86.70% is 0.26 points above nnU-Net (86.44%) and 0.13 points above nnFormer (86.57%). These differences should therefore be interpreted as benchmark-level improvements rather than evidence

of clinically perceptible benefit. Establishing clinical significance would require dedicated reader studies, task-specific clinical endpoints, or prospectively defined thresholds, which were beyond the scope of the present study.

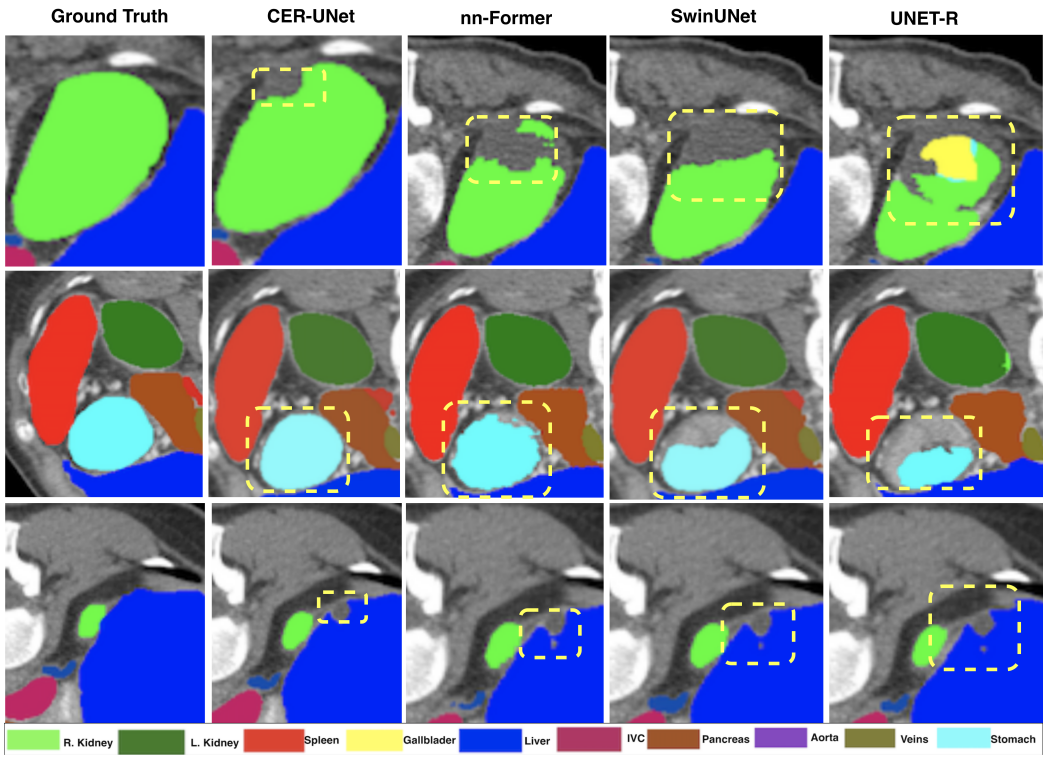

Figure 2. Qualitative comparison on Synapse. Highlighted regions show representative differences.

