SwinUNETR-V2: Stronger Swin Transformers with Stagewise Convolutions for 3D Medical Image Segmentation

Yufan He(✉), Vishwesh Nath, Dong Yang, Yucheng Tang, Andriy Myronenko, and Daguang Xu

NVidia, Santa Clara, US
yufanh@nvidia.com

Abstract. Transformers for medical image segmentation have attracted broad interest. Unlike convolutional networks (CNNs), transformers use self-attentions that do not have a strong inductive bias. This gives transformers the ability to learn long-range dependencies and stronger modeling capacities. Although they, e.g. SwinUNETR, achieve state-of-the-art (SOTA) results on some benchmarks, the lack of inductive bias makes transformers harder to train, requires much more training data, and are sensitive to training recipes. In many clinical scenarios and challenges, transformers can still have inferior performances than SOTA CNNs like nnUNet. A transformer backbone and corresponding training recipe, which can achieve top performances under different medical image segmentation scenarios, still needs to be developed. In this paper, we enhance the SwinUNETR with convolutions, which results in a surprisingly stronger backbone, the SwinUNETR-V2, for 3D medical image segmentation. It achieves top performance on a variety of benchmarks of different sizes and modalities, including the Whole abdominal ORgan Dataset (WORD), MICCAI FLARE2021 dataset, MSD pancreas dataset, MSD prostate dataset, and MSD lung cancer dataset, all using the same training recipe (https://github.com/Project-MONAI/research-contributions/tree/main/SwinUNETR/BTCV, our training recipe is the same as that by SwinUNETR) with minimum changes across tasks.

Keywords: Swin transformer · Convolution · Hybrid model · Medical image segmentation

1 Introduction

Medical image segmentation is a core step for quantitative and precision medicine. In the past decade, Convolutional Neural Networks (CNNs) became the SOTA method to achieve accurate and fast medical image segmentation [10,12,21]. nn-UNet [12], which is based on UNet [21], has achieved top performances on over 20 medical segmentation challenges. Parallel to manually created networks such as nn-UNet, DiNTS [10], a CNN designed by automated neural network search, also achieved top performances in medical segmentation decathlon (MSD) [1] challenges. The convolution operation in CNN provides a strong inductive bias which is translational equivalent and efficient in capturing local features like boundary and texture. However, this inductive bias limits the representation power of CNN models which means a potentially lower

H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14223, pp. 416–426, 2023.
https://doi.org/10.1007/978-3-031-43901-8_40

performance ceiling on more challenging tasks [7]. Additionally, CNN has a local receptive field and are not able to capture long-range dependencies unlike transformers. Recently, vision transformers have been proposed, which adopt the transformers in natural language processing by splitting images into patches (tokens) [6], and use self-attention to learn features. The self-attention mechanism enables learning long-range dependencies between far-away tokens. This is intriguing and numerous works have been proposed to incorporate transformer attentions into medical image segmentation [2,3,9,23,24,30,32,35]. Among them, SwinUNETR [23] has achieved the new top performance in the MSD challenge and Beyond the Cranial Vault (BTCV) Segmentation Challenge by pretraining on large datasets. It has a U-shaped structure where the encoder is a Swin-Transformer [16].

Although transformers have achieved certain success in medical imaging, the lack of inductive bias makes them harder to be trained and requires much more training data to avoid overfitting. The self-attentions are good at learning complicated relational interactions for high-level concepts [5] but are also observed to be ignoring local feature details [5]. Unlike natural image segmentation benchmarks, e.g. ADE20k [34], where the challenge is in learning complex relationships and scene understanding from a large amount of labeled training images, many medical image segmentation networks need to be extremely focused on local boundary details while less in need of high-level relationships. Moreover, the number of training data is also limited. Hence in real clinical studies and challenges, CNNs can still achieve better results than transformers. For example, the top solutions in the last year MICCAI challenges HECTOR [19], FLARE [11], INSTANCE [15,22] and AMOS [13] are all CNN based. Besides lacking inductive bias and enough training data, one extra reason could be that transformers are computationally much expensive and harder to tune. More improvements and empirical evidence are needed before we say transformers are ready to replace CNNs for medical image segmentation.

