UNETR++: Delving Into Efficient and Accurate 3D Medical Image Segmentation

Abdelrahman Shaker, Muhammad Maaz, Hanoona Rasheed, Salman Khan, *Senior Member, IEEE*, Ming-Hsuan Yang, *Fellow, IEEE*, and Fahad Shahbaz Khan, *Senior Member, IEEE*

***Abstract*— Owing to the success of transformer models, recent works study their applicability in 3D medical segmentation tasks. Within the transformer models, the self-attention mechanism is one of the main building blocks that strives to capture long-range dependencies, compared to the local convolutional-based design. However, the self-attention operation has quadratic complexity which proves to be a computational bottleneck, especially in volumetric medical imaging, where the inputs are 3D with numerous slices. In this paper, we propose a 3D medical image segmentation approach, named UNETR++, that offers both high-quality segmentation masks as well as efficiency in terms of parameters, compute cost, and inference speed. The core of our design is the introduction of a novel efficient paired attention (EPA) block that efficiently learns spatial and channel-wise discriminative features using a pair of inter-dependent branches based on spatial and channel attention. Our spatial attention formulation is efficient and has linear complexity with respect to the input. To enable communication between spatial and channel-focused branches, we share the weights of query and key mapping functions that provide a complimentary benefit (paired attention), while also reducing the complexity. Our extensive evaluations on five benchmarks, Synapse, BTCV, ACDC, BraTS, and Decathlon-Lung, reveal the effectiveness of our contributions in terms of both efficiency and accuracy. On Synapse, our UNETR++ sets a new state-of-the-art with a Dice Score of 87.2%, while significantly reducing parameters and FLOPs by over 71%, compared to the best method in the literature. Our code and models are available at: https://tinyurl.com/2p87x5xn.**

***Index Terms*— Deep learning, efficient attention, hybrid architecture, medical image segmentation.**

I. Introduction

VOLUMETRIC (3D) segmentation is a fundamental problem in medical imaging with numerous applications including, tumor identification and organ localization for diagnostic purposes [1], [2]. The task is typically addressed by utilizing a U-Net [3] like encoder-decoder architecture where the encoder generates a hierarchical low-dimensional representation of a 3D image and the decoder maps this learned representation to a voxel-wise segmentation. Earlier CNN-based methods use convolutions and deconvolutions in the encoder and the decoder, respectively, but struggle to achieve accurate segmentation results. A recent approach [4] aims to address the challenges of the CNN-based methods by expanding the CNN's receptive field to enhance the ability of their method to model better contextual representations. Despite the progress in encoding improved contextual representations, the limited receptive fields, local connections, and stationary weights remain challenging for CNN-based methods. These challenges may impact the effectiveness of capturing extensive global dependencies. Additionally, while 3D CNN-based methods are efficient in terms of parameters through weight sharing, they trade off with increased FLOPs and slower inference speed due to the higher number of operations. In contrast, transformer-based methods are inherently global and have recently demonstrated competitive performance at the cost of increased model complexity. Specifically, these methods rely on global self-attention, which has a quadratic complexity with respect to the input. Moreover, the complexity becomes more problematic in volumetric medical image segmentation tasks where the input comprises 3D volumes.

To address these challenges, there is a growing interest in exploring hybrid architectures that integrate the strengths of both CNNs and transformers. Recently, several works [1], [5], and [6] have explored designing hybrid architectures to combine the merits of both local convolutions and global attention. While some approaches [1] use transformer-based encoder with convolutional decoder, others [5] and [6] aim at designing hybrid blocks for both encoder and decoder subnetworks. However, these works mainly focus on increasing the segmentation accuracy which in turn substantially increases the model sizes in terms of both parameters and FLOPs, leading to unsatisfactory robustness. We argue that this unsatisfactory robustness is likely due to their inefficient self-attention design, which becomes even more problematic in volumetric medical image segmentation tasks. Further, these existing approaches do not capture the explicit dependency between spatial and channel features which can improve the segmentation quality. In this work, we aim to simultaneously improve both the segmentation accuracy and the model efficiency in a single unified framework.

Manuscript received 22 January 2024; revised 14 March 2024; accepted 4 May 2024. Date of publication 9 May 2024; date of current version 3 September 2024. *(Corresponding author: Abdelrahman Shaker.)*

Abdelrahman Shaker, Muhammad Maaz, Hanoona Rasheed, and Salman Khan are with the Computer Vision Department, Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates (e-mail: abdelrahman.youssief@mbzuai.ac.ae; muhammad.maaz@mbzuai.ac.ae; hanoona.bangalath@mbzuai.ac.ae; salman.khan@mbzuai.ac.ae).

Ming-Hsuan Yang is with the Electrical Engineering and Computer Science Department, University of California at Merced, Merced, CA 95343 USA, also with the College of Computing, Yonsei University, Seoul 03722, South Korea, and also with Google, Mountain View, CA 95344 USA (e-mail: mhyang@ucmerced.edu).

Fahad Shahbaz Khan is with Mohamed bin Zayed University, Abu Dhabi, United Arab Emirates, and also with the Electrical Engineering Department, Linköping University, 581 83 Linköping, Sweden (e-mail: fahad.khan@mbzuai.ac.ae).

Digital Object Identifier 10.1109/TMI.2024.3398728

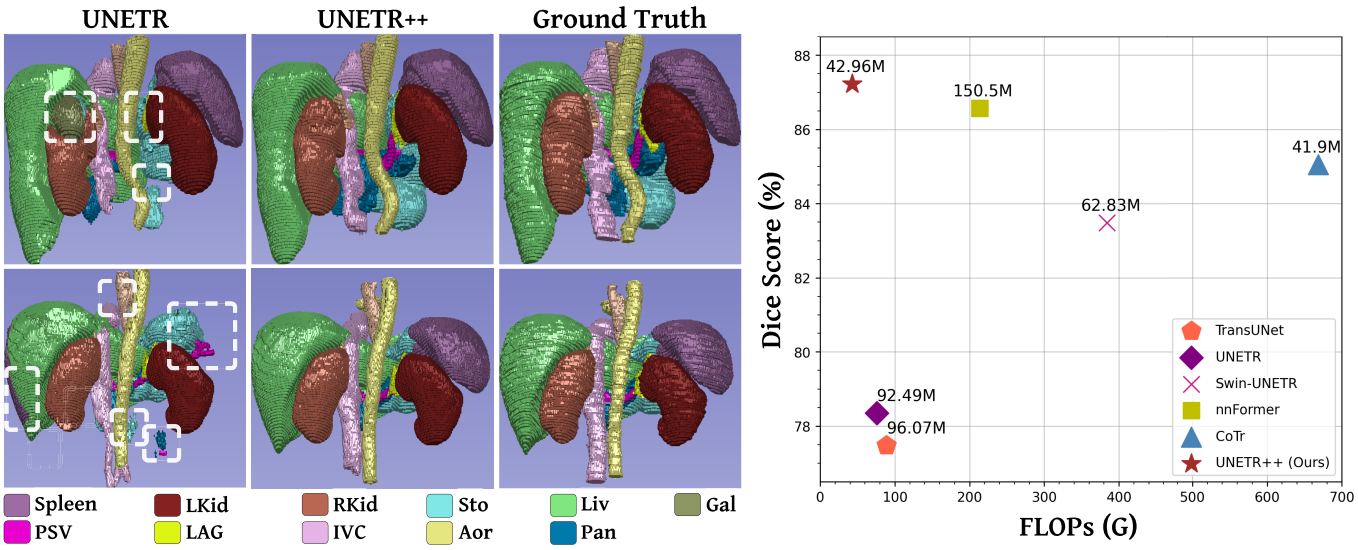

Fig. 1. **Left: Qualitative comparison between the baseline UNETR [1] and our UNETR++ on Synapse.** We present two examples containing multiple organs. Each inaccurate segmented region is marked with a white dashed box. In the first row, UNETR struggles to accurately segment the *right kidney* (RKid) and confuses it with *gallbladder* (Gal). Further, both the *stomach* (Sto) and *left adrenal gland* (LAG) tissues are inaccurately segmented. In the second row, UNETR struggles to segment the whole *spleen* and mixes it with *stomach* (Sto) and *portal and splenic veins* (PSV). Moreover, it under and over-segments certain organs (e.g., PSV and Sto). In comparison, our UNETR++ which efficiently encodes enriched inter-dependent spatial and channel features within the proposed EPA block, accurately segments all organs in these examples. Best viewed zoomed in. Additional qualitative comparisons are presented in Fig. 3 and Fig. 4. **Right: Accuracy (Dice score) vs. model complexity (FLOPs and parameters) comparison on Synapse.** Compared to nnFormer [5], UNETR++ achieves better segmentation performance while significantly reducing the model complexity by over 71%.

A. Contributions

We propose an efficient hybrid hierarchical architecture for 3D medical image segmentation, named UNETR++, that strives to achieve both better segmentation accuracy and efficiency in terms of parameters, FLOPs, GPU memory consumption, and inference speed. Built on the recent UNETR framework [1], our proposed UNETR++ hierarchical approach introduces a novel *efficient paired attention* (EPA) block that efficiently captures enriched inter-dependent spatial and channel features by applying both spatial and channel attention in two branches. Our spatial attention in EPA projects the keys and values to a fixed lower dimensional space, making the self-attention computation linear with respect to the number of input tokens. On the other hand, our channel attention emphasizes the dependencies between the channel feature maps by performing the dot-product operation between queries and keys in the channel dimension. Further, to capture a strong correlation between the spatial and channel features, the weights for queries and keys are shared across the branches which also aids in controlling the number of network parameters. In contrast, the weights for values are kept independent to enforce learning complementary features in both branches.

UNETR++ offers a substantial advantage over existing methods by significantly reducing GPU consumption with faster inference speed and fewer FLOPs, which is crucial for 3D segmentation tasks due to their complexities. For example, UNETR++ outperforms nnFormer [5] with a 2.4× faster GPU inference speed, utilizing 5× less GPU memory and 6× fewer FLOPs. Compared to Swin-UNETR [6], UNETR++ achieves a 3.6× faster inference speed while consuming 8× less GPU memory and FLOPs. Additionally, UNETR++ exhibits a 35% reduction in memory consumption compared to the highly efficient CNN-based method nnUNet [2], with 6× fewer FLOPs. We prioritize efficient resource utilization, resulting in faster inference speed on CPU and GPU. Furthermore, we argue that these aforementioned hybrid approaches struggle to effectively capture the inter-dependencies between feature channels to obtain an enriched feature representation that encodes both the spatial information as well as the inter-channel feature dependencies. In this work, we set out to collectively address the above issues in a unified hybrid segmentation framework.