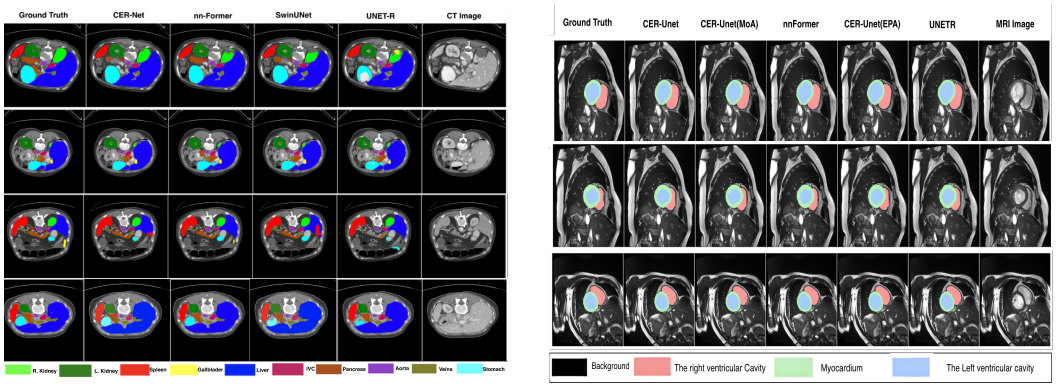

Figure 3. Representative qualitative results on Synapse (**left**) and ACDC (**right**). The examples illustrate differences in boundary placement and ventricular delineation; they are intended as visual complements to the quantitative evaluation rather than as independent evidence of superiority.

3.3. Pathology-Skew Stress Test

To examine how sparse routing behaves when the training distribution becomes strongly imbalanced, we designed a controlled sampling experiment on ACDC. Only the training sampler was modified, while the validation and test sets remained unchanged. Dilated cardiomyopathy (DCM) cases were sampled with probabilities of $p \in \{0.20, 0.50, 0.80, 0.90, 0.95\}$. For each sampling level, we compared parameter-matched top-k routing without the balance loss against the same routing setup trained with the soft-CV^2 objective. Three matched seeds (0, 1, and 2) were used for every condition, resulting in 30 training runs in total.

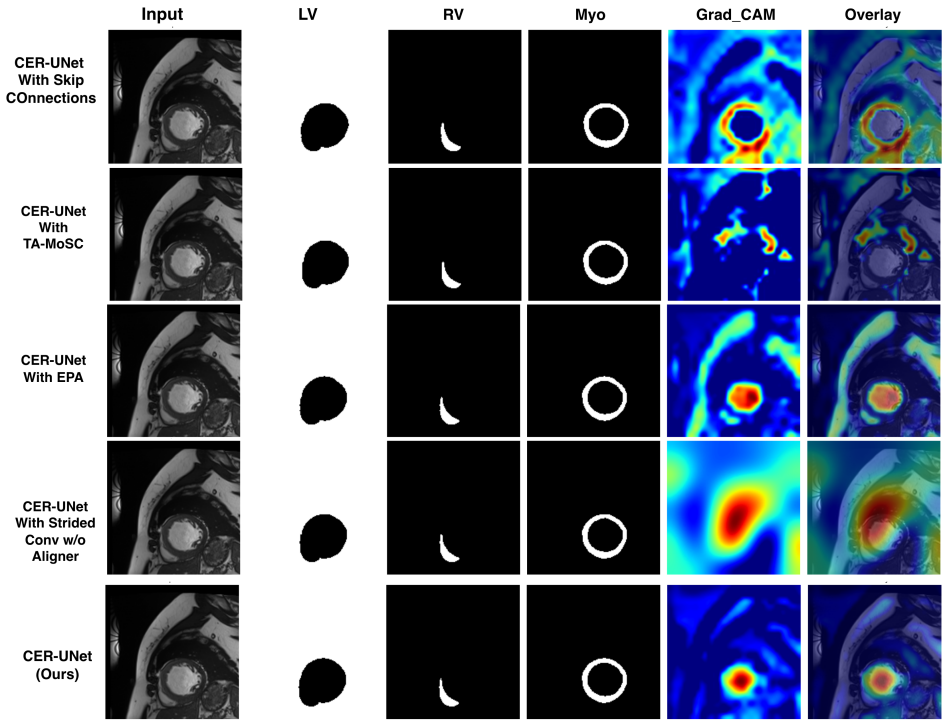

Figure 4. Qualitative comparison of class-wise cardiac segmentation and Grad-CAM model-response visualizations. From left to right, each row shows the input cardiac MRI slice and predicted masks for the left ventricle (LV), right ventricle (RV), and myocardium (Myo), followed by the Grad-CAM map and overlay. The visualizations are included to illustrate differences in model-sensitive regions across variants. They are not interpreted as causal explanations, evidence of diagnostic reasoning, or evidence of clinical utility.