In this paper, we try to develop a new "to-go" transformer for 3D medical image segmentation, which is expected to exhibit strong performance under different data situations and does not require extensive hyperparameter tuning. SwinUNETR reaches top performances on several large benchmarks, making itself the current SOTA, but without effective pretraining and excessive tuning, its performance on new datasets and challenges is not as high-performing as expected.

A straightforward direction to improve transformers is to combine the merits of both convolutions and self-attentions. Many methods have been proposed and most of them fall into two directions: 1) a new self-attention scheme to have convolution-like properties [5,7,16,25,26,29]. Swin-transformer [16] is a typical work in the first direction. It uses a local window instead of the whole image to perform self-attention. Although the basic operation is still self-attention, the local window and relative position embedding give self-attention a conv-like local receptive field and less computation cost. Another line in 1) is changing the self-attention operation directly. CoAtNet [5] unifies convolution and self-attention with relative attention, while ConViT [7] uses gated positional self-attention which is equipped with a soft convolutional inductive bias. Works in the second direction 2) employs both convolution and self-attention in the network [3,4,8,20,27,28,30,31,33,35]. For the works in this direction, we sum-

marize them into three major categories as shown in Fig. 1: 2.a) dual branch feature fusion. MobileFormer [4], Conformer [20], and TransFuse [33] use a CNN branch and a transformer branch in parallel to fuse the features, thus the local details and global features are learned separately and fused altogether. However, this doubles the computation cost. Another line of works 2.b) focuses on the bottleneck design. The low-level features are extracted by convolution blocks and the bottleneck is the transformer, like the TransUNet [3], Cotr [30] and TransBTS [27]. The third direction 2.c) is a new block containing both convolution and self-attention. MOAT [31] removes the MLP in self-attention and uses a mobile convolution block at the front. The MOAT block is then used as the basic block in building the network. CvT [28] uses convolution as the embedding layer for key, value, and query. nnFormer [35] replaces the patch merging with convolution with stride.

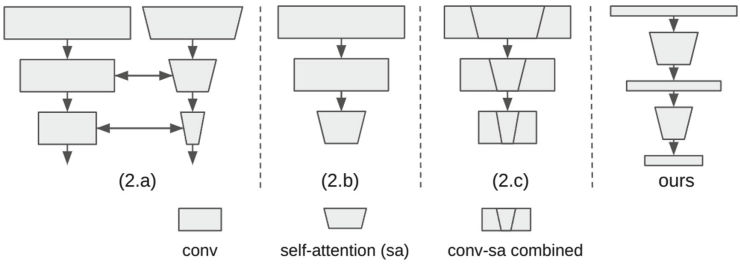

Fig. 1. Three major categories of methods combining convolution with transformers. (2.a): parallel branches with a CNN branch and a transformer branch [4,20,33]. (2.b): Using CNNs to extract local features in the lower level and use transformers in the bottleneck [3,27,30]. (2.c): New transformer blocks with convolution added [28,31]. Our SwinUNETR-V2 adds a convolution block at the beginning of each resolution stage.

Although those works showed strong performances, which works best and can be the "to go" transformer for 3D medical image segmentation is still unknown. For this purpose, we design the SwinUNETR-V2, which improves the current SOTA SwinUNETR by introducing stage-wise convolutions into the backbone. Our network belongs to the second category, which employs convolution and self-attention directly. At each resolution level, we add a residual convolution (ResConv) block at the beginning, and the output is then used as input to the swin transformer blocks (contains a swin block and a shifted window swin block). MOAT [31] and CvT [28] add convolution before self-attention as a micro-level building block, and nnFormer has a similar design that uses convolution with stride to replace the patch merging layer for downsampling. Differently, our work only adds a ResConv block at the beginning of each stage, which is a macro-network level design. It is used to regularize the features for the following transformers. Although simple, we found it surprisingly effective for 3D medical image segmentation. The network is evaluated extensively on a variety of benchmarks and achieved top performances on the WORD [17], FLARE2021 [18], MSD prostate, MSD lung cancer, and MSD pancreas cancer datasets [1]. Compared to the original SwinUNETR which needs extensive recipe tuning on a new dataset, we utilized the same

training recipe with minimum changes across all benchmarks, showcasing the straightforward applicability of SwinUNETR-V2 to reach state-of-the-art without extensive hyperparameter tuning or pretraining. We also experimented with four design variations inspired by existing works to justify the SwinUNETR-V2 design.