We perform a comprehensive evaluation to assess the performance of UNETR++ across five widely used benchmarks, including Synapse [7], BTCV [7], ACDC [8], BraTS [9], and Decathlon-Lung [10]. Our findings reveal the effectiveness of our contributions in terms of both efficiency and accuracy, illustrating the remarkable generalization ability of UNETR++ across diverse datasets with different modalities, including CT scans and MRI. On Synapse, our UNETR++ achieves a Dice Score of 87.2%, while significantly reducing parameters and FLOPs by over 71%, compared to the best-performing method [5]. On ACDC, UNETR++ achieves an average Dice score of 93.83%, which is better than the recently introduced MexNeXt [4] by 1.4%. On BraTS and Decathalon-Lung Segmentation, UNETR++ outperforms the current SOTA in three evaluation metrics, demonstrating better segmentation performance while operating at a faster inference speed as well as requiring significantly less GPU memory.

II. Related Work

A. CNN-Based Segmentation Methods

Since the introduction of the U-Net design [3], several CNN-based approaches [11], [12], [13], [14] have extended the standard U-Net architecture for various medical image segmentation tasks. In the case of 3D medical image segmentation [15], [16], [17], [18], [19], the full volumetric image is typically processed as a sequence of 2D slices. Several works have explored hierarchical frameworks to capture contextual information. Milletari et al. [18] propose to use 3D representations of the volumetric image by down-sampling the volume to lower resolutions for preserving the beneficial image features. Çiçek et al. [17] extend the U-Net architecture to volumetric segmentation by replacing the 2D operations with their 3D counterparts, learning from sparsely annotated volumetric images. Isensee et al. [2] introduce a generalized segmentation framework, named nnUNet, that automatically configures the architecture to extract features at multiple scales. Roth et al. [20] propose a multi-scale 3D fully convolution network to learn representations from varying resolutions for multi-organ segmentation. Further, several efforts in the literature have been made to encode holistic contextual information within CNN-based frameworks using, e.g.,image pyramids [21], large kernels [22], dilated convolution [23], and deformable convolution [24]. Recently, Roy et al. [4] introduced a fully convolutional 3D Encoder-Decoder network named MedNeXt. This architecture, which is an extension of the ConNeXt framework [25], incorporates adaptive kernel sizes with residual connections to enhance segmentation accuracy in the context of 3D medical imaging with limited data. Although MedNeXt-M-K3 [4] demonstrates promising accuracy, it comes at the cost of a 4.1× increase in GPU consumption compared to nnUNet [2], resulting in significantly slower GPU and CPU inference speed.

B. Transformers-Based Segmentation Methods

Vision transformers (ViTs) have recently gained popularity thanks to their ability to encode long-range dependencies leading to promising results on various vision tasks, including classification [26] and detection [27]. One of the main building blocks within the transformer's architecture is the self-attention operation that models the interactions among the sequence of image patches, thereby learning global relationships. Few recent works have explored alleviating the complexity issue of standard self-attention operation within transformer frameworks [28], [29], [30], [31]. However, most of these recent works mainly focus on the classification problem and have not been studied for dense prediction tasks.

In the context of medical image segmentation, few recent works [32], [33] have investigated pure transformers designs. Karimi et al. [32] propose to divide a volumetric image into 3D patches which are then flattened to construct a 1D embedding and passed to a backbone for global representations. Cao et al. [33] introduce an architecture with shifted windows for 2D medical image segmentation. Here, an image is divided into patches and fed into a U-shaped encoder-decoder for local-global representation learning.

C. Hybrid Segmentation Methods

Other than pure CNN or transformers-based designs, several recent works [1], [5], [19], [34], [35], [36] have explored hybrid architectures to combine convolution and self-attention operations for better segmentation. TransFuse [34] proposes a parallel CNN-transformer architecture with a BiFusion module to fuse multi-level features in the encoder. MedT [19] introduces a gated position-sensitive axial-attention mechanism in self-attention to control the positional embedding information in the encoder, while the ConvNet module in the decoder produces a segmentation model. TransUNet [35] combines transformers and the U-Net architecture, where transformers encode the embedded image patches from convolution features and the decoder combines the upsampled encoded features with high-resolution CNN features for localization. Ds-transunet [36] utilizes a dual-scale encoder based on Swin transformer [37] to handle multi-scale inputs and encode local and global feature representations from different semantic scales through self-attention.

Hatamizadeh et al. [1] introduce a 3D hybrid model, UNETR, that combines the long-range spatial dependencies of transformers with the CNN's inductive bias into a "U-shaped" encoder-decoder architecture. The transformer blocks in UNETR are mainly used in the encoder to extract fixed global representations and then are merged at multiple resolutions with a CNN-based decoder. Zhou et al. [5] introduce an approach, named nnFormer, that adapts the Swin-UNet [33] architecture. Here, convolution layers transform the input scans into 3D patches and volume-based self-attention modules are introduced to build hierarchical feature pyramids. While achieving promising performance, the computational complexity of nnFormer is significantly higher compared to UNETR and other hybrid methods. CoTr [38] is a hybrid architecture that consists of CNN encoder and efficient deformable transformers. The convolutional encoder extracts the local feature maps, and the deformable transformer is employed to attend only to a few key positions and encode the partial dependencies on the extracted feature representations.

As discussed above, most recent hybrid approaches, such as UNETR [1] and nnFormer [5], achieve improved segmentation performance compared to their pure CNNs and transformers-based counterparts. However, we note that this pursuit of increasing the segmentation accuracy by these hybrid approaches comes at the cost of substantially larger models (in terms of parameters and FLOPs), which can further lead to unsatisfactory robustness. For instance, UNETR achieves favorable accuracy but comprises 2.5× more parameters, compared to the best existing CNN-based nnUNet [2]. Moreover, nnFormer obtains improved performance over UNETR but further increases the parameters by 1.6× and FLOPs by 2.8×.

D. Efficient Attention Methods

Designing efficient attention blocks for 2D vision applications has received much attention in recent years. CBAM [39] is an efficient attention module based on convolutional neural networks. This module processes the feature maps along the

spatial and channel dimensions efficiently. Then, the resulting attention maps are multiplied with the input feature map, facilitating adaptive feature refinement. Although the proposed EPA block encodes channel and spatial information, there are major differences between the EPA block and CBAM: (1) CBAM encodes the spatial and channel representations using two sequential sub-modules. In contrast, the EPA block encodes them in a parallel way. (2) The formulation of the channel and spatial attentions in the EPA block is based on the attention mechanism, while CBAM is based on pooling and convolution operations. (3) The channel and spatial attentions have separate weight matrices in CBAM. In the EPA, we share the weights of QK matrices and claim that this sharing mechanism reduces the parameters by 25% and improves the performance by 0.23% by learning only the complementary features.

Transformer-CBAM [40] serves as an enhancement to CBAM [39] through the integration of a multi-scale transformer module. This addition enables the modeling of context information across different scales, making it particularly effective for remote sensing image change detection. The Squeeze-and-Excitation [41] focuses on the relationships between the channel feature maps. It introduces the "Squeeze-and-Excitation" (SE) block, which readjusts channel-specific feature maps by explicitly capturing the inter-dependencies between the channels. The Attention Gated U-Net [42] expands the U-Net [3] architecture by incorporating an attention gate (AG) module designed for medical imaging. This AG module learns to prioritize target structures of different shapes and sizes through different gating mechanisms, with minimal computational overhead. While most of the methods mentioned above demonstrate a promising trade-off between efficiency and accuracy, they are primarily designed for 2D vision tasks. To validate the effectiveness of the proposed EPA in the context of 3D medical segmentation, we conduct experiments by substituting the EPA block with the 3D counterparts of these efficient methods, as detailed in Table IX.

III. Method

A. Overall Architecture

Fig. 2 presents our UNETR++ architecture, comprising a hierarchical encoder-decoder structure. We base our UNETR++ framework on the recently introduced UNETR [1] with skip connections between the encoders and decoders, followed by convolutional blocks (ConvBlocks) to generate the prediction masks. Instead of using a fixed feature resolution throughout the encoders, our UNETR++ employs a hierarchical design where the resolution of the features is gradually decreased by a factor of two in each stage. Within our UNETR++ framework, the encoder has four stages. The number of channels at the four stages is $[C_1, C_2, C_3, C_4]$. The first stage consists of patch embedding to divide volumetric input into 3D patches, followed by our novel efficient paired-attention (EPA) block. In the patch embedding, we divide each 3D input (volume) $\boldsymbol{x} \in \mathbb{R}^{H \times W \times D}$ into non-overlapping patches $\boldsymbol{x}_u \in \mathbb{R}^{N \times (P_1, P_2, P_3)}$, where (P_1, P_2, P_3) is the resolution of each patch and $N = (\frac{H}{P_1} \times \frac{W}{P_2} \times \frac{D}{P_3})$ denotes the length of the sequence. Then, the patches are projected into C channel dimensions, producing feature maps of size $\frac{H}{P_1} \times \frac{W}{P_2} \times \frac{D}{P_3} \times C$. We use the same patch resolution (4, 4, 2), as in [5]. For each of the remaining encoder stages, we employ downsampling layers using non-overlapping convolution to decrease the resolution by a factor of two, followed by the EPA block.

Within our proposed UNETR++, each EPA block comprises two attention modules to efficiently learn enriched spatial-channel feature representations by encoding the information in both spatial and channel dimensions with shared keys-queries scheme. The encoder stages are connected with the decoder stages via skip connections to merge the outputs at different resolutions. This enables the recovery of the spatial information lost during the downsampling operations, leading to predicting a more precise output. Similar to the encoder, the decoder also comprises four stages, where each decoder stage consists of an upsampling layer using deconvolution to increase the resolution of the feature maps by a factor of two, followed by the EPA block (except the last decoder). The number of channels is decreased by a factor of two between each two decoder stages. Consequently, the outputs of the last decoder are fused with convolutional features maps to recover the spatial information and enhance the feature representation. The resulting output is then fed into $3 \times 3 \times 3$ and $1 \times 1 \times 1$ convolutional blocks to generate voxel-wise final mask predictions. Next, we present in detail the EPA block.