We report the mean Dice, HD95, diagnostic load (CV^2), normalized routing entropy, minimum expert-load fraction, maximum-to-minimum load ratio, and number of seeds in which at least one expert receives less than 1% of the realized routing load. This stress test is evaluated separately from the standard-distribution routing audit reported in Tables 6 and 7. Because each skew condition requires retraining under a different sampling distribution, the Dice and HD95 values are not expected to match those reported in the standard routing audit.

Stress-test interpretation: The routing diagnostics show a clear and consistent benefit from the soft-CV^2 balance loss. Across all five DCM sampling probabilities, the balanced router maintains lower diagnostic load (CV^2), higher normalized entropy, and a larger minimum expert-load fraction than the corresponding unregularized router. This difference becomes more noticeable as the sampling skew increases. At $p = 0.80$, one unregularized seed falls below the 1% starvation threshold, and the mean max/min ratio increases to 46.88 ± 74.79, while the balanced router keeps the ratio at 1.27 ± 0.17 with no starved seed. A similar pattern is observed at $p = 0.90$, where one unregularized seed assigns zero load to an expert, making the max/min ratio unbounded, whereas no starvation occurs when the balance loss is used. The effect on segmentation performance is less consistent. The balanced router achieves a higher mean Dice only at $p = 0.80$ and a lower HD95 at $p = 0.50$ and $p = 0.80$. At the other sampling levels, the unregularized router performs slightly better on one or both segmentation metrics. These results suggest that the main benefit of the soft-CV^2 objective is more stable expert utilization and reduced routing collapse, rather than a consistent improvement in Dice or HD95 across all skew conditions.

3.4. Ablation Studies on CER-UNet

Table 4 compares different skip-fusion, attention, expert-routing, and alignment configurations under the ACDC benchmark setting. The complete CER-UNet achieves the highest mean Dice of 92.52%, although the differences from several simpler variants are relatively small. The two-expert configuration reaches 92.44%, the variant without ALIGNER obtains 92.41%, and the model with static skip connections achieves 92.34%, while TA-MoSC reaches 91.96%. Overall, the ablation results favor the complete CER-UNet configuration, but they also indicate that the final improvement comes from the combined design rather than from a single component alone.

The 92.52% value in the main ACDC benchmark and architectural ablation is a single benchmark result, whereas Tables 6 and 7 constitute a separate controlled three-seed routing audit. The audit is intended to compare routing formulations under a matched repeated-training protocol, not to reproduce the main-table point estimate. To isolate the routing mechanism from the broader architectural ablation, Tables 6 and 7 compare the routing variants using the same backbone and an effectively matched parameter budget over three seeds. Robust CapMoE without volume context provides the strongest overall segmentation result in this audit, with the highest mean Dice ($92.40 \pm 0.44\%$) and the lowest mean HD95 (4.54 ± 2.94). The utilization statistics point in the same direction: the diagnostic load (CV^2) is 0.00132 ± 0.00069, normalized entropy is 0.99953 ± 0.00024, the minimum expert-load fraction is 0.2402 ± 0.0023, and the max/min ratio is 1.094 ± 0.021. Dense learned routing, in contrast, shows a near-zero minimum load (0.00049 ± 0.00057) and a very large max/min ratio, indicating severe imbalance.

Adding the audit-specific volume-context signal does not provide a consistent empirical benefit. The volume-context variant achieves the highest Myo Dice ($90.44 \pm 0.60\%$), but its mean Dice decreases from $92.40 \pm 0.44\%$ to $91.89 \pm 0.69\%$, and its HD95 increases from 4.54 ± 2.94 to 7.37 ± 1.76. The no-volume configuration is also numerically better for RV Dice, LV Dice, and the reported utilization diagnostics. We therefore retain the no-volume variant as the stronger configuration in this controlled audit and do not claim an additional overall benefit from explicit volume context.

