Some Network for Medical Image Segmentation

First Author[1,2*], Second Author[2,3†] and Third Author[1,2†]

[1*]Department, Organization, Street, City, 100190, State, Country.
[2]Department, Organization, Street, City, 10587, State, Country.
[3]Department, Organization, Street, City, 610101, State, Country.

*Corresponding author(s). E-mail(s): iauthor@gmail.com;
Contributing authors: iiauthor@gmail.com; iiiauthor@gmail.com;
†These authors contributed equally to this work.

Abstract

Large amount of high-quality annotated 3D data for training is required by medical image segmentation area, this limitation restricts their broader clinical use in disease diagnosis, treatment planning, and monitoring. Semi-supervised learning (SSL) has emerged as a promising paradigm to address this data scarcity challenge by effectively leveraging both labeled and unlabeled data to enhance segmentation performance. While existing SSL approaches have primarily focused on model-centric innovations through novel regularization strategies, the recent emergence of promptable segmentation foundation models presents unprecedented opportunities for paradigmatic advancement. Segment Anything Model(SAM)'s sophisticated encoder-decoder architecture, with its innovative mask decoder and prompt encoder design, demonstrates exceptional adaptability and generalization capabilities across diverse segmentation tasks. However, the integration of such prompt-based segmentation models into semi-supervised medical image segmentation presents unique challenges. Specifically, these models require datasets with additional prompt annotations to guide the segmentation process, yet most existing semi-supervised medical imaging datasets lack such prompt information. This absence creates a critical bottleneck: how to generate appropriate prompts and achieve high-performance segmentation of target lesion regions without explicit prompt supervision remains an open and challenging problem. To address this fundamental limitation, we introduce a novel Semi-Supervised 3D Medical Image Segmentation framework with Decoupled Uncertainty Prompt Generator (DUPG), which intelligently generates effective prompts while maintaining superior segmentation performance in data-scarce scenarios. To address these limitations, we propose XXX, a novel semi-supervised learning training processes that harnesses the power of foundation models while maintaining computational efficiency for medical image segmentation.

Our approaches and demonstrates superior performance across multiple medical imaging modalities while requiring significantly reduced computational resources compared to existing methods.

Keywords: Medical Image Segmentation, Semi-Supervised Learning, Deep Learning

1 Introduction

Medical image segmentation plays a pivotal role in medical imaging by enabling precise delineation of anatomical structures and pathological regions. Accurate segmentation is indispensable for clinical applications, including disease diagnosis, treatment planning, and longitudinal monitoring of disease progression. Deep learning-based methods have revolutionized medical image segmentation, achieving remarkable success in various tasks. However, these fully-supervised methods often face two main challenges: (1) A large amounts of annotated data is required in training part, which is a significant limitation in clinical practice due to the time-consuming and labor-intensive nature of manual annotation. (2) The performance of these methods heavily relies on the availability of high-quality annotated data, which is often scarce in many medical imaging tasks. Besides, many commonly used medical images like computed tomography (CT) and magnetic resonance imaging (MRI) scans are 3D volumetric data, further increase the burden of manual annotation compared with 2D images. Therefore, training deep learning models with limited annotated data remains a significant challenge in the field of medical image segmentation.

To address the challenge of limited annotated data, researchers have made substantial efforts in developing annotation-efficient methods in medical image segmentation. Semi-supervised learning (SSL) could effectively leverage the information contained in both labeled and unlabeled data and improves model performance. this method have shown great potential in medical image segmentation, enabling the utilization of both labeled and unlabeled data to improve segmentation performance. there are mainly two types of SSL methods: consistency regularization and pseudo-labeling. Using pseudo-labeling path,it could enlarge the training data through label generalization to improve stability of the model while training. it also could use data augmentation and consistency regularization to enhance the model's robustness and generalization ability. these methods have shown promising results in various medical image segmentation tasks, demonstrating their effectiveness in improving model performance with limited annotated data.

Recent years have witnessed the emergence of foundation models in computer vision, notably represented by the Segment Anything Model (SAM) for 2D image segmentation, which demonstrates impressive zero-shot generalization capabilities across various natural image segmentation tasks. However, because of the suubstantial domaingap, its performance remains suboptimal in medical images segmentation tasks.

2 Relative works

2.1 U-shaped CNNs Methods

U-shaped convolutional neural networks are essential technology driving groundbreaking progress in the field of medical image segmentation, particularly for their powerful feature extraction capabilities. Jonathan Long et al. [1] revolutionized segmentation with FCNs by replacing fully connected layers in traditional CNNs with convolutional layers, enabling segmentation for images of arbitrary sizes and significantly improving accuracy. Olaf Ronneberger et al. [2] further extended this architecture with U-Net, adopting an encoder-decoder structure with skip connections. The encoder gradually extracts features and downsamples the input, while the decoder upsamples the feature maps to recover resolution. Skip connections allow low-level features to directly feed into the corresponding decoder stage, greatly enhancing the integration of multi-scale features [3]. Milletari et al. [4] proposed that V-Net can better capture information on 3D space by using 3D convolution operations. Zongwei Zhou et al. [5] introduced UNet++, a nested U-shaped architecture that optimizes feature transmission and integration through dense skip connections and multi-scale feature fusion. Ange Lou et al. [6] proposed DC-UNet, this method incorporates a dual-channel design into the U-Net architecture to effectively combine multi-modal feature representations, resulting in substantial improvements in both segmentation performance for multi-modal medical images and computational efficiency. Huimin Huang et al. [7] introduced UNet3+, the architecture leverages CNNs to dynamically combine low-level and high-level information, achieving efficient 3D medical image segmentation. The U-shaped architecture has achieved remarkable results in medical image segmentation, we also adopted it to develop a multi-organ segmentation network.

2.2 Attention-Based Methods

Attention mechanisms were first applied in Natural Language Processing (NLP) and later introduced to computer vision, yielding promising results in tasks such as classification and detection. This strategy dynamically adjusts the focus of the network on the feature map, significantly enhancing performance in complex visual tasks[8]. Hu et al. [9] proposed the Channel Attention Mechanism (CAM), it dynamically adjusts the feature map along the channel dimension and improves the model's ability to evaluate feature importance. Oktay et al. [10] introduced the attention gate mechanism in U-Net, which dynamically weighs feature maps during the decoder stage, suppressing irrelevant background information and improving segmentation accuracy and robustness. Dosovitskiy et al. [11] introduced the Vision Transformer (ViT) architecture, this work applies Transformer mechanisms to visual tasks by dividing images into patches, further advancing the field. Chen et al. [12] proposed combining the Transformer and U-Net structures, enhancing global feature modeling in the encoder while retaining multi-scale feature advantages. Liu et al. [13] presented the Swin Transformer, which achieves efficient global feature modeling through a shifted window mechanism. Following this, Cao et al. [14] introduced the Swin-Unet architecture, integrating Swin Transformer modules into U-Net encoder and decoder stages, effectively utilizing local

and global feature integration to improve segmentation accuracy and computational efficiency. Also, Huang et al. [15] proposed an enhanced transformer context bridge with the enhanced transformer block that extracts the long-range dependencies and local context of multi-scale features generated by a hierarchical transformer encoder. This progress underlines the importance of global feature extraction to effectively tackle complex visual tasks in medical image segmentation.