B. Efficient Paired-Attention Block

As discussed earlier, most existing hybrid methods employ the self-attention operation having quadratic complexity with the number of tokens. This is computationally expensive in the case of volumetric segmentation and becomes more problematic when interleaving window attention and convolution components in hybrid designs. Further, the spatial attention information can be efficiently learned by projecting the spatial matrices of the keys and values into a lower-dimension space. Effectively combining the interactions in the spatial dimensions and the inter-dependencies between the channel features provides enriched contextual spatial-channel feature representations, leading to improved mask predictions.

The proposed EPA block performs efficient global attention and effectively captures enriched spatial-channel feature representations. The EPA block comprises spatial attention and channel attention modules. The spatial attention module reduces the complexity of self-attention from quadratic to linear. On the other hand, the channel attention module effectively learns the inter-dependencies between the channel feature maps. The EPA block is based on sharing keys-queries between the two attention modules to be mutually informed in order to generate better and efficient feature representation. This is likely due to learning complementary features by sharing the keys and queries but using different value layers. To enhance training stability, Layer Normalization (LayerNorm) is employed at the beginning of each EPA block.

As illustrated in Fig. 2 (right), the input feature maps $\boldsymbol{x}$ are fed into the channel and spatial attention modules of the EPA block. The weights of $\boldsymbol{Q}$ and $\boldsymbol{K}$ linear layers are shared

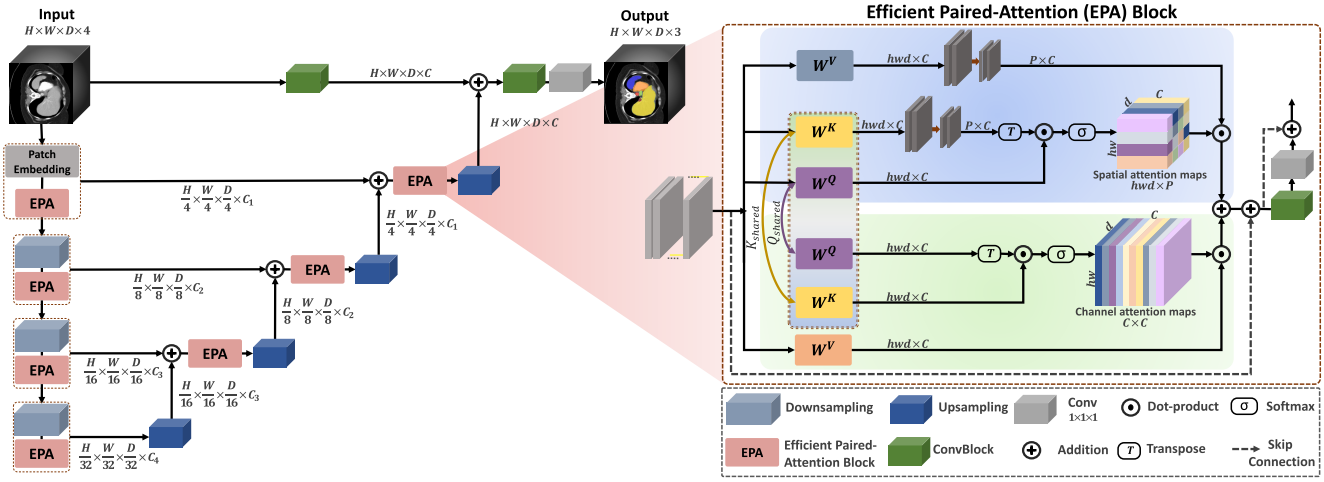

Fig. 2. **Overview of our UNETR++ approach with hierarchical encoder-decoder structure**. The 3D patches are fed to the encoder, whose outputs are then connected to the decoder via skip connections followed by convolutional blocks to produce the final segmentation mask. The focus of our design is the introduction of an *efficient paired-attention* (EPA) block (Sec. III-B). Each EPA block performs two tasks using parallel attention modules with shared keys-queries and different value layers to efficiently learn enriched spatial-channel feature representations. As illustrated in the EPA block diagram (on the right), the first (top) attention module aggregates the spatial features by a weighted sum of the projected features in a linear manner to compute the spatial attention maps, while the second (bottom) attention module emphasizes the dependencies in the channels and computes the channel attention maps. Finally, the outputs of the two attention modules are fused and passed to convolutional blocks to enhance the feature representation, leading to better segmentation masks.

across the two attention modules and different $\boldsymbol{V}$ layer is used for each attention module. The two attention modules are computed as follows:

$$\hat{\boldsymbol{X}}_s = \mathrm{SA}(\boldsymbol{Q}_{shared},\ \boldsymbol{K}_{shared},\ \boldsymbol{V}_{spatial}), \tag{1}$$

$$\hat{\boldsymbol{X}}_c = \mathrm{CA}(\boldsymbol{Q}_{shared},\ \boldsymbol{K}_{shared},\ \boldsymbol{V}_{channel}) \tag{2}$$

where, $\hat{\boldsymbol{X}}_s$, $\hat{\boldsymbol{X}}_c$, SA and SA denotes the spatial and channels attention maps, spatial and channel attention module respectively. $\boldsymbol{Q}_{shared}$, $\boldsymbol{K}_{shared}$, $\boldsymbol{V}_{spatial}$, and $\boldsymbol{V}_{channel}$ are the matrices for shared queries, shared keys, spatial value layer, and channel value layer, respectively.

1) Spatial Attention: We strive in this module to learn the spatial information efficiently by reducing the complexity from $O(n^2)$ to $O(np)$, where n is the number of tokens, and p is the dimension of the projected vector, where $p \ll n$. Given a normalized tensor $\boldsymbol{X}$ of shape $hwd \times C$, we compute $\boldsymbol{Q}_{shared}$, $\boldsymbol{K}_{shared}$, and $\boldsymbol{V}_{spatial}$ projections using three linear layers, yielding $\boldsymbol{Q}_{shared}=\boldsymbol{W}^{Q}\boldsymbol{X}$, $\boldsymbol{K}_{shared}=\boldsymbol{W}^{K}\boldsymbol{X}$, and $\boldsymbol{V}_{spatial}=\boldsymbol{W}^{V}\boldsymbol{X}$, with dimensions $hwd \times C$, where $\boldsymbol{W}^{Q}$,$\boldsymbol{W}^{K}$, and $\boldsymbol{W}^{V}$ are the projection weights for $\boldsymbol{Q}_{shared}$, $\boldsymbol{K}_{shared}$, and $\boldsymbol{V}_{spatial}$, respectively. Then, we perform three steps. First, the $\boldsymbol{K}_{shared}$ and $\boldsymbol{V}_{spatial}$ layers are projected from $hwd \times C$ into lower-dimensional matrices of shape $p \times C$. Second, the spatial attention maps are computed by multiplying the $\boldsymbol{Q}_{shared}$ layer by the transpose of the projected $\boldsymbol{K}_{shared}$, followed by softmax to measure the similarity between each feature and the rest of the spatial features. Third, these similarities are multiplied by the projected $\boldsymbol{V}_{spatial}$ layer to produce the final spatial attention maps of shape $hwd \times C$. The spatial attention is defined as follows:

$$\hat{\boldsymbol{X}}_s = \mathrm{Softmax}\left(\frac{\boldsymbol{Q}_{shared}\boldsymbol{K}^{\top}_{proj}}{\sqrt{d}}\right) \cdot \tilde{\boldsymbol{V}}_{spatial} \tag{3}$$

where, $\boldsymbol{Q}_{shared}$, $\boldsymbol{K}_{proj}$, $\tilde{\boldsymbol{V}}_{spatial}$ denote shared queries, projected shared keys, and projected spatial value layer, respectively, and d is the size of each vector.

2) Channel Attention: This module captures the inter-dependencies between feature channels by applying the dot-product operation in the channel dimension between the channel value layer and channel attention maps. Using the same $\boldsymbol{Q}_{shared}$ and $\boldsymbol{K}_{shared}$ of the spatial attention module, we compute the value layer for the channels to learn the complementary features using the linear layer, yielding $\boldsymbol{V}_{channel} = \boldsymbol{W}^{V}\boldsymbol{X}$, with dimensions $hwd \times C$, where $\boldsymbol{W}^{V}$ is the projection weight for $\boldsymbol{V}_{channel}$. The channel attention is defined as follows:

$$\hat{\boldsymbol{X}}_c = \boldsymbol{V}_{channel} \cdot \mathrm{Softmax}\left(\frac{\boldsymbol{Q}^{\top}_{shared}\boldsymbol{K}_{shared}}{\sqrt{d}}\right) \tag{4}$$

where, $\boldsymbol{V}_{channel}$, $\boldsymbol{Q}_{shared}$, $\boldsymbol{K}_{shared}$ denote channel value layer, shared queries, and shared keys, respectively.

Finally, we perform sum fusion and transform the outputs from the two attention modules by convolution blocks to obtain enriched feature representations. The final output $\hat{\boldsymbol{X}}$ of the EPA block is obtained as follows:

$$\hat{\boldsymbol{X}} = \mathrm{Conv}_1(\mathrm{Conv}_3(\hat{\boldsymbol{X}}_s + \hat{\boldsymbol{X}}_c)) \tag{5}$$

where, $\hat{\boldsymbol{X}}_s$ and $\hat{\boldsymbol{X}}_c$ denotes the spatial and channels attention maps, and Conv_1 and Conv_3 are $1 \times 1 \times 1$ and $3 \times 3 \times 3$ convolution blocks, respectively.

C. Loss Function

Following the baseline UNETR [1] and nnFormer [5], our loss function is based on a summation of the commonly used soft dice loss [18] and cross-entropy loss to simultaneously leverage the benefits of both complementary loss functions.

It is defined as follows:

$$\mathcal{L}(Y, P) = 1 - \sum_{i=1}^{I} \left(\frac{2 * \sum_{v=1}^{V} Y_{v,i} \cdot P_{v,i}}{\sum_{v=1}^{V} Y_{v,i}^2 + \sum_{v=1}^{V} P_{v,i}^2} + \sum_{v=1}^{V} Y_{v,i} \log P_{v,i} \right) \quad (6)$$

where, I denotes the number of classes, V denotes the number of voxels, $Y_{v,i}$ and $P_{v,i}$ denote the ground truths and output probabilities at voxel v for class i, respectively.