Robust CapMoE configuration in the routing audit: To evaluate whether expert utilization can be stabilized beyond the standard top-k router, we define a Robust CapMoE configuration for the controlled routing experiments. The architecture is unchanged and uses four capsule experts, four independent routing gates, and top-k selection with $k = 2$. For gate g, the pooled routing representation ($\mathbf{z}_n \in \mathbb{R}^C$) is projected to expert logits:

$$\mathbf{a}_n^{(g)} = \mathbf{z}_n \mathbf{G}^{(g)}.$$

During training only, zero-mean Gaussian noise with a standard deviation of 0.1 is added to the router logits before expert selection to encourage exploration and prevent premature expert collapse. Routing probabilities are subsequently computed as

$$\mathbf{p}_n^{(g)} = \mathrm{softmax}(\mathbf{a}_n^{(g)}),$$

and the two experts with the largest logits are selected for each routing token. Their selected logits are renormalized with a softmax before the expert outputs are combined. This audit-specific Robust CapMoE rule is intentionally distinct from the standard CapMoE formulation in Algorithm 1, which selects the top-k entries from the full softmax probabilities, then renormalizes the selected scores.

The robust auxiliary objective combines complementary soft- and hard-routing regularizers. Let

$$I_e^{(g)} = \frac{1}{N} \sum_{n=1}^{N} p_{n,e}^{(g)}$$

denote the mean soft importance assigned to expert e, and let $H_e^{(g)}$ denote the corresponding fraction of hard top-k assignments. The switch-style term is

$$\mathcal{L}_{\text{switch}} = E \sum_e \text{stopgrad}\Big(H_e^{(g)}\Big) I_e^{(g)},$$

while the soft importance imbalance is measured using

$$\mathcal{L}_{\text{CV}} = \text{CV}^2\Big(\mathbf{I}^{(g)}\Big).$$

Additionally we use the router-logit stabilization term, i.e.,

$$\mathcal{L}_z = \frac{1}{N} \sum_n \left[\log \sum_e \exp(a_{n,e}^{(g)}) \right]^2,$$

and a differentiable starvation penalty, i.e.,

$$\mathcal{L}_{\text{starve}} = \frac{1}{GE} \sum_{g,e} \Big[\max\Big(0, 0.05 - I_e^{(g)}\Big) \Big]^2.$$

After averaging the gate-wise quantities, the robust routing objective is

$$\mathcal{L}_{\text{robust}} = \mathcal{L}_{\text{switch}} + 0.1\, \mathcal{L}_{\text{CV}} + 10^{-3} \mathcal{L}_z + 0.1\, \mathcal{L}_{\text{starve}}.$$

An expert is counted as starved for diagnostic reporting when its mean hard routing load falls below 1%.

Volume-context ablation:

The robust-routing audit also tests whether a simple volume-level descriptor improves routing beyond the slice-wise gate representation. CapMoE receives the fused feature tensor as consecutive 2D slice tokens and first obtains the slice-wise routing descriptor:

$$\mathbf{z}_{v,t} = \text{GAP}(\mathbf{X}_{v,t}).$$

For the ACDC experiments, each volume contains $T = 16$ routing tokens. Their mean representation is

$$\bar{\mathbf{z}}_v = \frac{1}{T} \sum_{t=1}^{T} \mathbf{z}_{v,t}.$$

The volume-context variant uses

$$\tilde{\mathbf{z}}_{v,t} = \mathbf{z}_{v,t} + \tanh(\gamma) \bar{\mathbf{z}}_v,$$

where γ is a learnable scalar initialized to zero. The same volume descriptor is therefore shared across the T slice tokens of a volume, while the learnable scale allows the router to determine how strongly that context should modify the original slice-wise representation. In *Robust CapMoE without volume context*, this operation is disabled, and the gate receives $\mathbf{z}_{v,t}$ directly. All remaining expert, routing, and optimization settings are identical between the two robust variants.

