LoG-VMamba 🐍: Local-Global Vision Mamba for Medical Image Segmentation

Trung DQ. Dang, Huy Hoang Nguyen, and Aleksei Tiulpin

University of Oulu, Finland
`{trung.ng,huy.nguyen,aleksei.tiulpin}@oulu.fi`

Abstract. Mamba, a State Space Model (SSM), has recently shown competitive performance to Convolutional Neural Networks (CNNs) and Transformers in Natural Language Processing and general sequence modeling. Various attempts have been made to adapt Mamba to Computer Vision tasks, including medical image segmentation (MIS). Vision Mamba (VM)-based networks are particularly attractive due to their ability to achieve global receptive fields, similar to Vision Transformers, while also maintaining linear complexity in the number of tokens. However, the existing VM models still struggle to maintain both spatially local and global dependencies of tokens in high dimensional arrays due to their sequential nature. Employing multiple and/or complicated scanning strategies is computationally costly, which hinders applications of SSMs to high-dimensional 2D and 3D images that are common in MIS problems. In this work, we propose Local-Global Vision Mamba, LoG-VMamba, that explicitly enforces spatially adjacent tokens to remain nearby on the channel axis, and retains the global context in a compressed form. Our method allows the SSMs to access the local and global contexts even before reaching the last token while requiring only a simple scanning strategy. Our segmentation models are computationally efficient and substantially outperform both CNN and Transformers-based baselines on a diverse set of 2D and 3D MIS tasks. The implementation of LoG-VMamba is available at https://github.com/Oulu-IMEDS/LoG-VMamba.

Keywords: Semantic Segmentation · State Space Models · Medical Imaging

1 Introduction

Medical image segmentation (MIS) targets the delineation and location of tissues and lesions in 2D or 3D medical images. This process is crucial for developing automatic disease identification, staging, and treatment, as well as developing medical robotics. In recent years, the state-of-the-art approaches to MIS have been based on deep learning (DL), thanks to its ability to learn representations of complex patterns from large datasets. This proves to be essential in producing production-quality performances in medical applications [6,17,18,44,51,58].

Visual feature extraction plays a vital role in Computer Vision (CV) tasks, including MIS. In the early stage of DL, convolutional neural network (CNN)
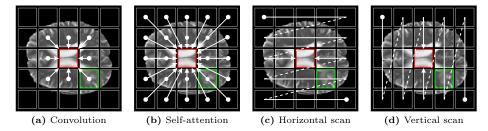
| **(a)** Convolution | **(b)** Self-attention | **(c)** Horizontal scan | **(d)** Vertical scan |

Fig. 1: The comparison between how feature extractors establish a correlation between the query token (in red) and a neighboring token (in green). In (c-d) [35], the distance between the query and its neighbor may be roughly one row (or column) of tokens.

was the ubiquitous feature extraction module, since this architecture is effective in learning local patterns by taking into account the regional context around pixels [19, 22, 37]. However, one drawback of CNN is the difficulty in capturing long-range dependencies (LRD), which is essential for extracting high-level features from data [29]. To enable LRD capturing (i.e. increasing the receptive field), some studies stacked a large number of layers with downsampling [31], while others used dilated convolutions [54]. However, the former is computationally expensive and the latter may result in missing fine-grained details.

In contrast to CNNs, Vision Transformer (ViT)-based architectures rely on the attention mechanism to learn LRD across the entire image [11, 57]. Despite their global receptive field (GRF) at all levels of learned representations, ViTs are hindered by their quadratic complexity with respect to the number of tokens. In the case of ViTs, these tokens are the patches into which an image is split before it is fed into the network. Therefore, utilizing ViTs in tasks with high-dimensional inputs and outputs such as semantic segmentation may create computational efficiency issues [35].

Recently, Mamba [12], a State Space Model (SSM)-based method, has been introduced in the field of Natural Language Processing (NLP). Mamba enables the acquisition of GRF via its recurrent mechanism, while maintaining computational efficiency due to the linearity of complexity w.r.t. data dimension. Several studies have been conducted to adapt Mamba to CV tasks such as image classification and MIS [34, 38, 52, 59]. Despite the computational benefits, in vision applications [35], Mamba struggles to maintain dependencies between both neighboring and distant tokens due to the restricted capacity of Mamba's state [46] and the necessity of a sequential approach to model LRD. Fig. 1 demonstrates the weakness of two common scanning protocols of Vision Mamba (VM)-based methods [34,35,52] compared to convolutions and ViTs. Some studies attempted to maintain the locality of neighboring tokens using a complicated scanning strategy such as a multidirectional or omnidirectional scan [52, 56]. However, these significantly increase the computational complexity.

In this work, we argue that developing more complex scanning strategies is unnecessary, should there be an image-friendly tokenizer. As such, we propose the

Local-Global Vision Mamba (LoG-VMamba), whose core components are Local Token eXtractor (LTX) and Global Token eXtractor (GTX). LTX maintains the locality of neighboring tokens in high-dimensional arrays, explicitly ensuring that spatially nearby tokens remain close along the channel axis. GTX, on the other hand, compresses features across all spatial dimensions, providing the SSM module with compressed versions of GRF before reaching the final time step. The combined effects of the two components make LoG-VMamba significantly distinct from prior VM-based methods as we eliminate the need for a sophisticated scanning strategy. In summary, our study has the following contributions:

1. We propose two visual token extractors, LTX and GTX, that provide the SSMs with both local and global visual contexts at early time steps, inspired by the strengths of both CNN and ViT.
2. We propose LoG-VMamba, a Mamba-based module for CV, leveraging the power of both LTX and GTX. Based on LoG-VMamba, we introduce segmentation models for 2D and 3D MIS problems.
3. Our experiments demonstrate that the developed models consistently outperform multiple well-known baselines in a variety of 2D and 3D MIS benchmarks. We also show that our approach does not require an advanced scanning strategy to achieve state-of-the-art performance, and is thus computationally efficient.

2 Related Work

Feature Extraction. The evolution of feature extraction modules has been central to the field of CV. After the success of classic CNNs such as AlexNet [30] and VGG [47], learnable feature extractors have gained widespread adoption and a plethora of improvements to CNN design have been developed [7, 19–21, 54]. These works have improved the performance of CNNs on various CV tasks, and they showcased the capabilities of CNNs to learn local patterns at low-level layers, as well as hierarchical representations at high-level layers. Afterward, inspired by the success of Transformers [49] in sequence modeling, ViT [11] emerged and challenged the dominance of CNNs in vision-related problems. ViT tokenizes an image along spatial dimensions into patches and employs multi-head self-attention (MSA) over the obtained sequence of tokens. Thanks to MSA, ViT is able to handle LRD at the early layers of a vision model. Numerous subsequent works [10, 16, 36, 48, 55] enhanced various aspects of ViT, such as efficiency and refined attention mechanisms. Moreover, some studies [8, 15, 33, 42] aimed to combine the desirable properties of both CNNs and ViTs.