2 Method

Our SwinUNETR-V2 is based on the original SwinUNETR, and we focus on the transformer encoder. The overall framework is shown in Fig. 2.

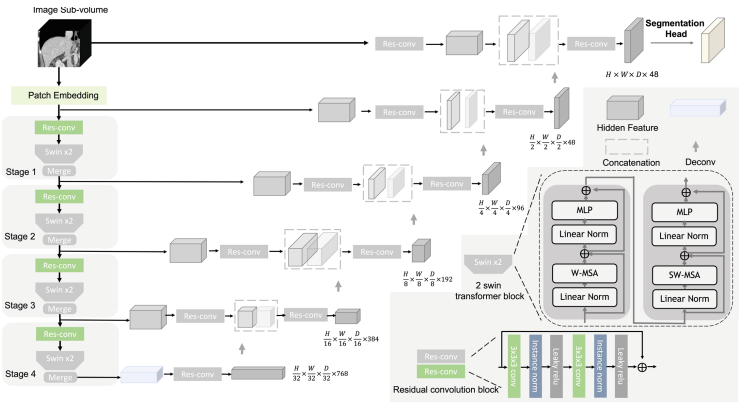

Fig. 2. The SwinUNETR-V2 architecture

Swin-Transformer. We briefly introduce the 3D swin-transformer as used in SwinUNETR [23]. A patch embedding layer of 3D convolution (stride = 2,2,2, kernel size = 2,2,2) is used to embed the patch into tokens. Four stages of swin transformer block followed by patch merging are used to encode the input patches. Given an input tensor z^i of size (B, C, H, W, D) at swin block i, the swin transformer block splits the tensor into $(\lceil H/M \rceil, \lceil W/M \rceil, \lceil D/M \rceil$ windows. It performs four operations

$$z^i = \text{W-MSA}(\text{LN}(z^{i-1})) + z^{i-1}; \quad z^i = \text{MLP}(\text{LN}(z^i)) + z^i$$
$$z^{i+1} = \text{SW-MSA}(\text{LN}(z^i)) + z^i; \quad z^{i+1} = \text{MLP}(\text{LN}(z^{i+1})) + z^{i+1}$$

W-MSA and SW-MSA represent regular window and shifted window multi-head self-attention, respectively. MLP and LN represent multilayer perceptron and layernorm, respectively. A patch merging layer is applied after every swin transformer block to reduce each spatial dimension by half.

Stage-Wise Convolution. Although Swin-transformer uses local window attention to introduce inductive bias like convolutions, self-attentions can still mess up with the local details. We experimented with multiple designs as in Fig. 3 and found that interleaved stage-wise convolution is the most effective for swin: convolution followed by swin blocks, then convolution goes on. At the beginning of each resolution level (stage), the input tokens are reshaped back to the original 3D volumes. A residual convolution (ResConv) block with two sets of 3×3x3 convolution, instance normalization, and leaky relu are used. The output then goes to a set of following swin transformer blocks (we use 2 in the paper). There are in total 4 ResConv blocks at 4 stages. We also tried inverted convolution blocks with depth-wise convolution like MOAT [31] or with original 3D convolution, they improve the performance but are worse than the ResConv block.

Decoder. The decoder is the same as SwinUNETR [23], where convolution blocks are used to extract outputs from those four swin blocks and the bottleneck. The extracted features are upsampled by deconvolutional layers and concatenated with features from a higher-resolution level(long-skip connection). A final convolution with $1 \times 1 \times 1$ kernel is used to map features to segmentation maps.