2.3 Lightweight-Based Methods

Despite significant advances in U-shaped CNNs and attention-based methods, there were still challenges with computational resources and efficiency. It mainly dues to the complexity of CNNs and the non-linear increase in the computational cost of attention mechanisms in Transformer architectures as input image size increases [16]. To address these issues, Xie et al. [17] proposed ResNeXt, this architecture reduces the computational cost of convolutions by introducing grouped convolutions in image classification. Grouped convolution significantly lowers computational complexity while maintaining model performance by dividing input features into groups. Similarly, Howard et al. [18] proposed MobileNet, which used depthwise and pointwise convolutions to reduce both parameter count and computational cost. This technique has been widely adopted in mobile and embedded devices and serves as a foundation for lightweight convolutional network design. Ding et al. [19] demonstrated the effectiveness of large-kernel convolutions, this large-kernel can replace parts of the self-attention mechanism and preserve efficient convolution operations. It also reduces computational complexity and retains the ability to model long-range dependencies. Trockman et al. [20] proposed ConvMixer, which uses large convolution kernels to mix distant spatial position information. Wang et al. [21] introduced the PVT model, this model combines multi-scale local convolutions with global attention mechanisms. Liu et al. proposed ConvNeXt [22], this work used large convolution kernels and Swin Transformer [13] architecture design tricks to improve the performance of fully convolutional networks in all aspects. Tolstikhin et al. [23] proposed UNeXt, which designed a lightweight medical image segmentation network based on a hybrid architecture of CNNs and multilayer perceptron. Tang et al. [24] demonstrated that ConvUNeXt leverages and improves the ConvNeXt block for efficient segmentation. Guo et al. [25] proposed the Visual Attention Network, which incorporates large-kernel convolution attention, expanding the receptive field to capture broader contextual information. However, while these lightweight methods have shown promising results in natural image processing, they still face challenges in medical image processing due to the differences in feature distribution between medical and natural images.

3 Method

3.1 Architecture Overview

As illustrated in Figure 1, the proposed network is primarily divided into an encoder and a decoder, this architecture adopts a multi-level U-shaped structure.

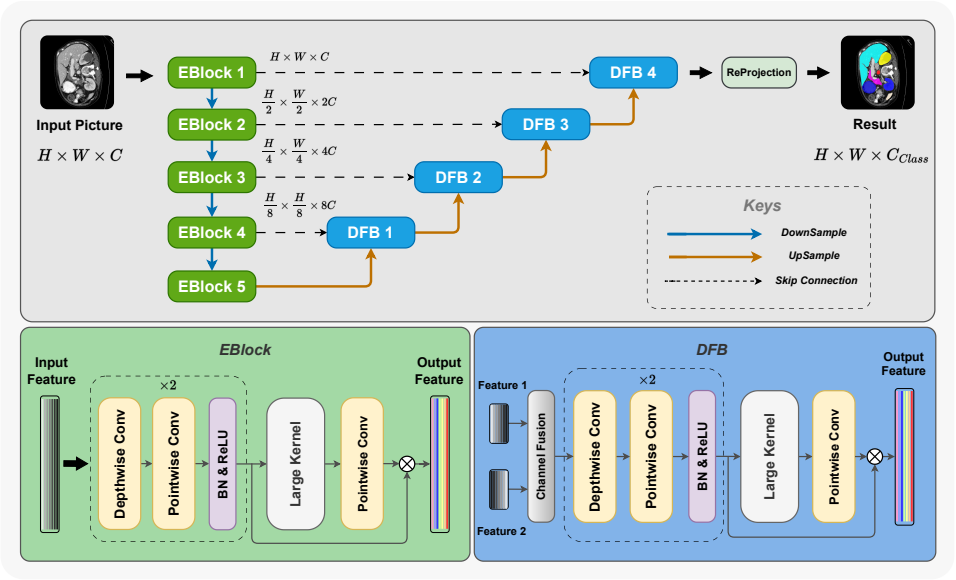


Fig. 1: An illustration of **DLKUNet**, which is composed of encoder (Green), decoder (Blue) and reprojection layer. The blue arrow represents the downsample between different modules in the encoder, and the orange arrow represents the upsample in the decoder. The two pictures below show the internal structure of **EBlock** and **DFB** respectively.

In the encoder, a progressive design strategy is employed to capture multi-scale information and effectively compress the feature representation of the input image. The encoder consists of five DLKBlock layers and the first layer performs initial feature extraction by expanding the input feature channels to C. DLKBlock consists of downsampling operations and an Encoder Block(EBlock) in each encoder layer. Similar to models like U-Net [2] and MobileVit [18], we decouple the downsampling layers to enhance training stability and overall model performance. To keep better trade-off computational efficiency and accuracy, we use the max-pooling downsampling method with 2×2 kernel size and stride as 2. This downsampling strategy retains crucial feature information while reducing computational complexity, thereby enhancing the effectiveness of the model in medical image segmentation.

In the decoder, a stepwise design is adopted to maintain synergy with the encoder and restore feature details while ensuring model performance. The decoder consists of four DLKBlock layers and a reprojection layer. The reprojection layer is responsible for restoring the number of channels and refining the feature information, ultimately producing precise multi-class segmentation results. Each DLKBlock in the decoder layer consists of a Decoder Fusion Block(DFB) and upsampling operations, primarily used to merge feature tensors from multiple pathways. The upsampling process uses a transposed convolution with 3×3 kernel size, stride as 1, and padding as 1 to double

the resolution of the feature maps. This provides richer and more precise input data for subsequent processing.

3.2 EBlock

As shown in Figure 2, The structure of the EBlock consists of two main components: the convolution part and the attention part.

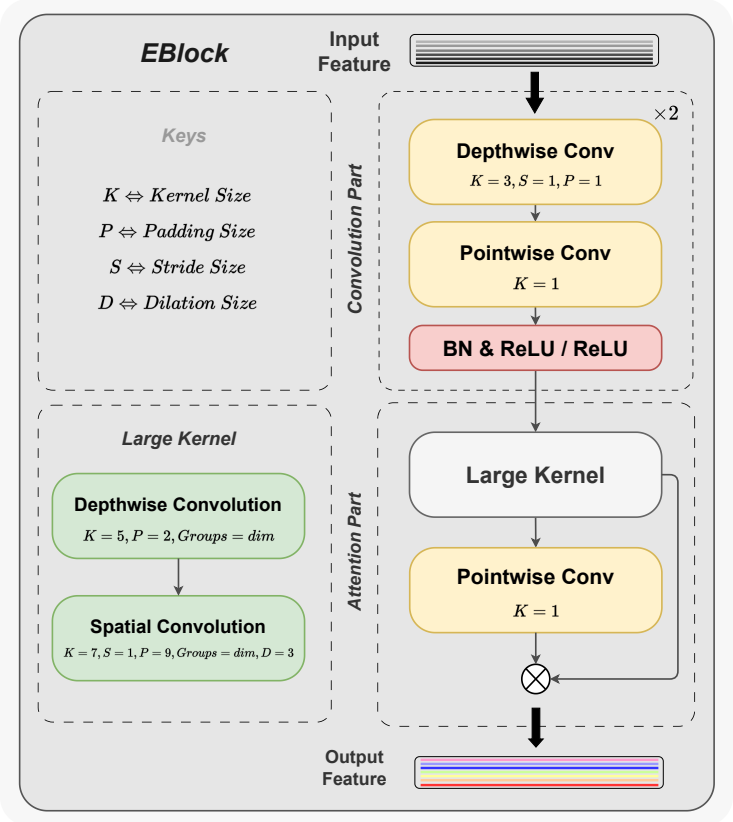


Fig. 2: The architecture of **EBlock** and the components of convolution part and attention part