State Space Models. Inspired by the principles from control theory, SSMs [12–14] have emerged as an alternative to recurrent neural networks [41] and Transformers [49] in NLP and sequence modeling. To adapt these models into DL, prior studies [12, 14] employed the zero-order hold with a timescale parameter to transform the operations of SSMs into discrete form. Mamba [12] let these

parameters of SSMs be input-dependent and designed a hardware-aware algorithm to boost the throughput. A plethora of works [24,35,46,50,56,59] explored the viability of SSMs in visual feature extraction. Most of these [24, 35, 56, 59] focused on designing suitable scanning strategies for processing images. Instead of finding how to *scan* the tokens, we focus on how to *build* informative visual tokens embedded with local-global dependencies.

Medical Image Segmentation. Progress in segmentation, particularly MIS, has been built upon the advancements in representation learning. U-Net [44] is the most prominent early representative of a U-shaped architecture, which influenced the design of nearly all modern MIS approaches, and it consists of an encoder and a decoder connected at each feature scale via skip connections. While the traditional U-Net uses pooling and transposed convolution layers to change the feature scale, its pure Transformer variant [6] replaces them with patch merging and patch expanding layers. Afterward, the trend of following this architecture [17, 18, 32, 51] has persisted in MIS until the recent emergence of Mamba-based models such as U-Mamba [38] and Swin-UMamba [34]. These Mamba-inspired architectures have demonstrated advances in both computational efficiency and effectiveness when compared to CNN and Transformer-based methods in MIS. In this study, we further enhance the performance of these two methods using our LoG-VMamba module in 2D and 3D MIS.

3   Methodology

3.1   Preliminaries

State Space Model. Time-continuous SSMs [26] refer to linear-time-invariant systems formulated as follows

$$\mathbf{h}'_t = \mathbf{A}\mathbf{h}_t + \mathbf{B}\mathbf{x}_t \tag{1}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{h}_t, \tag{2}$$

where $\mathbf{x}_t$ and $\mathbf{y}_t$ are D-dimensional input and output vectors, respectively. $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are learnable parameters independent of $\mathbf{x}_t$.

Selective State Space Model. Selective SSM (S6) [12] consists of a hidden state $\mathbf{h}_t$, three continuous parameters $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ as well as a step size parameter Δ. Among these learnable parameters, $\mathbf{B}$, $\mathbf{C}$, and Δ depend on $\mathbf{x}_t$, making S6 distinct from prior SSMs. Let $\{\mathbf{x}_t\}_{t\in[L]}$ represent an input sequence of row vectors $\mathbf{x}_t \in \mathbb{R}^{1\times D}$, where $[L] = \{1,\ldots,L\}$. As the objective was to make S6 work on discrete sequences of tokens (e.g. text) [12], we have to discretize the continuous parameters $\mathbf{A}$ and $\mathbf{B}$ using discretization functions f_A and f_B, that is $\overline{\mathbf{A}} = f_A(\Delta, \mathbf{A})$ and $\overline{\mathbf{B}} = f_B(\Delta, \mathbf{B})$, respectively. Finally, we can compute the hidden state $\mathbf{h}_t$ and the output $\mathbf{y}_t$ as follows

$$\mathbf{h}_t = \overline{\mathbf{A}}\mathbf{h}_{t-1} + \overline{\mathbf{B}}\mathbf{x}_t \tag{3}$$
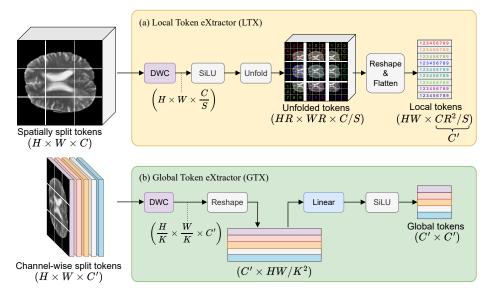
$$\mathbf{y}_t = \mathbf{C}\mathbf{h}_t. \tag{4}$$

Fig. 2: Local and global token extractors. DWC indicates a depthwise convolutional layer. S and K correspond to the depthwise and spatial compression in the DWCs of (a) and (b), respectively.

As S6 is our focus, the term SSM hereinafter refers specifically to the S6 block in [12].

Visual State Space (VSS) [35] is an extension of the Mamba block [12] for visual data (see Fig. 3). Firstly, the input feature map $\mathbf{x}$ is normalized and projected with an expansion factor α into two tensors going through two branches. In the first branch, one tensor is passed through a depthwise convolutional layer (DWC) and an activation function before getting processed by the SSM module in [12]. To enable SSM on 2D data, several scanning strategies are needed to convert a 2D array of tokens into a 1D sequence. Their required computational resources are proportional to the number of scanning directions M. After that, the output of the first branch is multiplied by that of the second branch, which has an activation function after the projection. The product is subsequently projected and added to the input $\mathbf{x}$, resulting in the output of this block.

3.2 Local-Global Token Extractors

Local Token Extractor. As Mamba is a sequence modeling module, several scanning strategies [24,35,56,59] have been introduced to transform 2D arrays of tokens into 1D sequences. However, their common pitfall, as illustrated in Figs. 1c and 1d, is that they fail to maintain the spatial proximity between neighboring tokens. Meanwhile, the local dependencies are important to visual tasks [24,36]. To improve the modeling of local features without resorting to an omnidirectional scan as in [56], we introduce the LTX module, graphically illustrated in Fig. 2a.
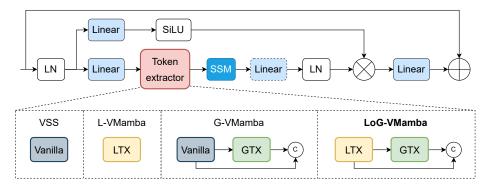
Fig. 3: LoG-VMamba and its simpler versions compared to the vanilla VSS [35]. LN and SSM mean layer normalization [2] and the S6 block in [12], respectively. Vanilla indicates the module consisting of a DWC layer and SiLU followed by a reshaping operator. The linear block after SSM is only needed in L-VMamba and LoG-VMamba. White blocks indicate modules without learnable parameters. $\oplus, \otimes$, and $\copyright$ represent element-wise addition, multiplication, and concatenation, respectively. The SSM block in our settings performs only 1 horizontal scan.

Specifically, we first utilize a DWC to squeeze the input channels C by a factor S, which allows us to avoid the computational overhead of later operators. After passing the compressed tokens through SiLU, we use a fixed convolution kernel of size $R \times R$ to unfold the tokens [23]. This operator allows us to replicate tokens and preserve the spatial relations of nearby tokens. Finally, we reshape and flatten the spatial dimensions to form a 1D sequence of local tokens, ensuring that neighboring tokens within the local window are along the channel axis of the query token. This spatial flattening is performed in a row-by-row manner, conceptually equivalent to the horizontal scan in Fig. 1c. As a result, the number of output channels is $C' = \frac{CR^2}{S}$.