3.5. Computational Analysis

Table 5 summarizes the computational profile of CER-UNet and representative 3D segmentation models on the Synapse setting using a 128 × 128 × 64 input and batch size of 1. CER-UNet contains 31.83 M parameters and requires 216.4 GFLOPs per volume, while its peak GPU memory consumption is approximately 2.72 GB and the measured inference time is 78.8 ms per volume. In terms of model size, CER-UNet remains comparable to nnU-Net (30.71 M parameters) and is substantially smaller than UNETR (92.58 M), Swin-UNETR (62.8 M), nnFormer (150.2 M), and CoTr (41.86 M). CER-UNet also requires considerably fewer FLOPs than nnU-Net (476.57 G), Swin-UNETR (384.2 G), and CoTr (337.48 G), while its computational cost is close to nnFormer (213.4 G); UNETR remains the least computationally intensive model in terms of FLOPs, at 86.02 G. The most notable advantage of CER-UNet is its memory footprint, which is the lowest among the compared models, at 2.72 GB, compared with 7.65 GB for nnU-Net, 15.24 GB for UNETR, 13.90 GB for Swin-UNETR, 9.74 GB for nnFormer, and 8.10 GB for CoTr. CER-UNet also achieves the second-lowest measured inference latency, at 78.8 ms, slightly behind UNETR (74.2 ms) but faster than nnU-Net (80.5 ms), nnFormer (128.2 ms), CoTr (158.4 ms), and Swin-UNETR (176.2 ms). Overall, these results indicate that CER-UNet provides a favorable memory–latency trade-off while maintaining a moderate computational cost, rather than uniformly minimizing every complexity measure.

3.6. Limitations and Future Work

Several limitations define the scope of the present evidence. First, Synapse, BTCV, and ACDC are evaluated independently within their benchmark distributions. Performance across CT and MRI tasks does not establish frozen-checkpoint generalization to unseen hospitals, scanner vendors, acquisition protocols, or anatomical domains.

Second, the pathology-skew experiment addresses a narrower question: whether expert utilization remains stable when the ACDC training sampler is deliberately imbalanced. The soft-CV^2 loss consistently improves the routing diagnostics and prevents the starvation events observed in the unregularized router, but it does not improve Dice or HD95 at every skew level. The experiment therefore supports the stability of expert utilization under pathology-prevalence imbalance, not general robustness to unrelated domain shifts.

Third, the main tables are benchmark comparisons. Small differences in Dice are not demonstrated clinical effects, and no reader study, manual-correction analysis, treatment-planning endpoint, or prospective clinical threshold was evaluated. The qualitative examples and Grad-CAM maps should be interpreted within the same limitation.

Finally, the 2.5D design emphasizes efficient in-plane and local through-plane processing. It may not capture long-range 3D continuity as completely as a fully volumetric architecture, and CapMoE/ALIGNER introduces additional routing and alignment operations relative to a static skip connection. Future work should therefore include frozen-checkpoint multi-center and multi-vendor evaluation, structure-size and failure-mode stratification, surface-distance and calibration metrics, and reader or workflow-based assessment.

4. Conclusions

CER-UNet combines 2.5D factorized representation learning, parameter-free statistical attention, capsule-inspired spatial experts, sparse routing, and decoder-stage alignment within an adaptive skip pathway. Across Synapse, BTCV, and ACDC, the model achieves competitive benchmark performance with a compact parameter budget. The routing experiments add a more specific result. Under the standard ACDC audit, the no-volume Robust CapMoE variant provides the best overall Dice and HD95 among the evaluated routing configurations, while the pathology-skew experiment shows that the soft-CV^2 ob-

jective keeps expert utilization balanced and prevents the collapse events observed without balancing. The routing gain is therefore most clearly supported as improved utilization stability rather than as a uniform segmentation-accuracy gain across skew conditions.

Overall, the results support adaptive capsule-expert routing as a practical mechanism for refining encoder–decoder feature transfer. External multi-center validation and clinically meaningful workflow endpoints remain necessary before drawing conclusions about deployment or cross-institutional robustness.