The convolution part primarily comprises two consecutive layers of depthwise convolution, pointwise convolution, an activation function, and a batch normalization layer. We configure the kernel size as 3×3, stride as 1, and padding as 1. Depthwise convolution processes each input channel independently, whereas traditional convolution spans all channels simultaneously. It is a specialized convolution operation that

performs convolutions independently within each input channel, which focuses on capturing local spatial features. Pointwise convolution focuses on feature combination and transformation along the channel dimension. It operates on each pixel of the input feature map, linearly combining features from different channels to generate new channel features with a 1×1 convolution kernel. This sequence design significantly enhances the model's feature extraction capability while improving computational efficiency.

Following the depthwise and pointwise convolution operations, we incorporate a subsequent layer combined with batch normalization and ReLU activation for the initial iteration. This combination will employ only ReLU activation for the next subsequent iteration. By utilizing EBlock, DLKUNet efficiently captures both spatial and semantic information and optimizes the parameter count by reducing computational overhead, this architecture ensures the model's effectiveness and performance in complex tasks.

Assuming the input feature map $Z \in \mathbb{R}^{H \times W \times C}$, where H and W represent the height and width of the feature map, and C denotes the number of channels, the convolution part can be formulated as follows

$$Z'_{l-1} = \sigma_1 \left(BN \left(PointwiseConv \left(DepthwiseConv \left(Z_{l-1} \right) \right) \right) \right) \tag{1}$$

$$Z_l = BN \left(PointwiseConv \left(DepthwiseConv \left(Z'_{l-1} \right) \right) \right) \tag{2}$$

where Z_{l-1} represents the feature tensor passed from the $(L-1)$ layer, σ_1 denotes the ReLU activation function, and BN refers to the batch normalization layer, which is used to standardize features. In the L layer of the network, the input feature map Z_{l-1} is first processed by depthwise convolution, an operation that independently applies convolutions on each input channel. The result is then passed through pointwise convolution, which facilitates the fusion of information across channels. The intermediate feature representation Z'_{l-1} is subsequently obtained through the combination of batch normalization and activation functions. This intermediate representation is processed again using similar depthwise and pointwise convolutions, followed by batch normalization, resulting in the output feature map Z_l.

The attention part consists of the large kernel module, the channel convolution module, and the Hadamard product operation. We configure the kernel size as 5, padding as 2 and the number of groups equals the input dimension. The attention part is designed to enhance the model's feature representation by capturing global contextual information and facilitating feature interaction across channels. By increasing the size of the receptive field, the large kernel is capable of aggregating features from distant spatial regions, effectively incorporating more global context. It helps the model better understand spatial relationships, making it more sensitive to subtle or complex features in the image.

The output from the large kernel is processed through pointwise convolution to reduce the number of channels, thereby decreasing computational complexity while preserving key information. The Hadamard product operation calculates inter-channel correlations to highlight important features while suppressing irrelevant or redundant information. This design ensures that critical features receive higher priority within

the channel dimension, improving the model's selective attention mechanism. Additionally, the residual connection maintains the integrity of information, preventing feature degradation.

The attention module can be described using the following methodology

$$Z_{LargeKernel} = PointwiseConv\left(SpatialConv\left(DepthwiseConv\left(Z_{l-1}\right)\right)\right) \tag{3}$$

$$Z_l = Z_{l-1} \otimes ChannelConv_{1\times1}\left(Z_{LargeKernel}\right) \tag{4}$$

where Z_{l-1} represents the input feature map from the convolution part, and each channel is processed independently. Spatial convolution is applied to expand the receptive field, capturing a broader range of contextual information. Following this, pointwise convolution is employed to integrate the channel information, further strengthening the model's ability to capture global features. The resulting feature map $Z_{LargeKernel}$ is combined with the output from the 1×1 channel convolution by using a Hadamard product operation to enable inter-channel information interaction as Z_l.

3.3 DFB

As shown in Figure 3, the Decoder Fusion Block consists of a fusion part, a convolution part, and an attention Part.

The fusion part primarily consists of an auto-resize component and a channel concatenation operation. The auto-resize component is designed to resolve the height and width differences between the skip connection feature map and the decoder feature map. To address these discrepancies, the decoder feature map is first upsampled to double its resolution, followed by zero-padding to match the dimensions of the skip connection feature map. By applying appropriate upsampling and padding operations, the model ensures that feature maps from different paths and scales can be seamlessly merged.

The convolution part and the attention part adopt a structure analogous to that of the encoder, facilitating close integration and synergy between the two. It includes depthwise convolution, batch normalization, ReLU activation, and large kernel convolution, and further integrates features through pointwise convolution (with 1×1 kernel size). This modular design enables the efficient processing of fused features, extracting more refined and high-quality feature information.

Through these computational modules, the decoder can effectively integrate the processed feature tensors, improving the network's feature representation.

3.4 Architecture Variants

We designed different network configurations as Figure 4 and Figure 1. These networks are named DLKUNet-S, DLKUNet-M, DLKUNet-L. "S" denotes the smallest network, which is suitable for resource-constrained environments such as edge and mobile computing. "M" represents the medium-sized network, striking a balance between computational cost and performance. "L" indicates a larger parameter network in

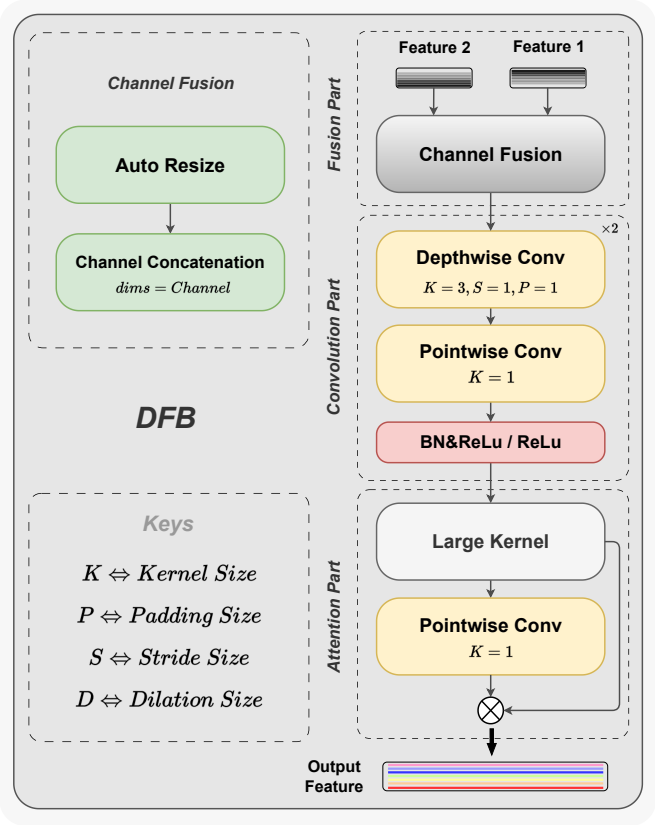


Fig. 3: The architecture of **DFB** and its major components with working processes

Figure 1, which offers superior segmentation performance compared to the smaller versions.