Global Token Extractor. In addition to the locality maintained by LTX, we propose the GTX module, demonstrated in Fig. 2b, to produce global (i.e. spatial-independent channel-wise) tokens. This module allows the SSM to access compressed versions of GRF at early time steps. Such an approach differs from the selective scan in prior VMs [24, 35, 56, 59], where only the token at the last time step has the context of all other tokens. Specifically, given an input feature map with dimensions $H \times W \times C'$, we spatially compress it using a dilated DWC with a stride of $K \times K$, and then flatten its spatial dimensions to produce global tokens of shape $C' \times \frac{HW}{K^2}$, with the channel dimension and spatial dimensions transposed. For computational efficiency, GTX merely compresses each group of γ input channels throughout all spatial dimensions into global tokens. The role of these steps is to learn an approximation of the global context in each input channel, and thus the concern of losing fine-grained details associated with dilated DWC is insignificant. Subsequently, we utilize a linear layer to project these tokens to a C'-dimensional space and apply the activation function SiLU.

Choosing the number of features as C' enables us to concatenate the output of the GTX with the spatial tokens of the LTX of C' channels in subsequent steps.

3.3 Local-Global Vision Mamba

We generalize the VSS block and incorporate the proposed LTX and GTX to introduce upgraded versions of VMamba, as graphically demonstrated in Fig. 3. As such, the *vanilla* block consisting of a DWC layer and SiLU in VSS is treated as a token extraction module, leading us to develop the following Mamba-based models.

Local Vision Mamba. For Local Vision Mamba (L-VMamba), we employ the introduced LTX block as the token extractor. Compared to VSS, L-VMamba includes the unfolding operator to effectively guarantee the spatial proximity of neighboring tokens in 2D or 3D arrays. We set the window size $R = 3$, which is a common kernel size of convolutional layers. Since the number of channels is changed to C' in the LTX, a linear layer is inserted after the SSM to project the tokens to the original C-dimensional space.

Global Vision Mamba. In Global Vision Mamba (G-VMamba), we combine the vanilla block, comprising a DWC layer and SiLU followed by a flattening operator, from VSS with the GTX block. After being processed by the vanilla block, the feature maps are passed through the GTX module to produce tokens with a GRF. The tokens produced by both modules are eventually combined and forwarded to the next SSM. As the number of channels in the outputs is unchanged while the sequence length increases, the linear layer after the SSM is unnecessary.

Local-Global Vision Mamba. Ultimately, we couple the proposed modules, LTX and GTX, to create the LoG-VMamba module. This combination leverages the local dependencies of LTX and the GRF of GTX, thus harnessing the strengths of both. Precisely, given an input feature map $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$, the token extractor of LoG-VMamba is formulated as

$$\mathbf{x}^L = \text{LTX}(\mathbf{x}), \qquad \mathbf{x}^G = \text{GTX}(\mathbf{x}^L) \tag{5}$$

$$\mathbf{x}^{LG} = \text{Concat}(\mathbf{x}^G, \mathbf{x}^L), \tag{6}$$

where $\mathbf{x}^L \in \mathbb{R}^{HW \times C'}$, $\mathbf{x}^G \in \mathbb{R}^{C' \times C'}$, $\mathbf{x}^{LG} \in \mathbb{R}^{(HW+C') \times C'}$, and $C' = \frac{CR^2}{S}$. Similar to L-VMamba, a linear layer after the SSM module is needed to map the output $\mathbf{x}^{LG}$ to a C-dimensional space.

Due to the sequential and input-dependent nature of Mamba, the concatenation between $\mathbf{x}^L$ with $\mathbf{x}^G$ is not trivial in how to harness both types of contexts. Therefore, we evaluate the following approaches:

- Head: Concatenating the global tokens at the beginning of the sequence.
- Middle: Placing the global tokens in the middle of the sequence.
- Split: Dividing the global tokens into two halves and appending them to both ends of the sequence.
- Interleaved: Inserting each global token between the local tokens at fixed intervals over the sequence, with excessive global tokens at the beginning.

Table 1: Configurations of the experimental setup for different datasets

Dataset	Image size	Batch size	Weight decay	Stages	Pooling	K	S
Endoscopy	640×384	8	$5 \cdot 10^{-2}$	4	5×5	2×2	8
Cell	512×512	8	$5 \cdot 10^{-2}$	4	5×5	2×2	8
BraTS	$128 \times 128 \times 128$	2	10^{-4}	5	$5 \times 5 \times 5$	$2 \times 2 \times 2$	16
ACDC	$16 \times 256 \times 224$	2	10^{-4}	6	$3 \times 5 \times 5$	$2 \times 4 \times 4$	16

3.4   Medical Image Segmentation Models

With the proposed Mamba-based modules, we present our segmentation models for both 2D and 3D medical imaging data. For 2D segmentation, we build our model on top of Swin-UMamba† [34]. Different from Swin-UMamba, the decoder of Swin-UMamba† comprises VSS blocks instead of CNNs. As empirically shown in [34], the pre-trained weights in its encoder are of significant importance, we thus retain the original encoder. In the decoder, we replace the original VSS blocks with our LoG-VMamba blocks. For 3D segmentation models, we modify a version of U-Mamba-Enc [38]. In the encoder, we replace the Mamba blocks in the original network with the LoG-VMamba blocks. The decoder remains unchanged as it does not incorporate any Mamba modules. Following the methods in [34,38], we tailor our models accordingly for each dataset. The details of the 2D and 3D segmentation models are graphically depicted in Fig. S1.

4   Experiments

4.1   2D Datasets

We followed prior works [34,38] and employed these two 2D segmentation datasets for our experiments: Endoscopy from the MICCAI 2017 EndoVis Challenge [1] and Cell from the NeurIPS 2022 Cell Segmentation Challenge [39].

   Endoscopy contained 1800 training and 1200 test images. We further split the training set into 1440 and 360 samples for training and validation, respectively. Its objective was to segment seven instruments: prograsp forceps, needle driver, monopolar curved scissors, bipolar forceps, cadiere forceps, vessel sealer, and drop-in ultrasound probe.

   Cell consisted of 1000 and 101 samples for training and evaluation, respectively. In addition, we divided the former into two portions of size 800 and 200 for training and validation, respectively. We performed both semantic and instance segmentation of cells on this dataset.

4.2   3D Datasets

We conducted experiments on two 3D segmentation datasets: BraTS 2020 [3,4, 40] and ACDC [5].

   BraTS consisted of multi-modal magnetic resonance (MR) images collected from 369 subjects. We split this dataset into training, validation, and test sets
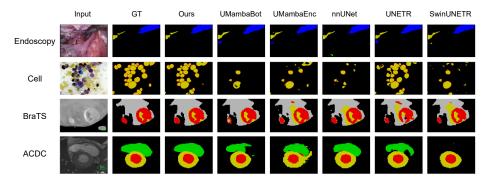
Fig. 4: Qualitative comparisons between our method and the baselines

with 236, 59, and 74 MR images, respectively. The common volume size was $240 \times 240 \times 155$. Following the literature, we focused on three objects of interest: enhancing tumor (ET), tumor core (TC), and whole tumor (WT).

ACDC was composed of a training and test split, whose size was 200 and 100 samples, respectively. The former was further divided into 160 training and 40 validation images. The segmentation targets were right ventricle (RV), left ventricle (LV), and myocardium (MYO).