IV. Experiments

A. Experimental Setup

We carry out experiments on five datasets: Synapse and BTCV for Multi-organ CT Segmentation [7], ACDC for Automated Cardiac Diagnosis [8], Brain Tumor Segmentation (BraTS) [9] and the Medical Segmentation Decathlon-Lung [10].

1) Datasets: The *Synapse* [7] dataset consists of abdominal CT scans of 30 subjects with 8 organs. Consistent with previous approaches, we follow the splits used in [35] and train our model on 18 samples, and evaluate the remaining 12 cases. We report the model performance using four evaluation metrics on 8 abdominal organs: *spleen*, *right kidney*, *left kidney*, *gallbladder*, *liver*, *stomach*, *aorta* and *pancreas*. The *BTCV* [7] dataset contains 30 subjects for training and 20 subjects for testing with abdominal CT scans. It consists of 13 organs, including 8 organs of Synapse, along with *esophagus*, *inferior vena cava*, *portal and splenic veins*, *right and left adrenal gland*. We report the DSC on all 13 abdominal organs and the results are obtained from the BTCV leaderboard. The *ACDC* [8] dataset comprises cardiac MRI images of 100 patients, with segmentation annotations of *right ventricle* (RV), *left ventricle* (LV) and *myocardium* (MYO). Consistent with [5], we split the data into 70, 10, and 20 train, validation, and test samples. We report the DSC on the three classes. The *BraTS* [9] comprises 484 MRI images, where each image consists of four channels, FLAIR, T1w, T1gd, and T2w. We split the dataset into 80:5:15 ratios for training, validation, and testing. The target categories are the whole tumor (WT), enhancing tumor (ET), and tumor core (TC). The *Decathlon-lung* [10] dataset comprises 63 CT volumes for a two-class problem with the goal to segment lung cancer from the background. We split the data into 80:20 ratios for training and testing. For each dataset, in case the official results for certain methods are not provided, we ensure a fair comparison by training those methods using the same data division and setting. Subsequently, We report the model performance using four evaluation on the testing set for all methods.

2) Evaluation Metrics: Following the methods in the literature [1], [2], [4], [5], [43], we evaluate the performance of all models using the Dice Similarity Coefficient (DSC) and 95% Hausdorff Distance (HD95). To capture a comprehensive performance of all segmentation methods, we further evaluate the performance of all models based on Normalized Surface Dice (NSD) and Mean Average Surface Distance (MASD) based on the recommendations of [44] and [45]. Our comprehensive evaluation using four different metrics provides a thorough assessment of segmentation performance and emphasizes the effectiveness of our proposed method.

DSC measures the overlap between the volumetric segmentation predictions and the voxels of the ground truths, it is defined as follows:

$$DSC(Y, P) = 2 * \frac{|Y \cap P|}{|Y| \cup |P|} = 2 * \frac{Y \cdot P}{Y^2 + P^2} \quad (7)$$

where, Y and P denote the ground truths and output probabilities for all voxels, respectively.

HD95 is commonly used as boundary-based metric to measure the 95^{th} percentile of the distances between boundaries of the volumetric segmentation predictions and the voxels of the ground truths. It is defined as follows:

$$HD_{95}(Y, P) = \max\{\boldsymbol{d}_{95}(Y, P), \boldsymbol{d}_{95}(P, Y)\} \quad (8)$$

where, $\boldsymbol{d}_{95}(Y, P)$ is the maximum 95^{th} percentile distance between the ground truth and predicted voxels, and $\boldsymbol{d}_{95}(P, Y)$ is the maximum 95^{th} percentile distance between the predicted and ground truth voxels.

NSD measures the overlap between the predicted and ground truth boundaries in the segmentation voxels, providing a normalized assessment of the alignment between the surfaces. It considers both the boundary and border regions to provide a comprehensive measure of how well the predicted surface aligns with the ground truth. It is defined as follows:

$$NSD(Y, P) = \frac{|S_Y \cap B_P| + |S_P \cap B_Y|}{|S_Y| + |S_P|} \quad (9)$$

where, S_Y and S_P denote the boundaries of the ground truths and output probabilities for all voxels, and B_Y and B_P denote the border regions for the ground truths and output probabilities for all voxels respectively.

MASD measures the average distance between predicted and ground truth voxels and computes the mean over those averages, showing how, on average, the predicted voxels deviate from their ground truth counterparts. This metric assesses the spatial dissimilarity by calculating the average distance between corresponding points on the predicted and ground truth surfaces. It is defined as follows:

$$MASD(Y, P) = \frac{1}{2} \left(\frac{\sum_{y \in Y} \boldsymbol{d}(y, P)}{|Y|} + \frac{\sum_{p \in P} \boldsymbol{d}(Y, p)}{|P|} \right) \quad (10)$$

where, $\boldsymbol{d}(y, P)$ denotes the distance between a point y of the ground truth and the predicted voxels P, and $\boldsymbol{d}(Y, p)$ denotes the distance between the ground truth voxels Y and a predicted point p.

3) Statistical Significance: To show the statistical significance for UNETR++, we use the two-sample t-test to compute the p-values between the average performance of UNETR++ and the best-performing method for each dataset in terms of DSC, NSD, HD95, and MASD. The null hypothesis is defined as UNETR++ has no advantage over any best-performing method. Notably, UNETR++ yields small p-values, consistently less than 1e-2 or 5e-2 across the evaluation metrics. This indicates strong evidence against the null

TABLE I
BASELINE COMPARISON ON SYNAPSE. WE SHOW THE RESULTS IN TERMS OF SEGMENTATION PERFORMANCE (DSC) AND MODEL COMPLEXITY (PARAMETERS AND FLOPS). FOR A FAIR COMPARISON, ALL RESULTS ARE OBTAINED USING THE SAME INPUT SIZE AND PRE-PROCESSING. EACH ROW BUILDS ON THE PREVIOUS ONE, REFLECTING A SEQUENTIAL PROGRESSION OF EXPERIMENTS

Model	Params (M)	FLOPs (G)	DSC (%)
UNETR (Baseline)	92.49	75.76	78.35
+ EPA in Encoder w/o QK sharing	28.94	39.36	85.17
++ EPA in Decoder w/o QK sharing	57.31	47.98	86.99
+++ QK sharing of Encoder & Decoder	42.96	47.98	87.22

hypothesis, supporting the conclusion that UNETR++ has significant improvements over previous methods in various benchmarks across the four evaluation metrics.

4) Implementation Details: We implement our approach in Pytorch v1.10.1 and using the MONAI libraries [46]. For a fair comparison with all methods, we use the same input size, pre-processing strategy, training loss, and no additional training data for the five datasets for all methods. The models are trained using a single A100 40GB GPU for 1k epochs with learning rate of 0.01 and weight decay of $3e^{-5}$. We employ a sliding window with 50% overlap for inference and report the Dice score of a single model without using ensemble techniques. For the Synapse dataset, all the models are trained with inputs of size $128 \times 128 \times 64$. For BTCV, we follow the same training recipe as in [1] and train all models at $96 \times 96 \times 96$ resolution. For ACDC, BraTS, and Decathlon-Lungs, we train all models at a resolution of $160 \times 160 \times 16$, $128 \times 128 \times 128$, and $192 \times 192 \times 32$, respectively. For ACDC, we use patch embeddings with a resolution of (4, 4, 1) instead of (4, 4, 2) to accommodate the number of slices. All other training hyper-parameters and data augmentation are the same as in nnFormer [5].

B. Baseline Comparison

Table I shows the impact of integrating the proposed contributions within the baseline UNETR [1] on Synapse. In addition to the Dice Similarity Coefficient (DSC), we report the model complexity in terms of parameters and FLOPs. In all cases, we report performance in terms of single-model accuracy. As discussed earlier, UNETR++ is a hierarchical architecture that downsamples the feature maps of the encoder by a factor of two after each stage. Hence, the model comprises four encoder stages and four decoder stages. This hierarchical design of our UNETR++ enables a significant reduction in model complexity by reducing the parameters from 92.49M to 16.60M and FLOPs from 75.76G to 30.75G while maintaining a comparable DSC of 78.29%, compared to the baseline. Introducing the EPA block within our UNETR++ encoders leads to a significant improvement in performance with an absolute gain of 6.82% in DSC over the baseline. The performance is further improved by integrating the EPA block in the decoder without QK sharing from 85.17% to 86.99%. To optimize the model's efficiency in terms of both the number of parameters and representation learning, we share the QK layers between spatial and channel attention mechanisms. This approach not only reduces the number of parameters by 25%, but also leads to a performance boost of 0.23% by enabling the model to learn more effective complementary features. Our final UNETR++ having a hierarchical design with the novel EPA block both in encoders and decoders leads to a significant improvement of 8.87% in DSC, while considerably reducing the model complexity by 54% in parameters and 37% in FLOPs, compared to the baseline. We further conduct an experiment to evaluate our spatial and channel attention within the proposed EPA block. Employing spatial and channel attention improves the performance significantly with DSC of 86.42% and 86.39%, respectively over the baseline. Combining both spatial and channel attention within our EPA block leads to a further improvement with DSC of 87.22%.

We show in Fig. 3 a comprehensive qualitative comparison between the baseline UNETR and our UNETR++ on the multi-organ segmentation task. We enlarge different organs (marked as green dashed boxes in the first row) from several cases.

In the first column, the baseline encounters difficulties in accurately segmenting *stomach* and *pancreas*. In the second column, it under-segments *portal and splenic veins*, *the inferior vena cava*, and *liver*. In the third column, UNETR struggles to segment adjacent small organs, such as *aorta* and *the inferior vena cava*, whereas in the fourth column, it tends to over-segment *portal and splenic veins* and *liver*. In the last column, the baseline completely misses the *stomach*. In contrast, UNETR++ demonstrates improved performance by accurately segmenting all organs.