The configurations of the different DLKUNet models are summarized in Table 1. Each model varies in terms of parameter count (Params), floating point operations (FLOPs) when input resolution is 224×224, the number of EBlock, the number of DFB, and the channel configuration for each skip connection (Channels Num).

DLKUNet-S is the smallest model, making it highly efficient for use in resource-constrained environments with only 1.13 million parameters and 15.95 GFLOPs. It utilizes 3 encoder blocks and 2 decoder fusion blocks, and skip connection channels configured as $C = 96$. DLKUNet-M is a medium-sized model, featuring 4.49 million parameters and 22.84 GFLOPs. This model balances computational efficiency with performance and has 4 encoder blocks, and 3 decoder fusion blocks, with skip connection channels configured as $C = 96$. DLKUNet-L is the largest model, with 17.7 million parameters and 29.64 GFLOPs. It is suitable for scenarios where computational resources are less of a concern, and higher performance is desired. DLKUNet-L

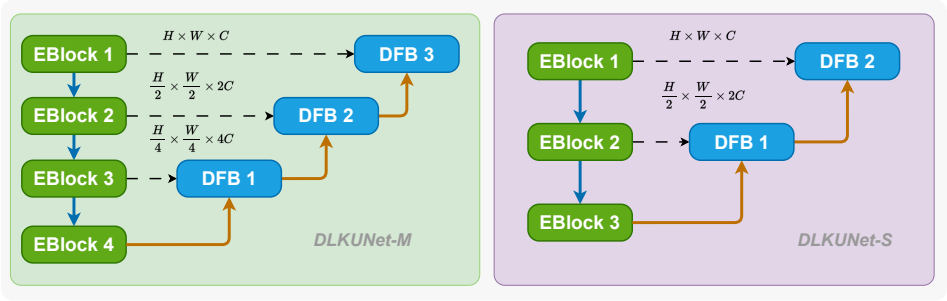


Fig. 4: The architectural configurations of DLKUNet-M and DLKUNet-S

includes 5 encoder blocks,4 decoder fusion blocks, and the channel configuration is $C = 96$.

These three models offer different trade-offs between computational complexity and feature extraction capability, enabling researchers and practitioners to select the appropriate model based on their specific resource limitations and performance requirements. By providing different model sizes, DLKUNet is versatile and adaptable for a wide range of medical image segmentation tasks.

Table 1: Configurations of different DLKUNet models

Methods	Params (M)	FLOPs (G)	EBlock Num	DFB Num	Channels Num
DLKUNet-S	1.13	15.95	3	2	{96, 192}
DLKUNet-M	4.49	22.84	4	3	{96, 192, 384}
DLKUNet-L	17.70	29.64	5	4	{96, 192, 384, 768}

4 Experiments and Results

4.1 Datasets

Multi-Organ Segmentation (Synapse) Dataset: To validate the effectiveness of our method, we employed the widely recognized Synapse dataset [1]. This dataset comprises 30 cases, with a total of 3,779 axial CT slices of the abdomen. Each case contains between 85 and 198 slices, with a spatial resolution ranging from 0.54×0.54 to $0.98 \times 0.98 mm^2$ and pixel resolution of 512×512. The slice thickness varies from 2.5 to 5.0 mm. The dataset covers eight abdominal organs, including the aorta, gallbladder, spleen, left kidney, right kidney, liver, pancreas, and stomach. In alignment with other related works [12, 14], we selected 18 cases from the Synapse dataset, totaling 2,211 longitudinal CT slices as the training dataset, with the remaining 12 cases used for testing.

[1] https://www.synapse.org/Synapse:syn3193805/wiki/217789

Automated Cardiac Diagnosis Challenge (ACDC) Dataset: The ACDC dataset [2] comprises MRI images of cardiac anatomy from different patients, with annotations for the left ventricle (LV), right ventricle (RV), and myocardium (MYO). For the ACDC dataset, we preprocessed the dataset and used 70 training samples, with a total of 1,304 slices for training, and 20 test samples, with 40 slices for testing[12, 14].

4.2 Evaluation Metrics

The Dice Similarity Coefficient (DSC): DSC is a metric used to measure the similarity between two samples, ranging from $[0, 1]$, where 1 represents perfect overlap and 0 represents no overlap[14, 26]. Let A represent the predicted segmentation result, and B represent the corresponding ground truth region. Higher DSC indicates better performance in medical image segmentation. The DSC is given as

$$DSC\left(A, B\right) = \frac{2 \times |A \cap B|}{|A| + |B|} \tag{5}$$

The 95% Hausdorff Distance (HD95): HD is a metric that quantifies the similarity between two sets by measuring the maximum distance between their boundaries[12, 27]. HD95 refers to the $95th$ percentile of the Hausdorff distance. Here, $d\left(s, t\right)$ represents the Euclidean distance between point s and t, while sup and inf denote the supremum and infimum respectively. Lower HD95 indicates better performance in medical image segmentation. The HD95 is given as

$$HD95(S, T) = Percentile_{95}\left(\left\{\inf_{t \in T} d(s, t) \mid s \in S\right\} \cup \left\{\inf_{s \in S} d(t, s) \mid t \in T\right\}\right) \tag{6}$$

4.3 Implementation details

We conducted model training in the PyTorch 2.4.1 environment, using Python version 3.12.4, and the GPU hardware configuration is Nvidia A800. In the training process, no third-party datasets were used for pretraining. We employed the Adam optimizer, with weight decay coefficient set to $1e-5$, input image size of 224×224, and initial learning rate of $1e-3$, and after 200 epochs, the learning rate was adjusted to $1e-5$. Also, a dynamic learning rate adjustment strategy was applied, reducing the learning rate by 90% every 20 epochs, optimizing the model's convergence speed and final performance at different stages of training.

To improve the network's robustness, we applied data augmentation techniques to the training datasets. For both the Synapse and ACDC datasets, we employed data augmentation strategies involving random flipping and random 90° rotation, each applied with a probability of 50%. These techniques effectively enhanced the network's robustness in segmentation tasks, ensuring high accuracy and stability when handling various image transformations and variations.

[2] https://www.creatis.insa-lyon.fr/Challenge/acdc/

4.4 Training Strategies

To balance both the boundary features and accurate segmentation of target regions, the following composite loss function is commonly used to optimize the segmentation performance in medical image segmentation.

$$OrdinaryLoss = DiceLoss + CELoss \tag{7}$$

In the aforementioned strategy for calculating the loss function, DiceLoss refers to the Dice Similarity Coefficient loss function, which is computed as

$$DiceLoss = 1 - \frac{2 \times |P \cap G| + \epsilon}{|P| + |G| + \epsilon} \tag{8}$$

where P represents the predicted segmentation region, G represents the ground truth region, $|P \cap G|$ denotes the intersection between the predicted and ground truth areas, and $|P|$ and $|G|$ represent the number of voxels in the predicted and ground truth areas, respectively. ϵ is used to avoid division by zero.