Author Contributions: Conceptualization, R.R. and K.S.; methodology, N.K.Y., R.R., W.C.W.C. and K.S.; software, N.K.Y.; validation, N.K.Y. and W.C.W.C.; formal analysis, N.K.Y. and W.C.W.C.; investigation, N.K.Y.; resources, W.C.W.C. and K.S.; data curation, N.K.Y.; writing original draft preparation, N.K.Y.; writing review and editing, N.K.Y., R.R., W.C.W.C. and K.S.; visualization, N.K.Y.; supervision, W.C.W.C. and K.S.; project administration, W.C.W.C. and K.S.; funding acquisition, W.C.W.C. and K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work received partial support from the National Science Foundation (Grant No. 2346643, https://www.nsf.gov/awardsearch/showAward?AWD_ID=2346643 (accessed on 31 August 2026)), the United States Defense Established Program to Stimulate Competitive Research (DEPSCoR) (Award No. FA9550-23-1-0495, https://dtic.dimensions.ai/details/grant/grant.14525543 (accessed on 31 August 2026)), the U.S. Department of Education (Grant No. P116Z240151), the University of South Dakota, the South Dakota Board of Regents Governor's Research Fund, and the United States National Institutes of Health (Grant No. 1R01EB038224). Any opinions, interpretations, conclusions, or recommendations expressed herein are those of the authors and should not be construed as representing the official views of the funding agencies.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new dataset was generated in this study. The experimental results reported in this article were obtained using publicly available benchmark datasets. The Synapse multi-organ CT dataset is available through the MICCAI 2015 Multi-Atlas Abdomen Labeling Challenge/Synapse repository at https://www.synapse.org/Synapse:syn3193805/wiki/217789 (accessed on 31 August 2026). The BTCV abdominal CT dataset used in this study is available from the Beyond the Cranial Vault benchmark data source, as cited in the manuscript [22]. The ACDC cardiac MRI dataset is publicly available from the Automated Cardiac Diagnosis Challenge website at https://www.creatis.insa-lyon.fr/Challenge/acdc/ (accessed on 31 August 2026). The implementation of CER-UNet is available at https://github.com/nandkyadavusa/CER-Unet.git (accessed on 31 August 2026).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xing, J.; Zhang, J. Segmentation of Brain Tumors Using a Multi-Modal Segment Anything Model (MSAM) with Missing Modality Adaptation. *Bioengineering* **2025**, *12*, 871. [CrossRef]
2. Yu, B.; Markham, H.; Moutasim, K.; Foria, V.; Liu, H. ClinSegNet: Towards Reliable and Enhanced Histopathology Screening. *Bioengineering* **2025**, *12*, 1156. [CrossRef] [PubMed]
3. Babaeipour, R.; Fox, M.S.; Parraga, G.; Ouriadov, A. Comparative analysis of foundational, advanced, and traditional deep learning models for hyperpolarized gas MRI lung segmentation: Robust performance in data-constrained scenarios. *Bioengineering* **2025**, *12*, 1062. [CrossRef] [PubMed]
4. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI)*; Springer: Cham, Switzerland, 2015; Volume 9351, pp. 234–241. [CrossRef]
5. Yadav, N.K.; Rizk, R.; Chen, W.C.; Santosh, K.C. I Detect What I Don't Know: Incremental Anomaly Learning with SWAG. In *Intelligent Human Computer Interaction*; Springer: Cham, Switzerland, 2026; Volume 16437, pp. 129–144. [CrossRef] [PubMed]