C. State-of-the-Art Comparison

1) Synapse Dataset: Table II shows the comprehensive evaluation on the multi-organ Synapse dataset. We report the segmentation performance using DSC, NSD, HD95, and MASD metrics on the abdominal organs. The segmentation performance is reported with a single model accuracy and without utilizing any pre-training, model ensemble, or additional data. The pure CNN-based U-Net [3] approach achieves a mean DSC of 76.85%. The recent CNN-based method nnUNet [2], MedNeX-M-K5 [4] achieves superior performance compared to U-Net [3]. Among existing hybrid transformer-CNN based methods, UNETR [1] and Swin-UNETR [6] achieve a mean DSC of 78.35% and 83.48%, respectively. On this dataset, nnFormer [5] obtains superior performance compared to other existing works. Our UNETR++ outperforms nnFormer by achieving a mean DSC of 87.22%. Further, UNETR++ obtains an absolute reduction in error of 3.1% 0.8% over nnFormer in terms of HD95 and MASD, respectively. Notably, UNETR++ achieves this improvement in segmentation performance by significantly reducing the model complexity by over 71% in terms of parameters and FLOPs. To validate the statistical significance of UNETR++ over the best existing baseline (nnFormer), we compute average p-values for the evaluation metrics as follows: DSC ($<$ 5e-2), NSD ($<$ 1e-2), HD95 ($<$1e-2), and MASD ($<$ 1e-2).

Fig. 4 shows a qualitative comparison of UNETR++ with the best existing approaches on abdominal multi-organ

TABLE II

STATE-OF-THE-ART COMPARISON ON THE ABDOMINAL MULTI-ORGAN SYNAPSE DATASET. WE REPORT THE DSC FOR EACH ORGAN AND THE AVERAGE SEGMENTATION PERFORMANCE OF ALL ORGANS USING FOUR EVALUATION METRICS (DSC, NSD, HD95, AND MASD). UNETR++ ACHIEVES FAVORABLE SEGMENTATION PERFORMANCE AGAINST EXISTING METHODS. THE BEST RESULTS ARE IN BOLD. ABBREVIATIONS STAND FOR: SPL: *spleen*, RKID: *right kidney*, LKID: *left kidney*, GAL: *gallbladder*, LIV: *liver*, STO: *stomach*, AOR: *aorta*, PAN: *pancreas*

Methods	Spl	RKid	LKid	Gal	Liv	Sto	Aor	Pan	Average			
									DSC ↑	NSD ↑	HD95 ↓	MASD ↓
U-Net [3]	86.67	68.60	77.77	69.72	93.43	75.58	89.07	53.98	76.85	-	39.70	-
TransUNet [35]	85.08	77.02	81.87	63.16	94.08	75.62	87.23	55.86	77.49	75.96	31.69	6.32
Swin-UNet [33]	90.66	79.61	83.28	66.53	94.29	76.60	85.47	56.58	79.13	78.65	21.55	4.83
UNETR [1]	85.00	84.52	85.60	56.30	94.57	70.46	89.80	60.47	78.35	76.58	18.59	8.81
MISSFormer [47]	91.92	82.00	85.21	68.65	94.41	80.81	86.99	65.67	81.96	-	18.20	-
TransBTS [43]	91.65	86.99	87.46	62.52	96.42	77.39	91.71	72.12	83.28	82.06	12.34	3.65
Swin-UNETR [6]	95.37	86.26	86.99	66.54	95.72	77.01	91.12	68.80	83.48	80.91	10.55	**3.29**
CoTr [38]	94.93	86.80	87.67	62.90	96.37	80.46	92.43	78.84	85.05	84.11	9.04	3.40
nnUNet [2]	91.16	86.21	86.92	69.77	96.49	85.92	91.78	83.23	86.44	83.81	10.91	3.53
MedNeXt-M-K3 [4]	90.63	86.50	87.66	73.00	**96.92**	77.89	92.25	80.81	85.71	85.25	19.10	3.65
MedNeXt-M-K5 [4]	91.16	87.51	87.67	71.31	97.01	80.46	92.48	80.20	85.97	82.79	17.59	3.89
nnFormer [5]	90.51	86.25	86.57	70.17	96.84	**86.83**	92.04	**83.35**	86.57	84.46	10.63	4.19
UNETR++	**95.77**	**87.18**	**87.54**	**71.25**	96.42	86.01	**92.52**	81.10	**87.22**	**85.99**	**7.53**	3.39

TABLE III

STATE-OF-THE-ART COMPARISON ON THE BTCV TEST SET FOR MULTI-ORGAN SEGMENTATION. ALL RESULTS ARE OBTAINED USING A SINGLE MODEL ACCURACY AND WITHOUT ANY ENSEMBLE, PRE-TRAINING, OR ADDITIONAL DATA. OUR UNETR++ ACHIEVES FAVORABLE PERFORMANCE AGAINST EXISTING 3D IMAGE SEGMENTATION METHODS. ABBREVIATIONS ARE AS FOLLOWS: SPL: *spleen*, RKID: *right kidney*, LKID: *left kidney*, GAL: *gallbladder*, ESO: *esophagus*, LIV: *liver*, STO: *stomach*, AOR: *aorta*, IVC: *the Inferior Vena cava*, PSV: *portal and Splenic veins*, PAN: *pancreas*, RAG: *right Adrenal gland*, LAG: *left Adrenal gland*. RESULTS ARE OBTAINED FROM THE BTCV LEADERBOARD

Methods	Spl	RKid	LKid	Gal	Eso	Liv	Sto	Aor	IVC	PSV	Pan	RAG	LAG	Avg
nnUNet [2]	**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
TransBTS [43]	94.55	89.20	90.97	68.38	75.61	96.44	83.52	88.55	82.48	74.21	76.02	67.23	67.03	81.31
UNETR [1]	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 [6]	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 [5]	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
UNETR++	94.94	**91.90**	93.62	**70.75**	**77.18**	95.95	85.15	**89.28**	**83.14**	76.91	77.42	72.56	68.17	**83.28**

segmentation. Here, the inaccurate segmentations are marked with red dashed boxes. In the first row, we observe that existing approaches struggle to accurately segment the *stomach* by either under-segment it in the case of UNETR and Swin UNETR or confusing it with *spleen* in the case of nnFormer. In comparison, our UNETR++ accurately segments the *stomach*. Further, existing methods fail to fully segment the *right kidney* in the second row. In contrast, our UNETR++ accurately segments the whole *right kidney*, likely due to learning the contextual information with the enriched spatial-channel representation. Moreover, UNETR++ smoothly delineates boundaries between *spleen*, *stomach*, and *liver*. In the last two rows, UNETR confuses *stomach*, *pancreas*, as well as *spleen*, leading to inaccurate segmentation. Additionally, it under-segments *portal and splenic veins*. On the other hand, Swin UNETR and nnFormer under-segment *stomach* and *left adrenal gland*, respectively. However, UNETR++ shows precise organ segmentation with improved boundary delineation in these particular examples.

2) BTCV Dataset: Table III presents the comparison on BTCV test set. Here, all results are based on a single model accuracy without any ensemble, pre-training, or additional data. We report the results on all 13 organs along with the corresponding mean performance over all organs. UNETR and Swin-UNETR achieve a mean DSC of 76.0% and 80.44%, respectively. Among the existing methods, nnUNet obtains a performance of 83.16% mean DSC, but it comes at the cost of 358G FLOPs. In comparison, UNETR++ performs favorably against nnUNet by achieving a mean DSC of 83.28%, while requiring significantly fewer FLOPs (358G nnUNet vs. 31G UNETR++).

3) ACDC Dataset: Table IV shows the comparison on ACDC. Here, all results are reported with a single model accuracy and without using any pre-training, model ensemble, or additional data. UNETR and nnFormer achieve mean DSC of 86.61% and 92.06%, respectively. UNETR++ achieves improved performance with a mean DSC of 92.83%. To validate the statistical significance of UNETR++ over the best existing baseline (nnUNet), we compute the average p-values between UNETR++ and nnUNet for the evaluation metrics as follows: DSC (< 1e-2), NSD (< 1e-2), HD95 (<1e-2), and MASD (< 1e-2). Fig. 5 shows qualitative comparisons for

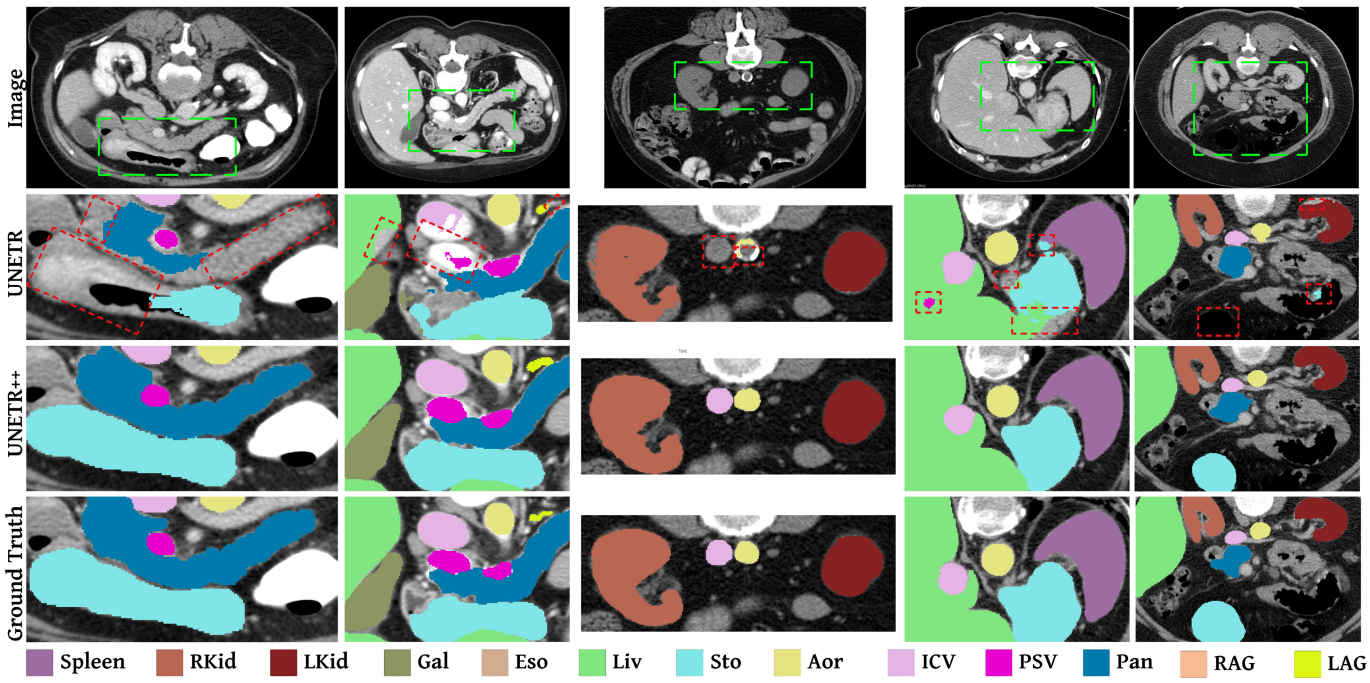

Fig. 3. **Qualitative comparison between UNETR++ and the baseline UNETR.** The baseline struggles to correctly segment different organs (marked in red dashed box). We enlarge multiple organs (marked with green dashed boxes in the first row) from several cases. Our UNETR++ achieves promising segmentation performance by accurately segmenting the organs. Best viewed zoomed in.