4.3   Implementation Details

Model Training and Hyperparameters. Our models were trained using Nvidia V100 GPUs. We implemented our methods using Pytorch [43]. On each dataset, we followed a standard data preprocessing pipeline for all models. While training on 2D inputs, we extracted patches of a standard size, followed by augmentations by flipping, elastic deformation, color jittering, and noise addition. During the optimization on 3D data, the MR images were randomly cropped to a fixed volume size, and then augmented by flipping, intensity scaling, and intensity shifting. We applied a sliding window in the validation and test stages.

Our 2D and 3D segmentation models used α's of 2 and 1, respectively. We merely performed 1 horizontal scan ($M = 1$) for our methods. In our GTX blocks, except for the highest level that used 2 channels per token ($\gamma = 2$), other global tokens corresponded to 1 channel ($\gamma = 1$). Among different concatenation strategies, the "Interleaved" setting was empirically chosen. To train our proposed models, we used the Adam optimizer [28] with an initial learning rate of 10^{-4}. We utilized a threshold of 0.5 to binarize predictions. The more specific configurations are presented in Tab. 1. Our employed loss function was the sum of Dice and Cross-Entropy losses. All experiments were repeated using 5 different random seeds and folds, and we reported the means and standard errors across these 5 runs.

Metrics. We evaluated 2D models with Dice score and Intersection-over-Union (IoU). Regarding 3D models, we computed the Dice score for each class. In addition, we employed one surface-distance-based performance metric. We
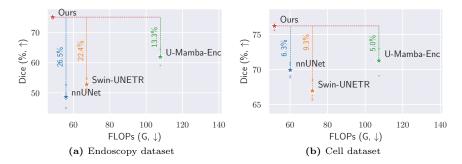
Fig. 5: Computational efficiency and performance comparisons on the 2D datasets. Stars indicate the means while blurry dots represent the individual results.
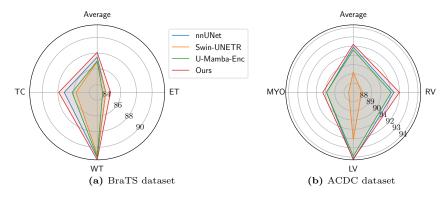

Fig. 6: Fine-grained performance comparisons on the 3D datasets (Dice, % ↑)

used Normalized Surface Dice (NSD) for 2D models following [34,38] and 95% Hausdorff distance (HD95) following [9,45,53].

4.4   Comparison with State-of-The-Art Methods

We compared our method to a diverse array of references on both 2D and 3D medical imaging datasets. These references included convolutional, Transformer-based, and Mamba-based models, whose representatives were nnUNet [25], Swin-UNETR [17], and U-Mamba-Enc [38], respectively. In general, our proposed method consistently acquired improvements over the baselines across 4 different datasets without compromising efficiency. The quantitative comparisons are presented in Figs. 5 and 6, Tabs. 2 to 5, and Tabs. S1 and S2. The qualitative results are demonstrated in Fig. 4.

Endoscopy. In Fig. 5a, we compared our method to three representative baselines: nnUNet (CNN-based) [25], Swin-UNETR (Transformer-based) [17], and U-Mamba-Enc (SSM-based) [38]. The results demonstrate that our method overcame the typical trade-off between computational cost and performance among the baselines. While achieving the lowest FLOPs, it significantly outper-

Table 2: Comparisons on the Endoscopy test set. The best results are highlighted in bold.

Method	Size	FLOPs	Dice (%) ↑	IoU (%) ↑	NSD (%) ↑
UNet [44]	32.5M	40.8G	$32.61_{\pm 2.67}$	$29.00_{\pm 2.67}$	$33.69_{\pm 2.69}$
nnUNet [25]	47.6M	55.9G	$48.64_{\pm 1.13}$	$45.21_{\pm 1.09}$	$49.88_{\pm 1.15}$
UNETR [18]	88.3M	111.7G	$41.87_{\pm 0.49}$	$38.45_{\pm 0.48}$	$43.27_{\pm 0.50}$
Swin-UNETR [17]	25.1M	67.3G	$52.78_{\pm 0.81}$	$49.55_{\pm 0.78}$	$54.31_{\pm 0.83}$
U-Mamba-Bot [38]	64.0M	99.3G	$62.20_{\pm 1.85}$	$58.11_{\pm 1.93}$	$63.78_{\pm 1.85}$
U-Mamba-Enc [38]	66.0M	107.8G	$61.91_{\pm 0.77}$	$57.85_{\pm 0.76}$	$63.43_{\pm 0.77}$
Swin-UMamba [34]	59.9M	163.7G	$65.15_{\pm 0.70}$	$61.65_{\pm 0.72}$	$66.64_{\pm 0.71}$
Swin-UMamba† [34]	27.5M	45.4G	$71.23_{\pm 1.00}$	$67.81_{\pm 0.99}$	$72.77_{\pm 1.02}$
Ours	30.3M	48.6G	$\mathbf{75.17}_{\pm 0.24}$	$\mathbf{71.68}_{\pm 0.23}$	$\mathbf{76.83}_{\pm 0.25}$

Table 3: Comparisons on the Cell test set. Metric[i] denotes an instance segmentation metric while Metric[s] denotes a semantic segmentation metric. The best results are highlighted in bold.

Method	Size	FLOPs	Dice[i] (%) ↑	IoU[i] (%) ↑	Dice[s] (%) ↑	NSD[s] (%) ↑
UNet [44]	32.5M	43.1G	$41.23_{\pm 1.35}$	$30.99_{\pm 1.11}$	$53.00_{\pm 1.50}$	$61.87_{\pm 1.70}$
nnUNet [25]	65.2M	60.1G	$53.44_{\pm 0.56}$	$41.41_{\pm 0.54}$	$69.92_{\pm 0.37}$	$79.03_{\pm 0.41}$
UNETR [18]	88.3M	120.3G	$44.42_{\pm 0.79}$	$32.52_{\pm 0.67}$	$71.28_{\pm 0.86}$	$80.03_{\pm 0.65}$
Swin-UNETR [17]	25.1M	71.9G	$42.90_{\pm 0.26}$	$31.91_{\pm 0.25}$	$66.95_{\pm 0.57}$	$76.48_{\pm 0.54}$
U-Mamba-Bot [38]	86.2M	101.8G	$57.86_{\pm 0.50}$	$45.16_{\pm 0.55}$	$70.86_{\pm 0.48}$	$80.52_{\pm 0.43}$
U-Mamba-Enc [38]	86.4M	107.3G	$59.50_{\pm 0.47}$	$46.99_{\pm 0.49}$	$71.25_{\pm 0.55}$	$80.54_{\pm 0.64}$
Swin-UMamba [34]	59.9M	174.5G	$52.92_{\pm 1.02}$	$40.02_{\pm 0.99}$	$72.52_{\pm 0.92}$	$82.80_{\pm 0.92}$
Swin-UMamba† [34]	27.5M	48.3G	$57.89_{\pm 1.17}$	$45.21_{\pm 1.11}$	$73.50_{\pm 0.86}$	$83.31_{\pm 0.66}$
Ours	30.3M	51.7G	$\mathbf{60.74}_{\pm 0.27}$	$\mathbf{47.88}_{\pm 0.32}$	$\mathbf{76.21}_{\pm 0.10}$	$\mathbf{86.44}_{\pm 0.09}$

formed the three baselines with differences of 26.5%, 22.4%, and 13.3% in Dice, respectively. Compared to the best-performing baseline, Swin-UMamba† [34], our method led to significant improvements of 3.94% in Dice, 3.87% in IoU, and 4.06% in NSD. More quantitative results are presented in Tab. 2.