3 Experiments

We use extensive experiments to show its effectiveness and justify its design for 3D medical image segmentation. To make fair comparisons with baselines, we did not use any pre-trained weights.

Datasets. The network is validated on five datasets of different sizes, targets and modalities:

1) **The WORD dataset** [17] (large-scale Whole abdominal ORgan Dataset) contains 150 high-resolution abdominal CT volumes, each with 16 pixel-level organ annotations. A predefined data split of 100 training, 30 validation, and 20 test are provided. We use this split for our experiments.
2) **The MICCAI FLARE 2021 dataset** [18]. It provides 361 training scans with manual labels from 11 medical centers. Each scan is an abdominal 3D CT image with 4 organ annotations. We follow the test split in the 3D-UXNET[1] [14]: 20 hold-out test scans, and perform 5-fold 80%/20% train validation split on the rest 341 scans.
3) **MSD Task05 prostate, Task06 lung tumour and Task07 pancreas**. The Medical segmentation decathlon (MSD) [1] prostate dataset contains 32 labeled prostate MRI with two modalities for the prostate peripheral zone (PZ) and the transition zone (TZ). The challenges are the large inter-subject variability and limited training data. The lung tumor dataset contains 63 lung CT images with tumor annotations. The challenge comes from segmenting small tumors from large full 3D CT images. The pancreas dataset contains 281 3D CT scans with annotated pancreas and tumors

[1] https://github.com/MASILab/3DUX-Net.

(or cysts). The challenge is from the large label imbalances between the background, pancreas, and tumor structures. For all three MSD tasks, we perform 5-fold cross-validation with 70%/10%/20% train, validation, and test splits. These 20% test data will not overlap with other folds and cover all data by 5 folds.

Implementation Details

The training pipeline is based on the publicly available SwinUNETR codebase (https://github.com/Project-MONAI/research-contributions/tree/main/SwinUNETR/BTCV, our training recipe is the same as that by SwinUNETR). We changed the initial learning rate to 4e-4, and the training epoch is adapted to each task such that the total training iteration is about 40k. Random Gaussian smooth, Gaussian noise, and random gamma correction are also added as additional data augmentation. There are differences in data preprocessing across tasks. MSD data are resampled to 1×1x1 mm resolution and normalized to zero mean and standard deviation (CT images are firstly clipped by .5% and 99.5% foreground intensity percentile). For WORD and FLARE preprocessing, we use the default transforms in SwinUNETR codebase (https://github.com/Project-MONAI/research-contributions/tree/main/SwinUNETR/BTCV, our training recipe is the same as that by SwinUNETR) and 3D UXNet codebase (see footnote 1). Besides these, all other training hyperparameters are the same. We only made those minimal changes for different tasks and show surprisingly good generalizability of the SwinUNETR-V2 and the pipeline across tasks.

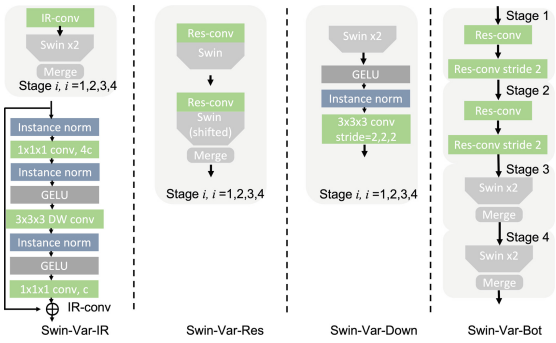

Fig. 3. Design variations for SwinUNETR-V2. Swin-Var-IR replaces the ResConv block in SwinUNETR-V2 with an inverted depth-wise convolution block. Swin-Var-Res added a ResConv block to every swin transformer block. Swin-Var-Down replaces the patch merging with a convolution block with stride 2. Swin-Var-Bot changes the top 2 stages of the encoder with ResConv blocks and only keeps transformer blocks in the higher stages.