CELoss represents Cross-Entropy Loss, and for multiclass classification problems like the Synapse or ACDC dataset, it can be expressed as

$$CELoss = -\sum_{i=1}^{C} G_i \log(P_i) \tag{9}$$

where C denotes the number of classes, P_i represents the predicted probability for the class, and G_i is the true label for the class.

Compared with using DiceLoss or CELoss individually, this composite method more effectively balances the learning of edge features and central regions. DiceLoss focuses on optimizing the overlap ratio, ensuring that the network captures the global contour of the target. In contrast, CELoss focuses on pixel-level classification, which is particularly beneficial for identifying small or challenging regions in imbalanced datasets. This combination improves both the segmentation accuracy and robustness of the model. However, when handling regions with varying complexities, such as intricate boundaries versus simpler central areas, the convergence rate of the loss function may differ between these regions. This discrepancy can lead to situations where one part of the data, such as the central regions, has already been well-fitted, while the complex boundaries continue to converge at a much slower rate, ultimately affecting the overall segmentation performance.

To improve the fitting of the loss function, we applied two specific dynamic loss function strategies.

Phase-based Loss Strategy: This strategy adjusts the composition ratio of loss functions and the learning rate based on different stages of the training process. In the early stages of training, global and local features are equally important, so the weights of CELoss and DiceLoss are set equally. In later stages, to refine the segmentation results, the weight of DiceLoss is increased. Simultaneously, the learning rate is adjusted according to the training phase to ensure stable convergence and optimal performance.

Dynamic Loss Strategy: This strategy automatically adjusts the ratio between DiceLoss and CELoss during training and dynamically modifies the loss weights based

on predefined rules. For example, we observed that the model fits global features faster than local features in the ACDC dataset. By applying this method, the model can flexibly adjust the weighting of the loss functions, improving training efficiency and segmentation accuracy.

Both the Phase-based and Dynamic Loss Strategy can be expressed as

$$Loss = \alpha \times DiceLoss + \beta \times CELoss \tag{10}$$

when using a **Phase-based Loss** Strategy, set α and β as

$$(\alpha, \beta) = \begin{cases} (0.6, 0.4), & \text{epochs} \in [1, 200] \\ (0.7, 0.3), & \text{epochs} \in (200, 400] \end{cases} \tag{11}$$

when using **Dynamic Loss** function, set as

$$\alpha = A + (1 - A) \times \left(1 - \frac{1}{1 + e^{-k(x - x_0)}}(epoch)\right) \tag{12}$$

$$\beta = 1 + (B - 1) \times \left(1 - \frac{1}{1 + e^{-k(x - x_0)}}(epoch)\right) \tag{13}$$

here k denotes the slope of the curve, and x_0 represents the midpoint offset value, which is typically set to $0.5 \times epoch$. A and B usually set as 0.3 and 0.7.

In the Synapse segmentation task, Figure 5 illustrates the DSC performance during training when using the DLKUNet-L with OrdinaryLoss and Phase-Based Loss (ours). The x-axis represents the number of training epochs, and the y-axis shows the DSC on the test set. As shown in the figure, the model trained with Phase-Based Loss (red curve) exhibited better convergence and higher segmentation accuracy throughout the training process. In contrast, the model with OrdinaryLoss (blue curve) showed slower convergence in the early training stages and exhibited greater fluctuation in the later stages. The Phase-Based Loss strategy significantly outperformed the OrdinaryLoss strategy in the Synapse dataset, improving segmentation capability and accelerating convergence.

Similarly, in the ACDC dataset, Figure 6 shows the scoring performance of PhaseBasedLoss and DynamicLoss strategies during training when using DLKUNet-L. The x-axis represents the number of training epochs. As shown in the figure, the red curve, which represents the Dynamic Loss strategy that continuously adjusts the loss function in real-time, demonstrates faster convergence and a more stable DSC compared to the blue curve representing PhaseBasedLoss.

From these two sets of experimental results, it can be concluded that the Phase-Based Loss strategy performs better in handling the complex Synapse multi-organ segmentation task, while the Dynamic Loss strategy is more advantageous in the ACDC segmentation task. These strategies significantly improve the model's convergence speed and final performance.

4.5 Experiment Results

As shown in Table 2 and Figure 7, we conducted a comparison of our method against other approaches on the Synapse segmentation task. We evaluate the segmentation

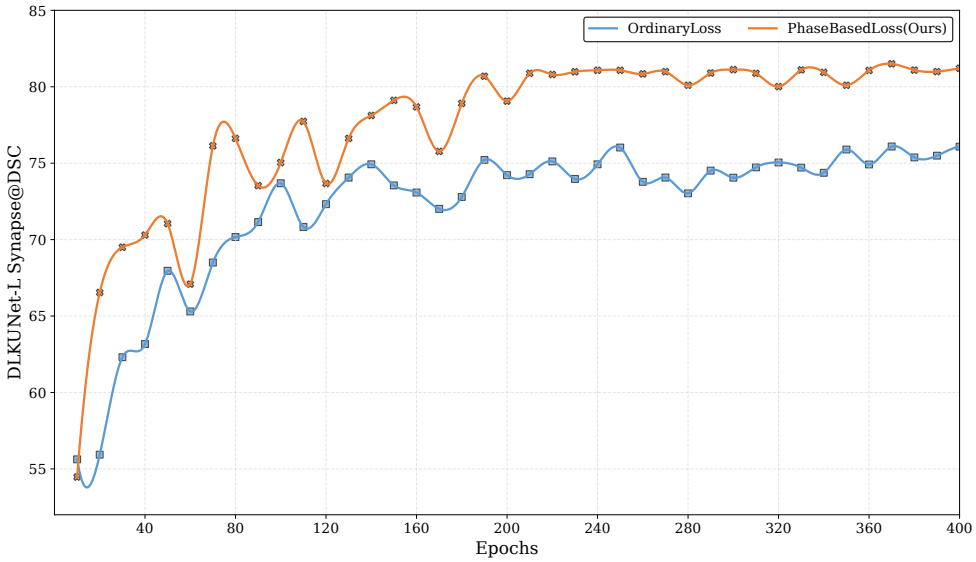


Fig. 5: DSC scores obtained from segmenting the Synapse dataset using DLKUNet-L under different loss functions. The blue line represents the OrdinaryLoss method, while the red line denotes the Phase-Based Loss method.

performance on eight organs and analyze the parameter count and computational complexity of each method.

Although Swin-Unet [14] and U-Net [2] utilize similar U-shaped hierarchical designs that combine encoder and decoder modules to enhance feature extraction, the DLKUNet architecture demonstrates superior segmentation performance across multiple organs. DLKUNet-L achieves a 12.12% improvement in DSC score compared to Swin-Unet for pancreas segmentation, while maintaining comparable performance in liver segmentation. Additionally, it demonstrates approximately 2% improvement in DSC score for other organs [14]. DLKUNet-M also demonstrates significant advantages, achieving the highest score in gallbladder segmentation, it represents an 8.8% improvement over Swin-Unet. Even the smallest model outperforms Swin-Unet, achieving a DSC score of 79.74 compared to the Swin-Unet score of 79.13.