6. Azad, R.; Arimond, R.; Aghdam, E.K.; Kazerouni, A.; Merhof, D. Dae-former: Dual attention-guided efficient transformer for medical image segmentation. In *Proceedings of the International Workshop on Predictive Intelligence in Medicine*; Springer: Cham, Switzerland, 2023; pp. 83–95.
7. Hatamizadeh, A.; Nath, V.; Tang, Y.; Yang, D.; Roth, H.R.; Xu, D. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *Proceedings of the International MICCAI Brainlesion Workshop*; Springer: Cham, Switzerland, 2021; pp. 272–284.
8. Yadav, N.K.; Singh, S.K.; Dubey, S.R. TranscoderGAN: Transformer-Inception Based Encoder-Decoder Generative Adversarial Network for Thermal-Visible Face Transformation. In *Proceedings of the International Conference on Computer Vision and Image Processing*; Springer: Cham, Switzerland, 2024; pp. 1–11.
9. Hatamizadeh, A.; Tang, Y.; Nath, V.; Yang, D.; Myronenko, A.; Landman, B.; Roth, H.R.; Xu, D. Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; IEEE: New York, NY, USA, 2022; pp. 574–584.
10. Ma, J.; Li, F.; Wang, B. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv* **2024**, arXiv:2401.04722.
11. Roy, S.; Koehler, G.; Ulrich, C.; Baumgartner, M.; Petersen, J.; Isensee, F.; Jaeger, P.F.; Maier-Hein, K. MedNeXt: Transformer-driven Scaling of ConvNets for Medical Image Segmentation. *arXiv* **2023**, arXiv:2303.09975.
12. Yadav, N.K.; Rizk, R.; Chen, W.C.W.; Santosh, K.C. DynaFormer: A Dynamic Dual-Attention Transformer with Linear-Complexity Fusion for Medical Image Segmentation. In *Proceedings of the 2026 IEEE Conference on Artificial Intelligence (CAI)*; IEEE: New York, NY, USA, 2026; pp. 1518–1523. [CrossRef]
13. Zhou, Z.; Siddiquee, M.M.R.; Tajbakhsh, N.; Liang, J. UNet++: A Nested U-Net Architecture for Medical Image Segmentation. In *Proceedings of the Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Cham, Switzerland, 2018; pp. 3–11.
14. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable Convolutional Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*; IEEE: New York, NY, USA, 2017; pp. 764–773.
15. Rahman, M.M.; Munir, M.; Marculescu, R. Emcad: Efficient multi-scale convolutional attention decoding for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2024; pp. 11769–11779.
16. Wang, H.; Cao, P.; Wang, J.; Zaiane, O.R. UCTransNet: Rethinking the Skip Connections in U-Net from a Channel-Wise Perspective with Transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*; Association for the Advancement of Artificial Intelligence (AAAI): Palo Alto, CA, USA, 2022; Volume 36, pp. 2441–2449. [CrossRef]
17. Luo, Z.; Zhu, X.; Zhang, L.; Sun, B. Rethinking U-Net: Task-Adaptive Mixture of Skip Connections for Enhanced Medical Image Segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*; Association for the Advancement of Artificial Intelligence (AAAI): Washington, DC, USA, 2025; Volume 39, pp. 5874–5882.
18. Yang, L.; Zhang, R.Y.; Li, L.; Xie, X. SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In *Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021*; Meila, M., Zhang, T., Eds.; Proceedings of Machine Learning Research, PMLR: Cambridge, MA, USA, 2021; Volume 139, pp. 11863–11874.
19. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2018; pp. 3–19.
20. Shaker, A.M.; Maaz, M.; Rasheed, H.; Khan, S.; Yang, M.H.; Khan, F.S. UNETR++: Delving into Efficient and Accurate 3D Medical Image Segmentation. *IEEE Trans. Med. Imaging* **2024**, *43*, 3377–3390. [CrossRef] [PubMed]
21. Cardoso, M.J.; Li, W.; Brown, R.; Ma, N.; Kerfoot, E.; Wang, Y.; Murrey, B.; Myronenko, A.; Zhao, C.; Yang, D.; et al. MONAI: An Open-Source Framework for Deep Learning in Healthcare. *arXiv* **2022**, arXiv:2211.02701. [CrossRef]
22. Landman, B.; Xu, Z.; Iglesias, J.E.; Styner, M.; Langerak, T.R.; Klein, A. (Eds.) *MICCAI Multi-Atlas Labeling Beyond the Cranial Vault—Workshop and Challenge*; Synapse: Munich, Germany, 2015. Available online: https://www.synapse.org/Synapse:syn3193805/wiki/ (accessed on 31 August 2026).
23. Bernard, O.; Lalande, A.; Zotti, C.; Cervenansky, F.; Yang, X.; Heng, P.A.; Cetin, I.; Lekadir, K.; Camara, O.; Ballester, M.A.G.; et al. Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Trans. Med. Imaging* **2018**, *37*, 2514–2525. [CrossRef] [PubMed]
24. Urschler, M.; Rechberger, E.; Thaler, F.; Štern, D. Cascaded Self-Supervision to Advance Cardiac MRI Segmentation in Low-Data Regimes. *Bioengineering* **2025**, *12*, 872. [CrossRef] [PubMed]
25. Chen, J.; Mei, J.; Li, X.; Lu, Y.; Yu, Q.; Wei, Q.; Luo, X.; Xie, Y.; Adeli, E.; Wang, Y.; et al. TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers. *Med. Image Anal.* **2024**, *97*, 103280. [CrossRef] [PubMed]
26. Cao, H.; Wang, Y.; Chen, J.; Jiang, D.; Zhang, X.; Tian, Q.; Wang, L. Swin-Unet: Unet-like Pure Transformer for Medical Image Segmentation. *arXiv* **2021**, arXiv:2105.05537.