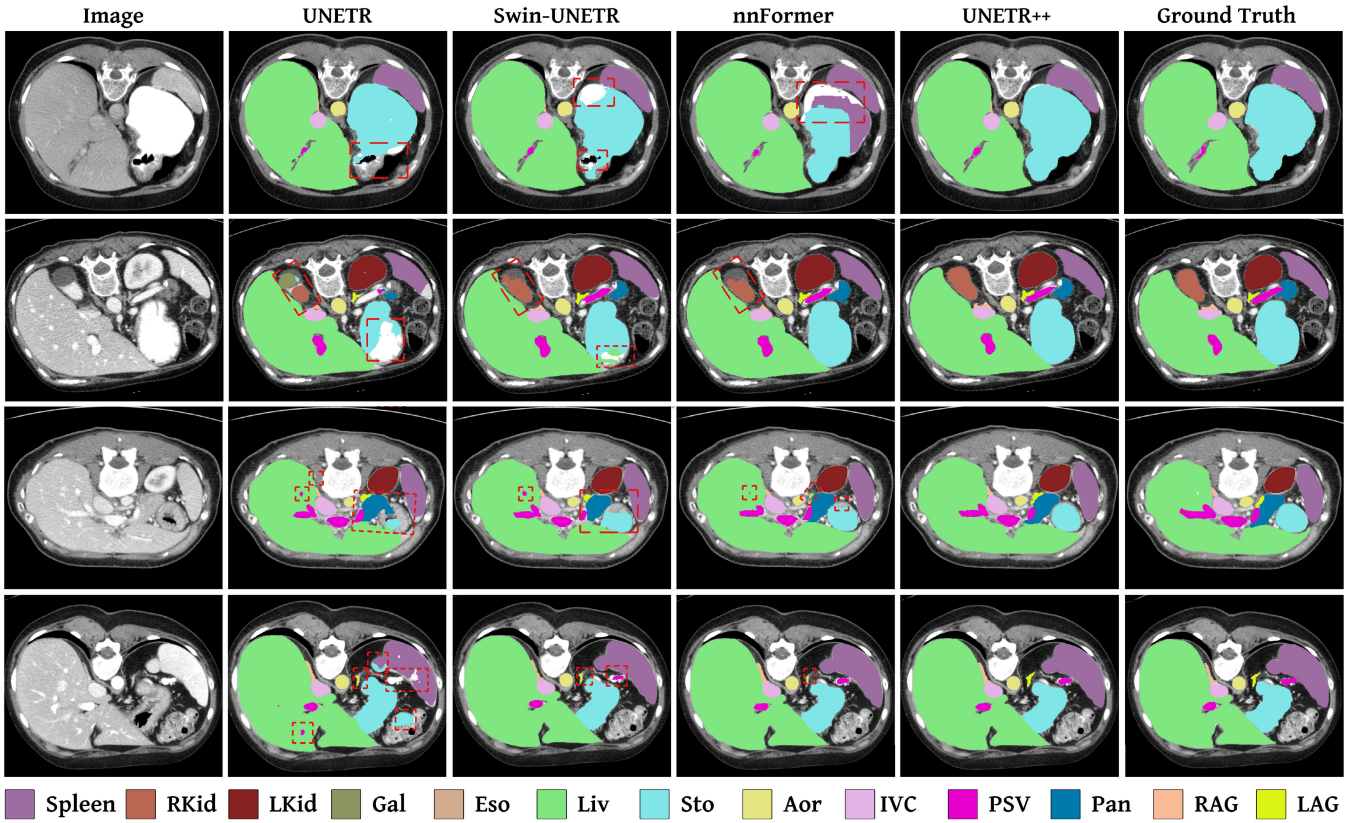

Fig. 4. **Qualitative comparison on multi-organ segmentation task.** Here, we compare our UNETR++ with existing methods: UNETR, Swin UNETR, and nnFormer. Existing methods struggle to correctly segment different organs (marked in red dashed box). UNETR++ achieves promising segmentation performance by accurately segmenting the organs. Best viewed zoomed in.

different cases between UNETR++ and existing approaches, nnFormer and UNETR on the ACDC dataset. The inaccurate predictions are marked with red dashed boxes. In the first row, UNETR and nnFormer under-segment the *right ventricular*

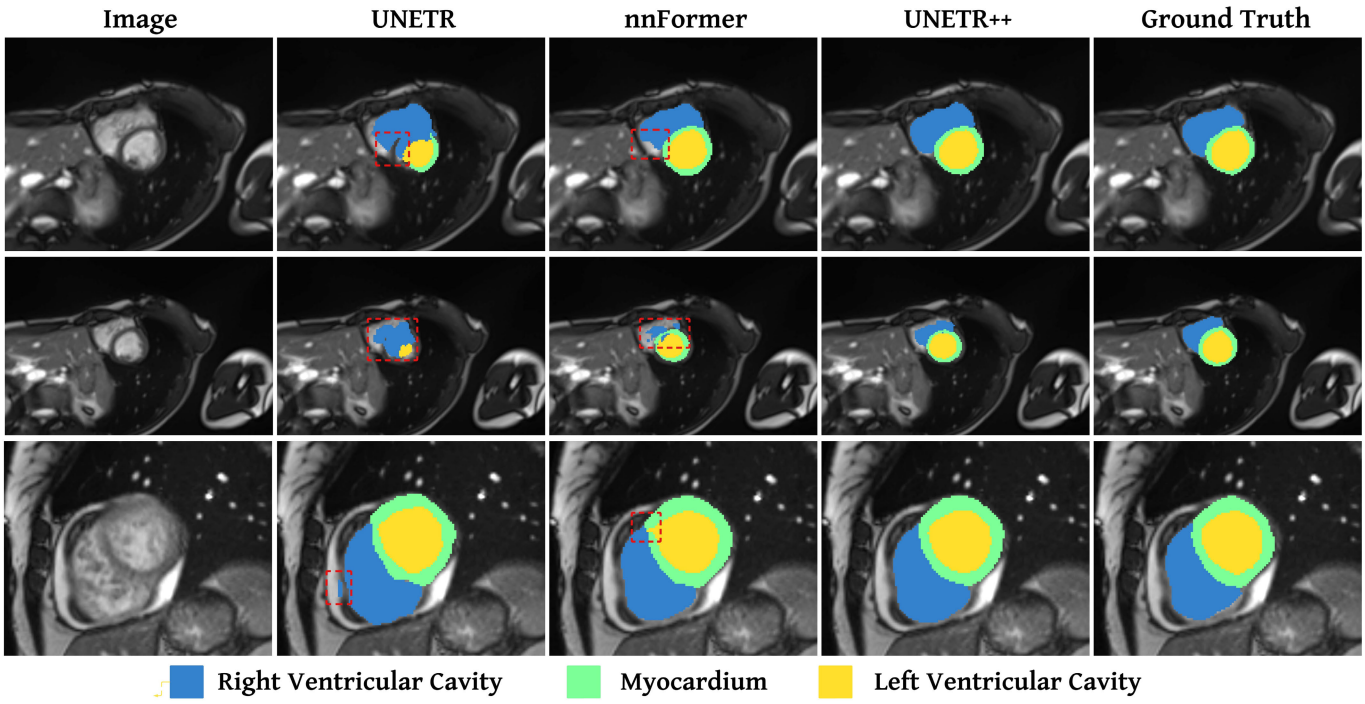

Fig. 5. **Qualitative comparison on the ACDC dataset.** We compare our UNETR++ with existing methods: UNETR and nnFormer. It is noticeable that the existing methods struggle to correctly segment different organs (marked in red dashed box). Our UNETR++ achieves favorable segmentation performance by accurately segmenting the organs. Best viewed zoomed in.

TABLE IV
STATE-OF-THE-ART COMPARISON ON ACDC DATASET. WE REPORT THE PERFORMANCE ON *right Ventricle* (RV), *left Ventricle* (LV), AND *myocardium* (MYO) ALONG WITH MEAN RESULTS OF DSC

Methods	RV	Myo	LV	Avg
TransUNet [35]	88.86	84.54	95.73	89.71
Swin-UNet [33]	88.55	85.62	95.83	90.00
UNETR [1]	85.29	86.52	94.02	86.61
MISSFormer [47]	86.36	85.75	91.59	87.90
CoTr [38]	89.13	88.84	95.16	91.04
nnUNet [2]	90.96	90.34	95.92	92.41
MedNeXt-M-K3 [4]	89.37	89.55	95.37	91.43
MedNeXt-M-K5 [4]	88.79	89.14	95.06	91.00
nnFormer [5]	90.94	89.58	95.65	92.06
UNETR++	**91.89**	**90.61**	**96.00**	**92.83**

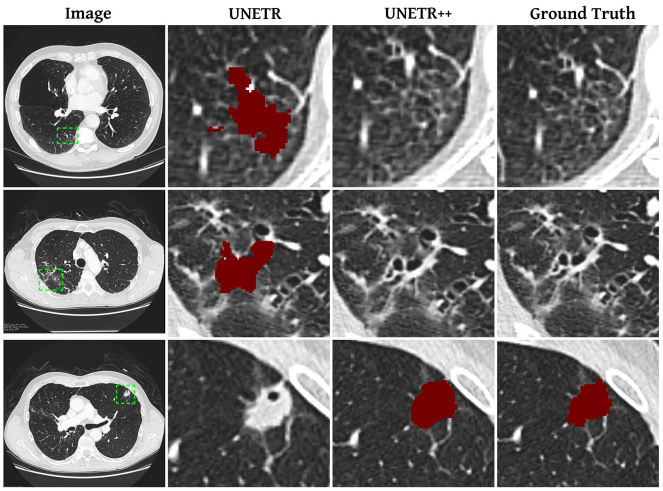

Fig. 6. **Qualitative comparison between the baseline UNETR [1] and our UNETR++ on Decathlon-Lung dataset.** The enlarged area is marked with a green box. UNETR++ has better segmentation and less *false positives* for segmenting the tumors as compared to the baseline. Best viewed zoomed in.