Our method excels in the DSC score, and also it shows superior performance in measuring the quality of segmentation boundaries. DLKUNet-L models outperformes Swin-Unet [14] in HD95, with improving boundary precision by 35%. This demonstrates the superior ability of our model in edge detection and fine feature extraction.

In addition to achieving outstanding segmentation results, our method also demonstrates significant advantages in terms of parameter efficiency. The DLKUNet-S uses only 4.1% of the parameters of Swin-Unet [14], and achieves a 0.7% higher DSC score. The DLKUNet-M uses 16.52% of the parameters of Swin-Unet and achieves a 2.1% higher DSC score. The DLKUNet-L, utilizing 65% of Swin-Unet parameters, surpasses

Table 2: Comparative results of various methods in the Synapse segmentation task. The parameter count is measured in Million (M), and FLOPs are calculated in Gigaflops (G). Higher DSC indicates better performance, while lower HD95 indicates better performance. Evaluation on Aorta(Aor), Gallbladder(Gal), Left Kidney(Kid(L)), Right Kidney(Kid(R)), Liver(Liv), Pancreas(Pan), Spleen(Spl), Stomach(Sto) Red highlights the best results, while Blue indicates the second-best results.

Method	**Params(M)**	**FLOPs(G)**	**DSC**	**HD95**	**Aor**	**Gal**	**Kid(L)**	**Kid(R)**	**Liv**	**Pan**	**Spl**	**Sto**
R50 U-Net[12]	147.80	41.09	74.68	36.87	87.74	63.66	80.60	78.19	93.74	56.90	85.87	74.16
U-Net[2]	34.52	65.52	76.85	39.7	89.07	69.37	77.77	68.60	93.43	53.98	86.67	75.58
Att-U-Net[10]	34.87	66.63	77.77	36.2	89.55	68.88	77.98	71.11	93.5	58.04	87.30	75.75
TransUNet[12]	96.07	88.91	77.48	32.69	87.23	63.16	81.87	77.02	94.09	55.86	85.08	75.62
Swin-Unet[14]	27.17	**6.16**	79.13	21.55	85.47	66.53	83.28	79.61	94.29	56.58	90.66	76.6
DLKUNet-S	**1.13**	15.95	79.74	28.95	87.46	66.27	84.15	76.06	92.74	64.72	90.38	76.17
DLKUNet-M	4.49	22.84	80.82	22.78	88.14	72.39	85.20	81.13	92.88	60.46	90.79	75.54
DLKUNet-L	17.70	29.64	**81.21**	**13.89**	87.39	67.4	87.64	83.47	92.45	63.44	90.99	76.92

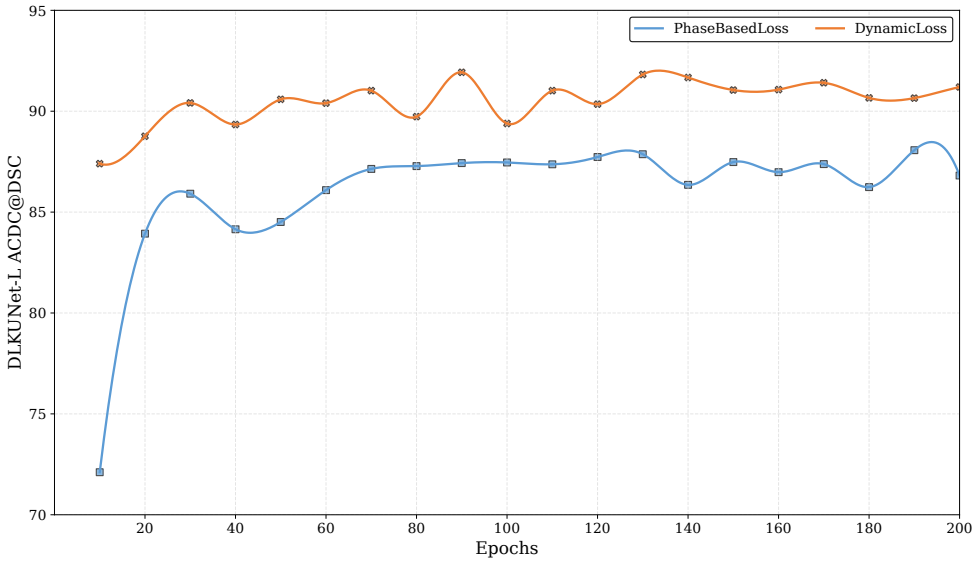


Fig. 6: DSC scores obtained from segmenting the ACDC dataset using DLKUNet-L under different loss functions. The blue line represents the use of the Phase-Based Loss strategy, while the red line denotes the use of the Dynamic Loss strategy.

it by 2.6% in the DSC score. Compared to traditional convolutional models such as R50 U-Net [12], our improvement is even more notable, requiring less than 1% of the parameter count.

Similar to the segmentation task on the Synapse dataset, we also evaluated our method on the ACDC dataset, using the DSC as the evaluation metric. As shown in Table 3 and Figure 8, DLKUNet-L achieves a DSC score of 91.93, surpassing Swin-Unet [14] by 2.14% while using only 65% of its parameter count. Moreover, the DLKUNet-M achieves a DSC score of 91.74 with a parameter count of 4.49M, this demonstrates excellent efficiency and segmentation performance. Although the DLKUNet-S has a parameter count of only 1.13M, it still achieves a DSC score of 91.71, surpassing Swin-Unet.

Our method demonstrates outstanding generalization capabilities and robustness through testing on the Synapse and ACDC datasets, maintaining excellent performance across various medical image segmentation tasks.

5 Ablation Study

To further investigate the factors influencing network performance, we conducted ablation experiments on the Synapse dataset.

Effect of Channels Count: Figure 9 presents an ablation study examining the effects of varying skip connection settings on the performance of the DLKUNet architecture. Specifically, the experiment evaluates DLKUNet-S, DLKUNet-M, and

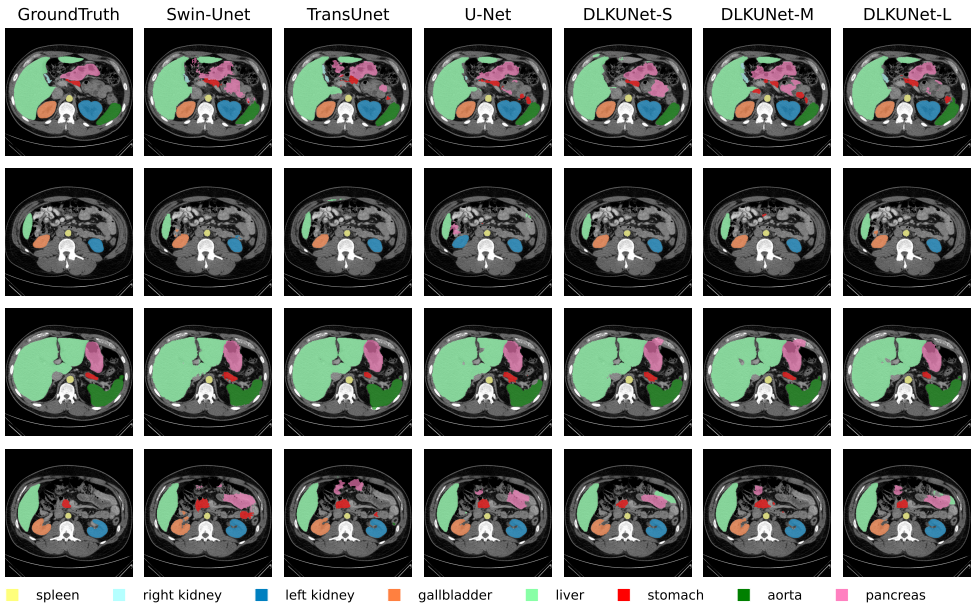


Fig. 7: Visualization results obtained using different methods on the Synapse dataset.