27. Huang, X.; Deng, Z.; Li, D.; Yuan, X.; Fu, Y. Missformer: An effective transformer for 2d medical image segmentation. *IEEE Trans. Med. Imaging* **2022**, *42*, 1484–1494. [CrossRef] [PubMed]
28. Rahman, M.M.; Marculescu, R. Multi-scale hierarchical vision transformer with cascaded attention decoding for medical image segmentation. In *Medical Imaging with Deep Learning*; PMLR: Cambridge, MA, USA, 2024; Volume 227, pp. 1526–1544.
29. Liu, J.; Yang, H.; Zhou, H.Y.; Xi, Y.; Yu, L.; Li, C.; Liang, Y.; Shi, G.; Yu, Y.; Zhang, S.; et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Cham, Switzerland, 2024; pp. 615–625.
30. Liu, J.; Li, K.; Huang, C.; Dong, H.; Song, Y.; Li, R. MixFormer: A Mixed CNN–Transformer Backbone for Medical Image Segmentation. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 5001220. [CrossRef]
31. Xie, Y.; Zhang, J.; Shen, C. CoTr: Efficiently Bridging CNN and Transformer for 3D Medical Image Segmentation. *arXiv* **2021**, arXiv:2103.03024.
32. Isensee, F.; Jaeger, P.F.; Kohl, S.A.; Petersen, J.; Maier-Hein, K.H. nnU-Net: A Self-Configuring Method for Deep Learning-Based Biomedical Image Segmentation. *Nat. Methods* **2021**, *18*, 203–211. [CrossRef] [PubMed]
33. Zhou, H.Y.; Guo, J.; Zhang, Y.; Han, X.; Yu, L.; Wang, L.; Yu, Y. nnFormer: Volumetric medical image segmentation via a 3D transformer. *IEEE Trans. Image Process.* **2023**, *32*, 4036–4045. [CrossRef] [PubMed]
34. Zhang, W.; Zhang, Y.; Zhang, L. Multiplanar Data Augmentation and Lightweight Skip Connection Design for Deep-Learning-Based Abdominal CT Image Segmentation. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 2532111. [CrossRef]
35. Huang, D.X.; Zhou, X.H.; Gui, M.J.; Xie, X.L.; Liu, S.Q.; Wang, S.Y.; Feng, Z.Q.; Hou, Z.G. MOSformer: Momentum encoder-based inter-slice fusion transformer for medical image segmentation. *arXiv* **2025**. [CrossRef]
36. Seghier, M.L. Image Segmentation Evaluation With the Dice Index: Methodological Issues. *Int. J. Imaging Syst. Technol.* **2024**, *34*, e23203. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.