(RV) cavity, while our UNETR++ accurately segments all three categories. In the second row, we present a difficult sample where the sizes of all three heart segments are comparatively smaller. In this case, both UNETR and nnFormer under-segments and struggle to delineate between the segments, while UNETR++ gives a better segmentation. In the last row, we present a simpler sample. However, the existing methods over-segment the *RV cavity* and the *myocardium* in this case, while UNETR++ provides better delineation and provides a segmentation very close to the ground truth. Similar to the observation from Synapse, these qualitative examples show that UNETR++ achieves better delineation for the three heart segments without under-segmenting or over-segmenting the tissues, thus suggesting the importance of its inter-dependent spatial and channel features encoded in the proposed EPA block.

4) BraTS Dataset: Table V shows segmentation performance of UNETR++ and other existing methods on brain tumor segmentation. UNETR++ is better than the baseline UNETR by 1.5% and 1% in DSC and NSD, respectively. Also, UNETR++ has less HD95 and MASD distances. Although the recently introduced MedNeXt-M-K3 [4] has superior NSD

TABLE V
COMPARISON ON BRATS (BRAIN TUMOR SEGMENTATION) AND DECATHALON-LUNG DATASETS. UNETR++ ACHIEVES FAVORABLE SEGMENTATION RESULTS IN TERMS OF (DSC, NSD, HD95, AND MASD)

Task/Modality	Brain tumor Segmentation (MRI)							Decathalon-Lung Segmentation (CT)			
Metrics	WT	ET	TC	DSC ↑	NSD ↑	HD95 ↓	MASD ↓	DSC ↑	NSD ↑	HD95 ↓	MASD ↓
UNETR [1]	90.35	76.30	77.02	81.22	38.03	6.61	1.85	73.29	63.70	23.84	20.86
TransBTS [43]	90.91	77.86	76.10	81.62	38.66	5.80	1.91	70.38	65.19	30.09	16.59
Swin-UNETR [6]	91.12	77.65	78.41	82.39	39.22	5.33	1.80	75.55	71.38	28.74	14.47
CoTr [38]	91.01	77.52	77.43	81.99	38.73	5.78	1.84	75.74	70.48	27.91	9.94
nnUNet [2]	91.21	77.96	78.05	82.41	38.37	5.58	1.78	74.31	69.28	28.52	18.91
nnFormer [5]	91.23	77.84	77.91	82.34	37.42	5.18	1.66	77.95	71.38	16.25	13.52
MedNeXt-M-K3 [4]	**91.42**	78.24	77.98	82.55	**40.46**	5.13	1.65	80.14	**75.01**	2.85	1.16
MedNeXt-M-K5 [4]	91.21	78.15	78.03	82.46	38.65	5.37	1.81	79.51	74.41	2.84	1.14
UNETR++	91.27	**78.39**	**78.60**	**82.75**	39.08	**5.05**	**1.50**	**80.68**	73.92	**2.79**	**1.0**

TABLE VI
COMPUTATIONAL COMPARISON ON BRATS DATASET. UNETR++ IS EFFICIENT IN GFLOPS AND OPERATES AT MUCH FASTER INFERENCE SPEED (GPU T. AND CPU T. IN MS) AND REQUIRES LESS GPU MEMORY (MEM IN GB)

Model	Params	FLOPs	Mem	GPU T.	CPU T.
nnUNet [2]	**16.8**	410.1	3.7	90.9	3625.5
MedNeXt-M-K3 [4]	17.6	248.0	15.3	186.1	19955.4
MedNeXt-M-K5 [4]	18.3	308.0	18.8	351.5	64260.8
UNETR [1]	92.5	153.5	3.3	82.5	2145.0
Swin-UNETR [6]	62.8	572.4	19.7	228.6	7612.3
CoTr [38]	41.9	668.5	9.8	174.6	4236.2
nnFormer [5]	149.6	421.5	12.6	148.0	5247.5
UNETR++	42.6	**70.1**	**2.4**	**62.4**	**1497.7**

than UNETR++, our method outperforms in DSC, HD95, and MASD with less GPU memory and faster inference speed. We validate the statistical significance of UNETR++ over MedNeXt-M-K3 [4] by computing the average p-values for the evaluation metrics: DSC (< 1e-2), NSD (< 1e-2), HD95 (<1e-2), and MASD (< 1e-2).

In Table VI, we show a comprehensive comparison of the number of parameters, FLOPs, GPU memory, and GPU and CPU inference speed for both state-of-the-art CNN-based and hybrid-based methods on BraTS dataset. For a fair comparison, we use the same input size and pre-processing strategy. We compare speed on Quadro RTX 6000 24 GB GPU & 32 Core Intel(R) Xeon(R) 4215 CPU. Here, inference speed is avg. forward pass time using $1 \times 128 \times 128 \times 128$ input size of BraTS. Compared to recent transformer-based methods, our UNETR++ achieves favorable performance while operating at a faster inference speed as well as requiring significantly less GPU memory. Although the CNN-based methods (nnUNet [2] and MedNeXt [4]) are more efficient in terms of number of parameters due to the inherent design of the convolutional kernels, UNETR++ has much fewer FLOPs, less GPU memory, and faster CPU and GPU inference speed. In particular, UNETR++ requires 6.4 × less GPU memory than MedNeXt-M-K3 [4], while achieving 3 × faster GPU inference and a remarkable 13× faster CPU inference.

TABLE VII
ABLATION ON THE ATTENTION MODULES OF THE EPA BLOCK ON SYNAPSE DATASET

Method	DSC ↑	NSD ↑	HD95 ↓	MASD ↓
UNETR (Baseline)	78.35	76.58	18.59	8.81
UNETR++ (SA only)	86.42	85.01	8.65	3.92
UNETR++ (CA only)	86.39	84.82	12.28	3.32
UNETR++ (EPA)	**87.22**	**85.99**	**7.53**	**3.39**

5) Decathalon-Lung Dataset: In Table V, we evaluate our UNETR++ and other existing methods on the Decathalon-Lung cancer segmentation. UNETR and nnFormer obtain DSC of 73.29% and 77.95%, respectively. Notably, UNETR, nnUNet, and nnFormer exhibit high HD95 and MASD, which means they are struggling to accurately delineate the boundaries of the lung tumors. The recently introduced MedNeXt variants [4] perform well in most evaluation metrics, but at the cost of GPU memory and inference speed. UNETR++ achieves better performance compared to the best existing methods by achieving a mean DSC of 80.68%, with less HD95 and MASD distances. We show in Fig. 6 a baseline comparison between UNETR and our UNETR++. In the first two rows, UNETR++ has less *false positives*, while in the third row, UNETR under-segments the whole tumor and UNETR++ segments it correctly. The average p-values between UNETR++ and the best existing baseline (MedNeXt-M-K3) are as follows: DSC (< 1e-2), NSD (< 1e-2), HD95 (<1e-2), and MASD (< 1e-2).

D. Ablations

In this section, we conduct various ablations to analyze the scalability and effectiveness of the proposed EPA block and provide more insights about our method. First, To investigate the scalability of UNETR++, we designed an experiment with feature maps of size [64, 128, 256, 512] instead of [32, 64, 128, 256] on the BTCV dataset. Although the number of parameters with this change increased to 94.24M and the FLOPs increased to 117G, the average dice similarity coefficient (DSC) is

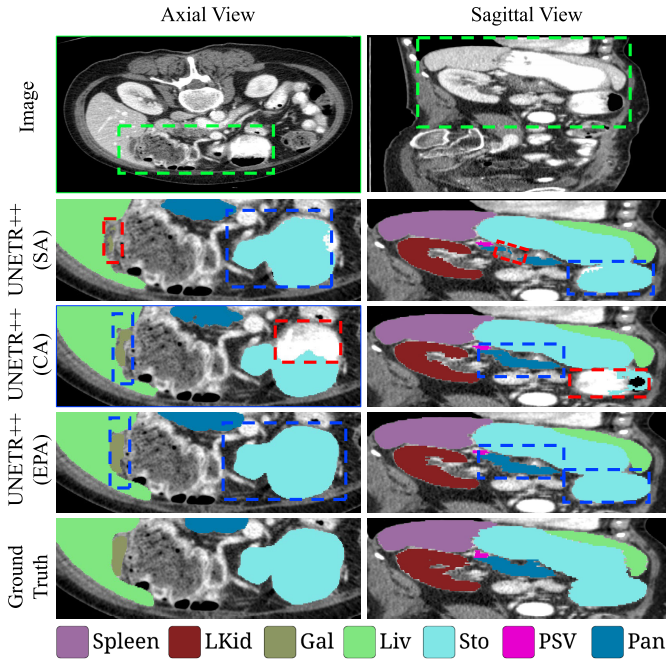

Fig. 7. **Qualitative comparison of UNETR++ with spatial attention (SA), channel attention (CA), and the proposed EPA block**. Green dashed boxes highlight the enlarged areas, red dashed boxes indicate miss-segmentation, and blue dashed boxes denote correct segmentation. Best viewed zoomed in.

improved from 83.28% to 84.27%, which proves the scalability of UNETR++ without using any ensemble, pre-training or additional custom data.

In Table VIII, we present an ablation study on the attention modules of the EPA block on Synapse dataset. The first row is the baseline UNETR, and the second and third rows show the impact of spatial attention (SA) and channel attention (CA) separately. In the last row, both attentions are combined in the proposed EPA block with shared QK layers to learn only the complementary features. In addition, we show in Fig. 7 a qualitative comparison for SA, CA, and the proposed EPA block with the same training seed. In the first row, we show two different slices from two views to show different orientations used to view the human body (Axial and Sagittal). In the second row, UNETR++ with SA only suffers from segmenting small organs likely due to not encoding the dependencies between the feature maps. In the third row, UNETR++ with CA only struggles to segment the large organs, probably due to not encoding the global information from the current slice. In the third row, UNETR++ with the proposed EPA block in both views is able to segment all organs correctly, due to encoding the inter-dependencies between the feature maps, as well as encoding global information (marked in blue dashed boxes).