Results

WORD Result. We follow the data split in [17] and report the test scores. All the baseline scores are from [17] except nnFormer and SwinUNETR. To make a fair comparison, we didn't use any test-time augmentation or model ensemble. The test set dice

Table 1. WORD test set Dice scores (%) and standard deviation in brackets. The best score is in bold. nnFormer, SwinUNETR and SwinUNETR-V2 results are from our codebase training, the rest is from the WORD paper [17].

Method	nnUNetV2 (2D)	ResUNet (2D)	AttUNet (3D)	nnUNet (3D)	nnUNetV2 (3D)	UNETR (3D)	nnFormer (3D)	CoTr (3D)	SwinUNETR (3D)	SwinUNETR-V2 (3D)
Liver	96.19	96.55	96.00	96.45	96.59	94.67	95.52	95.58	96.6	**96.65**(0.007)
Spleen	94.33	95.26	94.90	95.98	96.09	92.85	94.05	94.9	95.93	**96.16**(0.009)
Kidney (L)	91.29	95.63	94.65	95.40	95.63	91.49	92.8	93.26	94.93	**95.73**(0.009)
Kidney (R)	91.20	95.84	94.7	95.68	95.83	91.72	93.53	93.63	95.5	**95.91**(0.011)
Stomach	91.12	91.58	91.15	91.69	91.57	85.56	88.26	89.99	91.28	**92.31**(0.025)
Gallbladder	83.19	82.83	81.38	83.19	**83.72**	65.08	71.55	76.4	79.67	81.02(0.159)
Esophagus	77.79	77.17	76.87	**78.51**	77.36	67.71	58.89	74.37	77.68	78.36(0.122)
Pancreas	83.55	83.56	83.55	85.04	85.00	74.79	75.28	81.02	85.16	**85.51**(0.06)
Duodenum	64.47	66.67	67.68	68.31	67.73	57.56	58.76	63.58	68.11	**69.93**(0.152)
Colon	83.92	83.57	85.72	87.41	87.26	74.62	77.20	84.14	86.07	**87.46**(0.07)
Intestine	86.83	86.76	88.19	89.3	89.37	80.4	80.78	86.39	88.66	**89.71**(0.029)
Adrenal	70.0	70.9	70.23	72.38	**72.98**	60.76	57.13	69.06	70.58	71.75(0.09)
Rectum	81.49	82.16	80.47	82.41	82.32	74.06	73.42	80.0	81.73	**82.56**(0.05)
Bladder	90.15	91.0	89.71	**92.59**	92.11	85.42	86.97	89.27	91.79	91.56(0.11)
Head of Femur (L)	93.28	**93.39**	91.90	91.99	92.56	89.47	87.04	91.03	92.88	92.64 (0.04)
Head of Femur (R)	**93.93**	93.88	92.43	92.74	92.49	90.17	86.87	91.87	92.77	92.9 (0.037)
Mean	85.80	86.67	86.21	87.44	87.41	79.77	79.88	84.66	86.83	**87.51**(0.062)

Table 2. WORD test set HD95 scores and standard deviation in brackets. The best score is in bold. nnFormer, SwinUNETR, and SwinUNETR-V2 are from our codebase training, the rest is from the WORD paper [17].