Table 3: Comparison of results for the ACDC task using different methods in terms of parameters (M) and DSC. Higher DSC scores indicate better segmentation performance.

Method	Params (M)	DSC	RV	Myo	LV
R50 U-Net[2]	147.80	87.55	87.1	80.63	94.92
R50 Att-U-Net[10]	34.87	86.75	87.58	79.20	93.47
R50 ViT[11]	97.96	87.57	86.07	81.88	94.75
TransUNet[12]	96.07	89.71	88.86	84.53	95.73
Swin-Unet[14]	27.17	90	88.55	85.62	95.83
DLKUNet-S	**1.13**	91.71	**89.82**	89.56	95.75
DLKUNet-M	4.49	91.74	89.48	89.94	95.81
DLKUNet-L	17.70	**91.93**	89.52	**90.19**	**96.09**

DLKUNet-L model configurations, each employing three different configurations of channel sizes.

For the small model, three configurations were tested with channel settings of C = 64, 96, 128 in the y-axis. Similarly, the medium model and large model employ skip connections with channel configurations of C = 64, 96, 128. In total, nine model configurations were tested, producing eighteen experimental results, as represented in the heatmaps.

We conducted DSC score and HD95 comparisons on the Synapse dataset for DLKUNet-S, DLKUNet-M, and DLKUNet-L model configurations, and the input

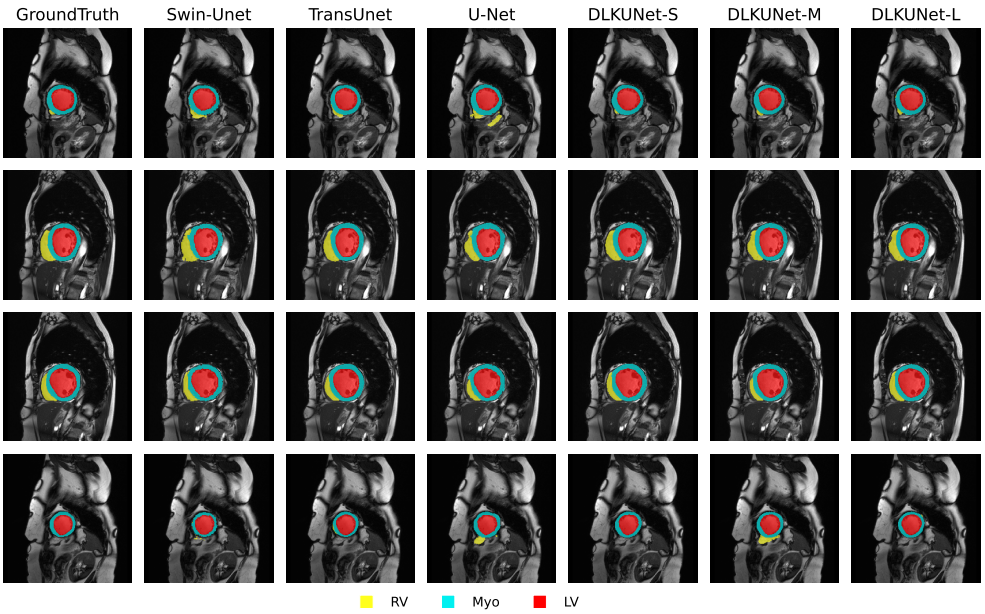


Fig. 8: Segmentation results obtained using different methods on the ACDC task.

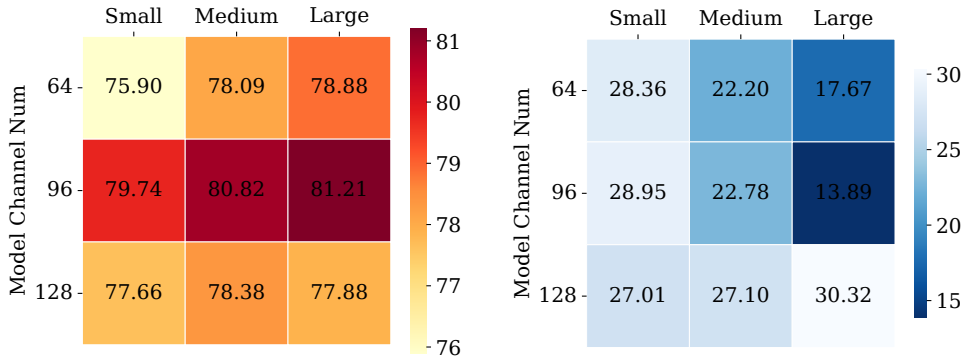
image size is 224×224. The experimental results show significant differences in segmentation performance across different channel counts and model sizes, as shown in Figure 9. When the channel count is 96, the model achieved the best balance between the DSC score and HD95, especially in the DLKUNet-L model, where it achieves a DSC score of 81.21 and maintains good boundary precision ($13.89mm$).

Effect of Input Image Size: To evaluate the impact of input image resolution on segmentation performance, we conducted experiments on the Synapse dataset using the DLKUNet-L model. Specifically, we tested two input sizes: 224×224, 512×512 and recorded the DSC score for each resolution.

The results showed in Table 4 that increasing the input resolution led to a slight overall improvement in DSC, with a gain of 0.3%. Notably, the score for certain organs like the Aorta improved by 4.42%, while others, such as the Stomach, experienced a decrease of approximately 10%. Additionally, increasing the input image size significantly raised the computational cost, with the floating point operations (FLOPs) increasing from $29.64G$ to $154.86G$. Therefore, the input size is set 224×224 can provide a favorable balance between segmentation accuracy and computational resource consumption.

Effect of Kernel Size: To investigate the effect of different kernel sizes on both the convolution part and the attention part, we designed several variations and tested them on the Synapse with an input size of 224×224. Specifically, we experimented with four configurations listed in the Table 5.

Where DLKUNet-5DW and DLKUNet-7DW model replaced the 3×3 kernels of depthwise convolution in the DLKUNet-L model with 5×5 and 7×7 kernels,



(a) DSC Score Heatmap

(b) HD95 Score Heatmap

Fig. 9: DSC scores (left) and HD95 scores (right) for models of different sizes under various channel configurations on the Synapse dataset.