Cell. For the Cell dataset, we evaluated metrics for both instance and semantic segmentation in Fig. 5b and Tab. 3. Each model predicted regions of cells, as well as their boundaries. These boundaries were used to separate the cell foreground into multiple cell instances. Compared to nnUNet [25], Swin-UNETR [17], and U-Mamba-Enc [38] on the semantic segmentation task, our method was the most computationally efficient while surpassing these baselines with differences of 6.3%, 9.3%, and 5.0% in Dice, respectively (see Fig. 5b). For both instance and semantic segmentation, our model consistently achieved the highest performances across all the metrics. Compared to Swin-UMamba† [34], on which our model was based, using LoG-VMamba resulted in substantial gains

Table 4: Comparisons on the BraTS test set. The best results are highlighted in bold.

Method	Size	FLOPs	Dice score (%) ↑	HD95 (mm) ↓
UNet3D [27]	5.7M	306.1G	$86.52_{\pm0.12}$	$5.42_{\pm0.37}$
nnUNet [25]	192.2M	727.5G	$87.54_{\pm0.26}$	$4.83_{\pm0.44}$
UNETR [18]	102.4M	184.9G	$84.78_{\pm0.24}$	$5.90_{\pm0.22}$
Swin-UNETR [17]	62.2M	777.3G	$86.90_{\pm0.20}$	$5.36_{\pm0.20}$
NestedFormer [53]	10.6M	207.3G	$86.71_{\pm0.05}$	$6.12_{\pm0.39}$
EoFormer [45]	6.4M	81.7G	$86.18_{\pm0.18}$	$5.34_{\pm0.24}$
U-Mamba-Bot [38]	30.1M	393.9G	$87.54_{\pm0.18}$	$4.30_{\pm0.16}$
U-Mamba-Enc [38]	32.0M	401.2G	$87.01_{\pm0.10}$	$4.38_{\pm0.09}$
SegMamba [52]	66.9M	1477.3G	$87.62_{\pm0.16}$	$4.73_{\pm0.22}$
Ours	31.5M	401.6G	$\mathbf{88.06}_{\pm0.08}$	$\mathbf{3.97}_{\pm0.04}$

Table 5: Comparisons on the ACDC test set. The best results are highlighted in bold.

Method	Size	FLOPs	Dice (%) ↑	HD95 (%) ↓
UNet3D [27]	5.7M	131.6G	$90.92_{\pm0.04}$	$1.16_{\pm0.01}$
nnUNet [25]	191.8M	479.7G	$91.88_{\pm0.04}$	$1.17_{\pm0.07}$
UNETR [18]	101.8M	240.5G	$86.54_{\pm0.09}$	$2.49_{\pm0.09}$
Swin-UNETR [17]	62.2M	769.4G	$89.19_{\pm0.19}$	$2.17_{\pm0.17}$
NestedFormer [53]	5.4M	201.7G	$90.15_{\pm0.11}$	$1.90_{\pm0.33}$
EoFormer [45]	6.4M	35.5G	$91.12_{\pm0.07}$	$1.19_{\pm0.02}$
U-Mamba-Bot [38]	57.3M	431.8G	$91.94_{\pm0.05}$	$1.26_{\pm0.15}$
U-Mamba-Enc [38]	59.9M	466.4G	$91.65_{\pm0.31}$	$1.13_{\pm0.02}$
SegMamba [52]	66.9M	637.2G	$90.74_{\pm0.09}$	$1.19_{\pm0.02}$
Ours	59.7M	467.9G	$\mathbf{92.18}_{\pm0.13}$	$\mathbf{1.10}_{\pm0.00}$

of 2.71% in Dice and 3.13% in NSD for semantic segmentation, as well as 2.85% in Dice and 2.67% in IoU for instance segmentation.

BraTS 2020. Apart from the gain in performances on 2D datasets, our method also showed competitive results on 3D datasets such as BraTS. In Fig. 6a, our method substantially performed better than the three baselines – nnUNet [25], Swin-UNETR [17], and U-Mamba-Enc [38] – on all individual classes. When we directly compared our method to its base model, U-Mamba-Enc, we observed substantial improvements of 1.05% in Dice and 0.41mm in HD95. We present more quantitative results in Tabs. S1 and 4. As we used an expansion factor $\alpha = 1$ in every Mamba block, we spent 0.5M fewer parameters than U-Mamba-Enc while using a similar number of FLOPs. Ultimately, compared to SegMamba [52], the most competitive baseline, our method obtained better results with differences of 0.44% in Dice and 0.76mm in HD95.

Table 6: Ablation studies on the Endoscopy and BraTS test sets. The best results are highlighted in bold, and the second-best ones are underlined.

Method	Endoscopy (2D)			BraTS (3D)	
	Dice (%) ↑	IoU (%) ↑	NSD (%) ↑	Dice (%) ↑	HD95 (mm) ↓
VSS	$71.23_{\pm1.00}$	$67.81_{\pm0.99}$	$72.77_{\pm1.02}$	$87.01_{\pm0.10}$	$4.38_{\pm0.09}$
G-VMamba	$72.64_{\pm0.21}$	$69.08_{\pm0.22}$	$74.24_{\pm0.22}$	$87.99_{\pm0.09}$	$\underline{4.05_{\pm0.07}}$
L-VMamba	$\underline{74.15_{\pm0.12}}$	$\underline{70.56_{\pm0.13}}$	$\underline{75.81_{\pm0.13}}$	$87.71_{\pm0.09}$	$4.12_{\pm0.03}$
LoG-VMamba	$\mathbf{75.17_{\pm0.24}}$	$\mathbf{71.68_{\pm0.23}}$	$\mathbf{76.83_{\pm0.25}}$	$\mathbf{88.06_{\pm0.08}}$	$\mathbf{3.97_{\pm0.04}}$

ACDC. In Tabs. S2 and 5, we reported the experimental results on the ACDC dataset. Our network achieved a Dice score of 92.18% and HD95 of 1.10mm, which were 0.53% higher in Dice and 0.03mm lower in HD95 than the original U-Mamba-Enc. In addition, Fig. 6b demonstrates that our method consistently performed better than nnUNet, Swin-UNETER, and U-Mamba-Enc across all the fine-grained classes. Compared to U-Mamba-Bot [38], which reached the highest Dice score among the baselines, our method also performed better with improvements of 0.24% in Dice and 0.16mm in HD95.

4.5   Ablation Studies

We conducted ablation studies on one 2D dataset, Endoscopy, and one 3D dataset, BraTS. Firstly, we investigated the impact of LTX and GTX. Furthermore, we examined the effects of distinct approaches to concatenate global to local tokens, as well as employing multiple scanning directions.