To show the effectiveness of the proposed EPA block, we conduct a comparison on Synapse dataset to compare our EPA block with other attention methods. In the first row, we replace the proposed EPA block in our UNETR++ with multi-head self-attention. In the second raw, we replace the EPA with the gated attention from Attention-UNet [42]. In the third row, the EPA block is replaced with 3D CBAM block [39]. In the fourth row, we use squeeze-and-excitation (SE) block [41] instead of the proposed EPA block. In the fifth row, we replace the EPA block with a MedNeXt block. Our EPA block achieves superior results on all evaluation metrics by effectively encoding both spatial and channel features, thereby learning complementary features compared to other attention methods in UNETR++.

TABLE VIII
ABLATION ON EPA BLOCK COMPARED TO OTHER ATTENTION MODULES ON SYNAPSE DATASET IN UNETR++ FRAMEWORK

Method	DSC ↑	NSD ↑	HD95 ↓	MASD ↓
EPA replaced with MHSA	86.97	85.89	10.47	3.71
EPA replaced with GA	84.23	83.18	12.52	4.80
EPA replaced with CBAM	85.44	84.33	14.98	4.43
EPA replaced with SE	85.81	85.34	10.97	4.24
EPA replaced with MedNeXt	86.27	85.22	10.60	3.94
UNETR++ (EPA)	**87.22**	**85.99**	**7.53**	**3.39**

TABLE IX
ABLATION ON THE HYPER-PARMETER P OF THE EPA BLOCK. WE ABLATE PROJECTING hwd TO DIFFERENT VALUES WITH RESPECT TO THE NUMBER OF PARAMETERS, FLOPS, AND DSC ON SYNAPSE DATASET

Method	Params (M)	FLOPs (G)	DSC (%)
Projection dim of 32	**39.04**	**47.04**	86.45
Projection dim of 128	49.80	48.63	86.96
Projection dim of 64	42.96	47.98	**87.22**

The hyper-parameter P denotes the projected dimension of hwd within the EPA block, it is used to reduce the quadratic complexity to linear complexity. In Table IX, we ablate different projection sizes for the first three stages to show the effect of the projection with respect to the number of parameters, FLOPs, and DSC on Synapse. It is notable that a projection size of 64 achieves an optimal trade-off between the complexity and the resulting DSC in the Synapse dataset.

V. CONCLUSION AND DISCUSSION

We propose a hierarchical approach, named UNETR++, for 3D medical segmentation. Our UNETR++ introduces an efficient paired attention (EPA) block to encode enriched inter-dependent spatial and channel features by using spatial and channel attention. Within the EPA block, we share the weights of query and key mapping functions to better communicate between spatial and channel branches, providing complementary benefits as well as reducing the parameters. Our UNETR++ achieves favorable segmentation results on five datasets (Synapse, ACDC, BTCV, BraTS, and Decathlon-Lung) while significantly reducing the model complexity (in terms of parameters, FLOPs), compared to the best existing methods. Furthermore, we show that our UNETR++ has less GPU consumption, which is a critical factor for 3D segmentation tasks, and operates at a faster inference speed on both GPU and CPU platforms. Hence, UNETR++ offers a more versatile and resource-efficient solution, enhancing

the feasibility of deploying medical segmentation models on mobile platforms for real-time medical image analysis. The proposed EPA block is generic and can be used in other works. To validate that, we replace the self-attention block of TransBTS [43] by the proposed EPA in their framework and evaluate the updated model on Synapse. We notice that the DSC is increased by 1.2% (from 83.28% to 84.47%), NSD increased by 0.8%, HD95 is significantly reduced from 12.34 to 7.92, and MASD is reduced from 3.65 to 3.11.

To observe potential limitations of UNETR++, we analyze different outlier cases. Although our predictions are better than the existing methods and more similar to the ground truth, we find that there are a few cases where our model, as well as most of the existing methods, struggle to segment certain organs. When the geometric shape of the organs in a few slices is abnormal (delineated by thin borders), our model and most of the existing models struggle to segment them accurately. The reason might be the limited availability of training samples with such abnormal shapes compared to the normal samples. These localization errors are quantitatively observed in lower NSD in some cases compared to the most recent CNN-based method MedNeXt [4]. The reason could be attributed to the inductive bias of CNN-based methods, which excel in capturing spatial hierarchies and local patterns. We are planning to solve this problem by applying geometric data augmentation techniques at the pre-processing stage.

REFERENCES

[1] A. Hatamizadeh et al., "UNETR: Transformers for 3D medical image segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 1748–1758.

[2] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. H. Maier-Hein, "NnU-Net: A self-configuring method for deep learning-based biomedical image segmentation," *Nature Methods*, vol. 18, no. 2, pp. 203–211, Feb. 2021.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[4] S. Roy et al., "MedNeXt: Transformer-driven scaling of ConvNets for medical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2023, pp. 405–415.

[5] H.-Y. Zhou et al., "NnFormer: Volumetric medical image segmentation via a 3D transformer," *IEEE Trans. Image Process.*, vol. 32, pp. 4036–4045, 2023, doi: 10.1109/TIP.2023.3293771.

[6] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. R. Roth, and D. Xu, "Swin UNETR: Swin Transformers for semantic segmentation of brain tumors in MRI images," in *Proc. Int. MICCAI Brainlesion Workshop*, 2022, pp. 272–284.

[7] B. Landman, Z. Xu, J. Igelsias, M. Styner, T. Langerak, and A. Klein, "MICCAI multi-atlas labeling beyond the cranial vault-workshop and challenge," in *Proc. MICCAI*, 2015, p. 12.

[8] O. Bernard et al., "Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved?" *IEEE Trans. Med. Imag.*, vol. 37, no. 11, pp. 2514–2525, Nov. 2018.

[9] B. H. Menze, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[10] A. L. Simpson et al., "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," 2019, *arXiv:1902.09063*.

[11] Q. Zhu, B. Du, B. Turkbey, P. L. Choyke, and P. Yan, "Deeply-supervised CNN for prostate segmentation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 178–184.

[12] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Proc. Int. Workshop Deep Learn. Med. Image Anal.*, 2018, pp. 3–11.

[13] H. Huang et al., "UNet3+: A full-scale connected UNet for medical image segmentation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 1055–1059, doi: 10.1109/ICASSP40776.2020.9053405.

[14] S. Cai, Y. Tian, H. Lui, H. Zeng, Y. Wu, and G. Chen, "Dense-UNet: A novel multiphoton in vivo cellular image segmentation model based on a convolutional neural network," *Quant. Imag. Med. Surg.*, vol. 10, no. 6, pp. 1275–1285, Jun. 2020.

[15] E. Gibson et al., "Automatic multi-organ segmentation on abdominal CT with dense V-networks," *IEEE Trans. Med. Imag.*, vol. 37, no. 8, pp. 1822–1834, Aug. 2018.

[16] Q. Dou et al., "3D deeply supervised network for automatic liver segmentation from CT volumes," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2016, pp. 149–157.

[17] O. Cicek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2016, pp. 424–432.

[18] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 565–571, doi: 10.1109/3DV.2016.79.

[19] J. Valanarasu, P. Oza, I. Hacihaliloglu, and V. Patel, "Medical transformer: Gated axial-attention for medical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2021, pp. 36–46.

[20] H. R. Roth et al., "Hierarchical 3D fully convolutional networks for multi-organ segmentation," 2017, *arXiv:1704.06382*.

[21] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[22] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters-improve semantic segmentation by global convolutional network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4353–4361.

[23] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 833–851.

[24] Z. Li, H. Pan, Y. Zhu, and A. K. Qin, "PGD-UNet: A position-guided deformable network for simultaneous segmentation of organs and tumors," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.

[25] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11966–11976.

[26] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–22.

[27] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

[28] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," 2019, *arXiv:1904.10509*.

[29] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020, *arXiv:2006.04768*.

[30] M. Maaz et al., "EdgeNeXt: Efficiently amalgamated CNN-transformer architecture for mobile vision applications," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2022, pp. 3–20.

[31] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=rkgNKkHtvB

[32] N. Kim, D. Kim, S. Kwak, C. Lan, and W. Zeng, "ReSTR: Convolution-free referring image segmentation using transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 78–88.

[33] H. Cao et al., "Swin-UNet: UNet-like pure transformer for medical image segmentation," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2023, pp. 205–218.

[34] Y. Zhang, H. Liu, and Q. Hu, "TransFuse: Fusing transformers and CNNs for medical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2021, pp. 14–24.

[35] J. Chen et al., "TransUNet: Transformers make strong encoders for medical image segmentation," 2021, *arXiv:2102.04306*.

[36] A. Lin, B. Chen, J. Xu, Z. Zhang, G. Lu, and D. Zhang, "DS-TransUNet: Dual Swin transformer U-Net for medical image segmentation," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–15, 2022, doi: 10.1109/TIM.2022.3178991.

[37] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 9992–10002, doi: 10.1109/ICCV48922.2021.00986.

[38] Y. Xie, J. Zhang, C. Shen, and Y. Xia, "CoTr: Efficiently bridging CNN and transformer for 3D medical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2021, pp. 171–180.

[39] S. Woo, J. Park, J. Lee, and I. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. Workshops,* 2018, pp. 3–19.

[40] W. Wang, X. Tan, P. Zhang, and X. Wang, "A CBAM based multiscale transformer fusion approach for remote sensing image change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6817–6825, 2022, doi: 10.1109/JSTARS.2022.3198517.

[41] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7132–7141.

[42] O. Oktay et al., "Attention U-Net: Learning where to look for the pancreas," in *Proc. Med. Imag. Deep Learn.*, 2018, pp. 1–10.

[43] W. Wang, C. Chen, M. Ding, H. Yu, S. Zha, and J. Li, "TransBTS: Multimodal brain tumor segmentation using transformer," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2021, pp. 109–119.

[44] A. Reinke et al., "Common limitations of image processing metrics: A picture story," 2021, *arXiv:2104.05642*.

[45] L. Maier-Hein et al., "Metrics reloaded: Recommendations for image analysis validation," 2022, *arXiv:2206.01653*.

[46] M. Jorge Cardoso et al., "MONAI: An open-source framework for deep learning in healthcare," 2022, *arXiv:2211.02701*.

[47] X. Huang, Z. Deng, D. Li, X. Yuan, and Y. Fu, "MISSFormer: An effective transformer for 2D medical image segmentation," *IEEE Trans. Med. Imag.*, vol. 42, no. 5, pp. 1484–1494, May 2023, doi: 10.1109/TMI.2022.3230943.