Table 4: The impact of input datasets of different resolutions on DLKUNet-L and segmentation results in DSC score

Size	DSC	Aor	Gal	Kid(L)	Kid(R)	Liv	Pan	Spl	Sto
224 × 224	81.21	87.39	67.40	87.64	83.47	92.41	63.44	90.99	76.92
512 × 512	**81.47**	**91.08**	**70.07**	87.16	84.30	**94.48**	**65.39**	90.59	68.68

Table 5: Impact of different convolutional kernel configurations on the segmentation results using DLKUNet-L

Models	Depthwise Kernel Size	Large Kernel Size	DSC
DLKUNet-L	3 × 3	5 × 5, 7 × 7	**81.21**
DLKUNet-5DW	5 × 5	5 × 5, 7 × 7	80.61
DLKUNet-7DW	7 × 7	5 × 5, 7 × 7	80.62
DLKUNet-LKA1	3 × 3	3 × 3, 7 × 7	80.68
DLKUNet-LKA2	3 × 3	5 × 5, 5 × 5	79.25

respectively. DLKUNet-LKA1 and DLKUNet-LKA2 replaced the depthwise separable convolution in the attention part with a 3×3 kernel and a dilated convolution kernel of size 5×5. The experimental results indicated that depthwise separable convolutions with kernel sizes of 3×3 and 5×5 achieved the best DSC scores on the Synapse dataset, while dilated convolutions performed optimally with a kernel size of 7×7.

6 Discussion

In this paper, we propose DLKUNet, which demonstrates satisfactory overall performance in the task of multi-organ abdominal segmentation. However, as shown in

Figure 10, the performance in Pancreas segmentation is suboptimal, with DSC scores below 65, indicating significant limitations in handling such a complex organ. This difficulty may be attributed to the complex anatomical structure, irregular shape, and significant variability in the position of the Pancreas, making it challenging to accurately identify and segment in medical images. Additionally, the low contrast between the Pancreas and surrounding tissues further complicates the segmentation process, revealing the limitations of current models when dealing with organs that are morphologically complex and have fuzzy boundaries. To enhance segmentation performance for the pancreas, future research could incorporate more advanced data augmentation techniques, such as random rotations, translations, and scaling, to improve model robustness. Additionally, utilizing more annotated medical image datasets or adopting transfer learning strategies could further enhance segmentation accuracy. These methods could help models better handle organ variability and irregularity, improving segmentation accuracy for challenging tasks.

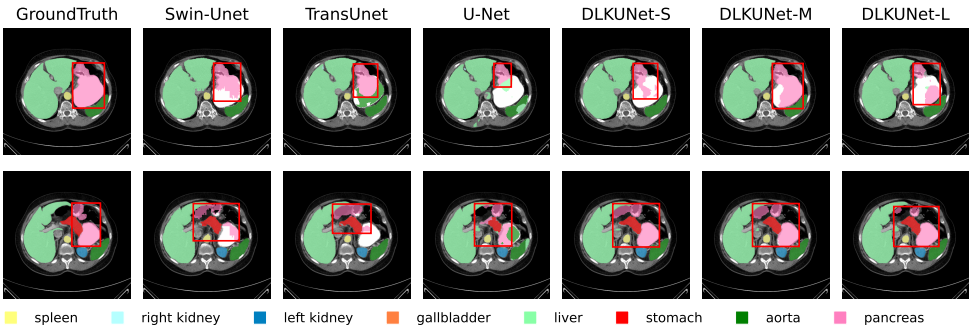


Fig. 10: Segmentation Results for Pancreas Using Different Methods

7 Conclusion

In this study, we introduced an innovative method for medical image segmentation. The DLKUNet significantly improves segmentation accuracy through effective multi-scale feature extraction, enhancing the model's ability to capture key features across different scales. We also designed several training optimization strategies, involving rational adjustments to loss function weights and dynamic learning rates, which accelerated model convergence and significantly improved training efficiency and accuracy. We conducted extensive experiments on two public medical imaging datasets, Synapse and ACDC. Results demonstrate that DLKUNet showed significant improvements in segmentation accuracy and parameter efficiency, validating the model's exceptional performance across different datasets and tasks. In summary, DLKUNet shows outstanding performance in the field of medical image segmentation, with high parameter efficiency and computational performance, offering extensive support for future medical image analysis and clinical applications.

References

[1] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)

[2] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-assisted intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, pp. 234–241 (2015). Springer

[3] Aitken, K., Ramasesh, V., Cao, Y., Maheswaranathan, N.: Understanding how encoder-decoder architectures attend. Advances in Neural Information Processing Systems **34**, 22184–22195 (2021)

[4] Milletari, F., Navab, N., Ahmadi, S.-A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571 (2016). Ieee

[5] Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J.: Unet++: A nested u-net architecture for medical image segmentation. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4, pp. 3–11 (2018). Springer

[6] Lou, A., Guan, S., Loew, M.: Dc-unet: rethinking the u-net architecture with dual channel efficient cnn for medical image segmentation. In: Medical Imaging 2021: Image Processing, vol. 11596, pp. 758–768 (2021). SPIE

[7] Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.-W., Wu, J.: Unet 3+: A full-scale connected unet for medical image segmentation. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1055–1059 (2020). IEEE

[8] Chen, S., Bortsova, G., García-Uceda Juárez, A., Van Tulder, G., De Bruijne, M.: Multi-task attention-based semi-supervised learning for medical image segmentation. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part III 22, pp. 457–465 (2019). Springer

[9] Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)

[10] Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., et al.: Attention u-net: Learning

where to look for the pancreas. arXiv preprint arXiv:1804.03999 (2018)

[11] Dosovitskiy, A.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

[12] Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y.: Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 (2021)

[13] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)

[14] Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M.: Swin-unet: Unet-like pure transformer for medical image segmentation. In: European Conference on Computer Vision, pp. 205–218 (2022). Springer

[15] Huang, X., Deng, Z., Li, D., Yuan, X., Fu, Y.: Missformer: An effective transformer for 2d medical image segmentation. IEEE Transactions on Medical Imaging **42**(5), 1484–1494 (2022)

[16] Zheng, W., Lu, S., Yang, Y., Yin, Z., Yin, L.: Lightweight transformer image feature extraction network. PeerJ Computer Science **10**, 1755 (2024)

[17] Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2017)

[18] Howard, A.G.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)

[19] Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13733–13742 (2021)

[20] Trockman, A., Kolter, J.Z.: Patches are all you need? arXiv preprint arXiv:2201.09792 (2022)

[21] Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 568–578 (2021)

[22] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11976–11986 (2022)

[23] Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., *et al.*: Mlp-mixer: An all-mlp architecture for vision. Advances in neural information processing systems **34**, 24261–24272 (2021)

[24] Tang, F., Wang, L., Ning, C., Xian, M., Ding, J.: Cmu-net: a strong convmixer-based medical ultrasound image segmentation network. In: 2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI), pp. 1–5 (2023). IEEE

[25] Guo, M.-H., Lu, C.-Z., Liu, Z.-N., Cheng, M.-M., Hu, S.-M.: Visual attention network. Computational Visual Media **9**(4), 733–752 (2023)

[26] Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Jorge Cardoso, M.: Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3, pp. 240–248 (2017). Springer

[27] Huttenlocher, D.P., Klanderman, G.A., Rucklidge, W.J.: Comparing images using the hausdorff distance. IEEE Transactions on pattern analysis and machine intelligence **15**(9), 850–863 (1993)