Impact of Each Component. In Tab. 6, instead of using the LoG-VMamba block, we experimented with using either L-VMamba or G-VMamba block in our 2D and 3D models. L-VMamba was more effective than the G-VMamba block on the 2D Endoscopy dataset. While G-VMamba led to an increase of 1.41% in Dice and 1.47% in NSD, L-VMamba resulted in improvements of 2.92% in Dice and 3.04% in NSD compared to the vanilla VSS. When compared to L-VMamba, the combined block LoG-VMamba outperformed with differences of 1.02% in both Dice and NSD. On the other hand, G-VMamba was better than L-VMamba on the 3D BraTS dataset, outperforming by 0.28% in Dice and 0.07mm in HD95. The combination of both LTX and GTX resulted in further improvements of 0.07% in Dice and 0.08mm in HD95.

Concatenation of Local and Global Tokens. We used only G-VMamba in this experiment, and present the detailed results in Tab. 7. The "Interleaved" strategy achieved the best performance on the Endoscopy dataset. As such, it helped G-VMamba to improve 0.28% in Dice and 0.32% in IoU in comparison with the second-best strategy "Center". The best-performing strategy on BraTS was not as obvious. While the "Interleaved" strategy reached the highest Dice at 87.99%, the lowest HD95 of 4.00mm belonged to the "Split" strategy.

Scanning Strategies. We defined M as the number of scanning directions and evaluated 3 scanning strategies: ($M = 1$) only horizontal scan was used;

Table 7: Effects of using different concatenation strategies on the Endoscopy and BraTS test sets

Strategy	Endoscopy (2D)			BraTS (3D)	
	Dice (%) ↑	IoU (%) ↑	NSD (%) ↑	Dice (%) ↑	HD95 (mm) ↓
Head	$71.20_{\pm0.16}$	$67.63_{\pm0.16}$	$72.78_{\pm0.17}$	$87.71_{\pm0.12}$	$4.01_{\pm0.02}$
Split	$72.01_{\pm0.21}$	$68.42_{\pm0.21}$	$73.60_{\pm0.21}$	$87.72_{\pm0.11}$	$4.00_{\pm0.04}$
Center	$72.36_{\pm0.30}$	$68.76_{\pm0.31}$	$73.96_{\pm0.31}$	$87.51_{\pm0.18}$	$4.05_{\pm0.05}$
Interleaved	$72.64_{\pm0.21}$	$69.08_{\pm0.22}$	$74.24_{\pm0.22}$	$87.99_{\pm0.09}$	$4.05_{\pm0.07}$

Table 8: Effects of using multiple scanning directions on performances on the Endoscopy and BraTS test sets. The substantially best results are highlighted in bold.

M	Endoscopy (2D)			BraTS (3D)	
	Dice (%) ↑	IoU (%) ↑	NSD (%) ↑	Dice (%) ↑	HD95 (mm) ↓
1	$\mathbf{75.17}_{\pm0.24}$	$\mathbf{71.68}_{\pm0.23}$	$\mathbf{76.83}_{\pm0.25}$	$\mathbf{88.06}_{\pm0.08}$	$\mathbf{3.97}_{\pm0.04}$
2	$74.03_{\pm0.31}$	$70.50_{\pm0.31}$	$75.67_{\pm0.31}$	$87.88_{\pm0.08}$	$4.20_{\pm0.07}$
4	$75.08_{\pm0.13}$	$71.59_{\pm0.13}$	$76.75_{\pm0.13}$	$87.84_{\pm0.13}$	$4.13_{\pm0.13}$

($M = 2$) both horizontal and vertical scan were used; ($M = 4$) Mamba scanned horizontally and vertically in both forward and backward directions. As shown in Tab. 8, we did not observe the benefits of employing multiple scanning directions in the SSM module for our models. On both datasets, using the most computationally efficient approach, $M = 1$, obtained the best performance. Utilizing more than one scanning direction negatively impacted the performance.

5   Conclusion

In this study, we have introduced a straightforward, yet effective and efficient approach to advance the SSMs for MIS. Our proposed framework addresses the fundamental sequential limitations of SSM-based methods in handling high-dimensional data such as 2D and 3D medical images. As such, we propose the LTX and GTX modules to enhance tokens with both local and global receptive fields, which are inspired by the strength of CNN and ViT respectively. We then leverage these two components to form the LoG-VMamba block.

As our framework is not specifically designed for MIS, it may be applicable to other problems such as classification and detection, or even multimodal applications. It is, however, outside the scope of the present work, the focus of which is improving MIS. Our experiments show that LoG-VMamba can be well integrated into advanced segmentation models such as Swin-UMamba and U-Mamba-Enc, leading to consistent improvements across distinct 2D and 3D medical imaging datasets. Furthermore, our method's enriched tokens eliminate the need for a complex scanning strategy, thereby enhancing computational efficiency.

References

1. Allan, M., Shvets, A., Kurmann, T., Zhang, Z., Duggal, R., Su, Y.H., Rieke, N., Laina, I., Kalavakonda, N., Bodenstedt, S., et al.: 2017 robotic instrument segmentation challenge. arXiv preprint arXiv:1902.06426 (2019) 8

2. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016) 6

3. Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J.S., Freymann, J.B., Farahani, K., Davatzikos, C.: Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. Scientific data **4**(1), 1–13 (2017) 8

4. Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., Shinohara, R.T., Berger, C., Ha, S.M., Rozycki, M., et al.: Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. arXiv preprint arXiv:1811.02629 (2018) 8

5. Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.A., Cetin, I., Lekadir, K., Camara, O., Ballester, M.A.G., et al.: Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? IEEE transactions on medical imaging **37**(11), 2514–2525 (2018) 8

6. 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. Springer (2022) 1, 4

7. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1251–1258 (2017) 3

8. Dai, Z., Liu, H., Le, Q.V., Tan, M.: Coatnet: Marrying convolution and attention for all data sizes. Advances in neural information processing systems **34**, 3965–3977 (2021) 3

9. Dang, T., Nguyen, H.H., Tiulpin, A.: Singr: Brain tumor segmentation via signed normalized geodesic transform regression. arXiv preprint arXiv:2405.16813 (2024) 10

10. Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., Guo, B.: Cswin transformer: A general vision transformer backbone with cross-shaped windows. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12124–12134 (2022) 3

11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 2, 3

12. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023) 2, 3, 4, 5, 6

13. Gu, A., Dao, T., Ermon, S., Rudra, A., Ré, C.: Hippo: Recurrent memory with optimal polynomial projections. Advances in neural information processing systems **33**, 1474–1487 (2020) 3

14. Gu, A., Goel, K., Ré, C.: Efficiently modeling long sequences with structured state spaces. arXiv preprint arXiv:2111.00396 (2021) 3

15. Guo, J., Han, K., Wu, H., Xu, C., Tang, Y., Xu, C., Wang, Y.: Cmt: Convolutional neural networks meet vision transformers. 2022 ieee. In: CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12165–12175 (2022) 3

16. Hassani, A., Walton, S., Li, J., Li, S., Shi, H.: Neighborhood attention transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6185–6194 (2023) 3

17. Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H.R., Xu, D.: Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In: International MICCAI Brainlesion Workshop. pp. 272–284. Springer (2021) 1, 4, 10, 11, 12, S2

18. Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H.R., Xu, D.: Unetr: Transformers for 3d medical image segmentation. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 574–584 (2022) 1, 4, 11, 12, S2

19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 2, 3

20. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017) 3

21. 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) 3

22. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017) 2

23. Huang, H., Zhou, X., Cao, J., He, R., Tan, T.: Vision transformer with super token sampling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22690–22699 (2023) 6

24. Huang, T., Pei, X., You, S., Wang, F., Qian, C., Xu, C.: Localmamba: Visual state space model with windowed selective scan. arXiv preprint arXiv:2403.09338 (2024) 4, 5, 6

25. Isensee, F., Wald, T., Ulrich, C., Baumgartner, M., Roy, S., Maier-Hein, K., Jaeger, P.F.: nnu-net revisited: A call for rigorous validation in 3d medical image segmentation. arXiv preprint arXiv:2404.09556 (2024) 10, 11, 12, S2

26. Kalman, R.E.: A new approach to linear filtering and prediction problems (1960) 4

27. Kerfoot, E., Clough, J., Oksuz, I., Lee, J., King, A.P., Schnabel, J.A.: Left-ventricle quantification using residual u-net. In: Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges: 9th International Workshop, STACOM 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers 9. pp. 371–380. Springer (2019) 12, S2

28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 9

29. Knigge, D.M., Romero, D.W., Gu, A., Gavves, E., Bekkers, E.J., Tomczak, J.M., Hoogendoorn, M., jakob Sonke, J.: Modelling long range dependencies in nd: From task-specific to a general purpose CNN. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=ZW5aK4yCRqU 2

30. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM 60(6), 84–90 (2017) 3

31. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521(7553), 436–444 (2015) 2

32. Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.W., Heng, P.A.: H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. IEEE transactions on medical imaging **37**(12), 2663–2674 (2018) 4

33. Lin, W., Wu, Z., Chen, J., Huang, J., Jin, L.: Scale-aware modulation meet transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6015–6026 (2023) 3

34. Liu, J., Yang, H., Zhou, H.Y., Xi, Y., Yu, L., Yu, Y., Liang, Y., Shi, G., Zhang, S., Zheng, H., et al.: Swin-umamba: Mamba-based unet with imagenet-based pretraining. arXiv preprint arXiv:2402.03302 (2024) 2, 4, 8, 10, 11, S1

35. Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Liu, Y.: Vmamba: Visual state space model. arXiv preprint arXiv:2401.10166 (2024) 2, 4, 5, 6

36. 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) 3, 5

37. 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) 2

38. Ma, J., Li, F., Wang, B.: U-mamba: Enhancing long-range dependency for biomedical image segmentation. arXiv preprint arXiv:2401.04722 (2024) 2, 4, 8, 10, 11, 12, 13, S1, S2

39. Ma, J., Xie, R., Ayyadhury, S., Ge, C., Gupta, A., Gupta, R., Gu, S., Zhang, Y., Lee, G., Kim, J., et al.: The multimodality cell segmentation challenge: toward universal solutions. Nature methods pp. 1–11 (2024) 8

40. Menze, B.H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al.: The multimodal brain tumor image segmentation benchmark (brats). IEEE transactions on medical imaging **34**(10), 1993–2024 (2014) 8

41. Orvieto, A., Smith, S.L., Gu, A., Fernando, A., Gulcehre, C., Pascanu, R., De, S.: Resurrecting recurrent neural networks for long sequences. In: International Conference on Machine Learning. pp. 26670–26698. PMLR (2023) 3

42. Pan, X., Ge, C., Lu, R., Song, S., Chen, G., Huang, Z., Huang, G.: On the integration of self-attention and convolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 815–825 (2022) 3

43. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019) 9

44. 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. Springer (2015) 1, 4, 11

45. She, D., Zhang, Y., Zhang, Z., Li, H., Yan, Z., Sun, X.: Eoformer: Edge-oriented transformer for brain tumor segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 333–343. Springer (2023) 10, 12, S2

46. Shi, Y., Dong, M., Xu, C.: Multi-scale vmamba: Hierarchy in hierarchy visual state space model. arXiv preprint arXiv:2405.14174 (2024) 2, 4

47. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 3

48. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021) 3

49. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 3

50. Wang, F., Wang, J., Ren, S., Wei, G., Mei, J., Shao, W., Zhou, Y., Yuille, A., Xie, C.: Mamba-r: Vision mamba also needs registers. arXiv preprint arXiv:2405.14858 (2024) 4

51. Xiao, X., Lian, S., Luo, Z., Li, S.: Weighted res-unet for high-quality retina vessel segmentation. In: 2018 9th international conference on information technology in medicine and education (ITME). pp. 327–331. IEEE (2018) 1, 4

52. Xing, Z., Ye, T., Yang, Y., Liu, G., Zhu, L.: Segmamba: Long-range sequential modeling mamba for 3d medical image segmentation. arXiv preprint arXiv:2401.13560 (2024) 2, 12, S2

53. Xing, Z., Yu, L., Wan, L., Han, T., Zhu, L.: Nestedformer: Nested modality-aware transformer for brain tumor segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 140–150. Springer (2022) 10, 12, S2

54. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015) 2, 3

55. Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S.: Metaformer is actually what you need for vision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10819–10829 (2022) 3

56. Zhao, S., Chen, H., Zhang, X., Xiao, P., Bai, L., Ouyang, W.: Rs-mamba for large remote sensing image dense prediction. arXiv preprint arXiv:2404.02668 (2024) 2, 4, 5, 6

57. Zhou, D., Kang, B., Jin, X., Yang, L., Lian, X., Jiang, Z., Hou, Q., Feng, J.: Deepvit: Towards deeper vision transformer. arXiv preprint arXiv:2103.11886 (2021) 2

58. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE transactions on medical imaging **39**(6), 1856–1867 (2019) 1

59. Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: Efficient visual representation learning with bidirectional state space model. arXiv preprint arXiv:2401.09417 (2024) 2, 4, 5, 6

Supplementary Material

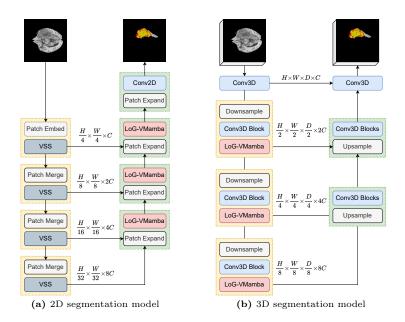**(a)** 2D segmentation model   **(b)** 3D segmentation model

Fig. S1: The overview of our 2D and 3D segmentation models. These illustrations serve as a conceptual representation. Following [34, 38], the number of blocks differs across datasets. The detailed configurations are shown in Tab. 1.

Table S1: Performance comparisons on the BraTS test set for each class. The best results are highlighted in bold while the second-best ones are underlined.

Method	Dice score (%) ↑				HD95 (mm) ↓			
	ET	TC	WT	Avg	ET	TC	WT	Avg
UNet3D [27]	$83.1_{\pm0.2}$	$86.1_{\pm0.3}$	$90.4_{\pm0.2}$	$86.5_{\pm0.1}$	$3.8_{\pm0.4}$	$5.9_{\pm0.3}$	$6.5_{\pm0.6}$	$5.4_{\pm0.4}$
nnUNet [25]	$84.1_{\pm0.3}$	$\underline{87.2}_{\pm0.5}$	$91.3_{\pm0.1}$	$87.5_{\pm0.3}$	$3.3_{\pm0.5}$	$5.7_{\pm0.6}$	$5.5_{\pm0.3}$	$4.8_{\pm0.4}$
UNETR [18]	$82.3_{\pm0.2}$	$82.0_{\pm0.4}$	$90.0_{\pm0.1}$	$84.8_{\pm0.2}$	$4.0_{\pm0.3}$	$7.4_{\pm0.2}$	$6.2_{\pm0.4}$	$5.9_{\pm0.2}$
Swin-UNETR [17]	$84.1_{\pm0.2}$	$85.7_{\pm0.4}$	$90.9_{\pm0.1}$	$86.9_{\pm0.2}$	$3.7_{\pm0.3}$	$6.4_{\pm0.2}$	$6.0_{\pm0.2}$	$5.4_{\pm0.2}$
NestedFormer [53]	$83.5_{\pm0.1}$	$85.4_{\pm0.1}$	$91.2_{\pm0.1}$	$86.7_{\pm0.1}$	$4.4_{\pm0.4}$	$7.4_{\pm0.4}$	$6.6_{\pm0.4}$	$6.1_{\pm0.4}$
EoFormer [45]	$82.5_{\pm0.2}$	$84.8_{\pm0.4}$	$91.3_{\pm0.0}$	$86.2_{\pm0.2}$	$3.7_{\pm0.2}$	$6.4_{\pm0.4}$	$5.9_{\pm0.3}$	$5.3_{\pm0.2}$
U-Mamba-Bot [38]	$84.1_{\pm0.2}$	$87.0_{\pm0.3}$	$\underline{91.5}_{\pm0.1}$	$87.5_{\pm0.2}$	$\underline{2.9}_{\pm0.1}$	$5.0_{\pm0.3}$	$\mathbf{5.0}_{\pm0.2}$	$4.3_{\pm0.2}$
U-Mamba-Enc [38]	$83.7_{\pm0.1}$	$86.2_{\pm0.2}$	$91.2_{\pm0.1}$	$87.0_{\pm0.1}$	$3.1_{\pm0.2}$	$\underline{5.0}_{\pm0.2}$	$\underline{5.1}_{\pm0.1}$	$4.4_{\pm0.1}$
SegMamba [52]	$\mathbf{85.0}_{\pm0.2}$	$86.7_{\pm0.3}$	$91.2_{\pm0.1}$	$\underline{87.6}_{\pm0.2}$	$3.2_{\pm0.3}$	$5.8_{\pm0.4}$	$5.2_{\pm0.1}$	$4.7_{\pm0.2}$
Ours	$\underline{84.7}_{\pm0.2}$	$\mathbf{87.9}_{\pm0.3}$	$\mathbf{91.6}_{\pm0.1}$	$\mathbf{88.1}_{\pm0.1}$	$\mathbf{2.4}_{\pm0.1}$	$\mathbf{4.5}_{\pm0.1}$	$\mathbf{5.0}_{\pm0.2}$	$\mathbf{4.0}_{\pm0.0}$

Table S2: Performance comparisons on the ACDC test set for each class. The best results are highlighted in bold while the second-best ones are underlined.

Method	Dice score (%) ↑				HD95 (mm) ↓			
	RV	MYO	LV	Avg	RV	Myo	LV	Avg
UNet3D [27]	$90.2_{\pm0.1}$	$89.3_{\pm0.1}$	$93.3_{\pm0.1}$	$90.9_{\pm0.0}$	$1.3_{\pm0.0}$	$1.1_{\pm0.0}$	$1.2_{\pm0.0}$	$1.2_{\pm0.0}$
nnUNet [25]	$91.4_{\pm0.0}$	$89.9_{\pm0.0}$	$\mathbf{94.3}_{\pm0.1}$	$\underline{91.9}_{\pm0.0}$	$\mathbf{1.2}_{\pm0.0}$	$\mathbf{1.0}_{\pm0.0}$	$1.3_{\pm0.2}$	$\underline{1.2}_{\pm0.1}$
UNETR [18]	$85.0_{\pm0.2}$	$84.7_{\pm0.2}$	$89.9_{\pm0.2}$	$86.5_{\pm0.1}$	$2.8_{\pm0.1}$	$2.1_{\pm0.1}$	$2.6_{\pm0.1}$	$2.5_{\pm0.1}$
Swin-UNETR [17]	$87.9_{\pm0.2}$	$87.5_{\pm0.2}$	$92.1_{\pm0.3}$	$89.2_{\pm0.2}$	$2.8_{\pm0.3}$	$1.4_{\pm0.1}$	$2.3_{\pm0.2}$	$2.2_{\pm0.2}$
NestedFormer [53]	$89.2_{\pm0.1}$	$88.3_{\pm0.1}$	$92.9_{\pm0.1}$	$90.1_{\pm0.1}$	$1.6_{\pm0.2}$	$1.6_{\pm0.4}$	$2.5_{\pm0.6}$	$1.9_{\pm0.3}$
EoFormer [45]	$89.9_{\pm0.0}$	$89.8_{\pm0.0}$	$93.7_{\pm0.2}$	$91.1_{\pm0.1}$	$\underline{1.3}_{\pm0.0}$	$\mathbf{1.0}_{\pm0.0}$	$\underline{1.2}_{\pm0.1}$	$1.2_{\pm0.0}$
U-Mamba-Bot [38]	$\underline{91.6}_{\pm0.1}$	$90.2_{\pm0.0}$	$94.0_{\pm0.1}$	$\underline{91.9}_{\pm0.1}$	$\mathbf{1.2}_{\pm0.0}$	$1.3_{\pm0.2}$	$1.4_{\pm0.2}$	$1.3_{\pm0.1}$
U-Mamba-Enc [38]	$91.1_{\pm0.4}$	$89.9_{\pm0.3}$	$93.9_{\pm0.2}$	$91.6_{\pm0.3}$	$\mathbf{1.2}_{\pm0.0}$	$\mathbf{1.0}_{\pm0.0}$	$\mathbf{1.1}_{\pm0.0}$	$\mathbf{1.1}_{\pm0.0}$
SegMamba [52]	$89.6_{\pm0.2}$	$89.0_{\pm0.1}$	$93.6_{\pm0.0}$	$90.7_{\pm0.1}$	$1.3_{\pm0.0}$	$1.1_{\pm0.0}$	$1.2_{\pm0.1}$	$1.2_{\pm0.0}$
Ours	$\mathbf{92.0}_{\pm0.1}$	$\mathbf{90.3}_{\pm0.0}$	$\underline{94.2}_{\pm0.1}$	$\mathbf{92.2}_{\pm0.0}$	$\mathbf{1.2}_{\pm0.0}$	$\mathbf{1.0}_{\pm0.0}$	$\mathbf{1.1}_{\pm0.0}$	$\mathbf{1.1}_{\pm0.0}$