Method	nnUNetV2 (2D)	ResUNet (2D)	AttUNet (3D)	nnUNet (3D)	nnUNetV2 (3D)	UNETR (3D)	nnFormer (3D)	CoTr (3D)	SwinUNETR (3D)	SwinUNETR-V2 (3D)
Liver	7.34	4.64	3.61	3.31	3.17	8.36	3.95	7.47	2.63	**2.54**(1.36)
Spleen	9.53	8.7	2.74	2.15	2.12	14.84	3.02	8.14	1.78	**1.44**(0.48)
Kidney (L)	10.33	5.4	6.28	6.07	**2.46**	23.37	9.28	16.42	5.24	5.18 (18.98)
Kidney (R)	10.85	2.47	2.86	2.35	2.24	7.9	9.69	12.79	5.77	**1.58** (0.59)
Stomach	13.97	9.98	**8.23**	8.47	9.47	19.25	11.99	10.26	9.95	8.61 (7.86)
Gallbladder	7.91	9.48	**5.11**	5.24	6.04	12.72	6.58	11.32	6.46	5.29 (7.54)
Esophagus	6.7	6.7	5.35	5.49	5.83	9.31	7.99	6.29	3.89	**3.32** (1.83)
Pancreas	7.82	7.82	6.96	6.84	6.87	10.66	7.96	8.88	4.84	**4.98** (5.66)
Duodenum	23.29	21.79	21.61	21.3	21.15	25.15	18.18	24.83	18.03	**17.13** (10.44)
Colon	15.68	17.41	10.21	9.99	10.42	20.32	15.38	12.41	9.93	**8.48** (9.28)
Intestine	8.96	9.54	5.68	5.14	5.27	12.62	8.82	7.96	5.33	**3.84** (2.33)
Adrenal	6.42	6.67	5.98	5.46	5.43	8.73	7.53	6.76	5.32	**4.81** (3.89)
Rectum	11.15	10.62	11.67	11.57	12.39	12.79	9.79	11.26	7.71	**7.16** (4.03)
Bladder	4.97	5.02	4.83	3.68	4.17	14.71	4.7	14.34	**2.38**	2.74 (4.15)
Head of Femur (L)	6.54	6.56	6.93	35.18	17.05	38.11	4.21	19.42	**2.78**	2.84 (2.45)
Head of Femur (R)	5.74	5.98	6.06	33.03	27.29	38.62	4.3	26.78	2.99	**2.79** (2.19)
Mean	9.88	8.6	7.13	10.33	8.84	17.34	8.34	12.83	5.94	**5.17** (5.19)

Table 3. FLARE 2021 5-fold cross-validation average test dice scores (on held-out test scans) and standard deviation in brackets. Baseline results from 3D UX-Net paper [14].

	3D U-Net	SegResNet	RAP-Net	nn-UNet	TransBTS	UNETR	nnFormer	SwinUNETR	3D UX-Net	SwinUNETR-V2
Spleen	0.911	0.963	0.946	0.971	0.964	0.927	0.973	0.979	**0.981**	0.980 (0.018)
Kidney	0.962	0.934	0.967	0.966	0.959	0.947	0.960	0.965	0.969	**0.973** (0.013)
Liver	0.905	0.965	0.940	0.976	0.974	0.960	0.975	0.980	0.982	**0.983** (0.008)
Pancreas	0.789	0.745	0.799	0.792	0.711	0.710	0.717	0.788	0.801	**0.851** (0.037)
Mean	0.892	0.902	0.913	0.926	0.902	0.886	0.906	0.929	0.934	**0.947** (0.019)

Table 4. MSD prostate, lung, and pancreas 5-fold cross-validation average test dice scores and standard deviation in brackets. The best score is in bold.

	Task05 Prostate			Task06 Lung	Task07 Pancreas		
	Peripheral zone	Transition zone	Avg.	Tumour (Avg.)	Pancreas	Tumour	Avg.
nnUNet2D	0.5838(0.1789)	0.8063(0.0902)	0.6950(0.1345)	–	–	–	–
nnUNet3D	0.5764(0.1697)	0.7922(0.0979)	0.6843(0.1338)	0.6067(0.2545)	0.7937(0.0882)	0.4507(0.3321)	0.6222(0.2101)
nnFormer	0.5666(0.1955)	0.7876(0.1228)	0.6771(0.1591)	0.4363(0.2080)	0.6405(0.1340)	0.3061(0.2687)	0.4733(0.2013)
UNETR	0.5440(0.1881)	0.7618(0.1213)	0.6529(0.1547)	0.2999(0.1785)	0.7262(0.1109)	0.2606(0.2732)	0.4934(0.1920)
3D-UXNet	0.6102(0.1760)	0.8410(0.0637)	0.7256(0.1198)	0.5999(0.2057)	0.7643(0.0987)	0.4374(0.2930)	0.6009(0.1959)
SwinUNETR	0.6167(0.1862)	**0.8498**(0.0518)	0.7332(0.1190)	0.5672(0.1968)	0.7546(0.0978)	0.3552(0.2514)	0.5549(0.1746)
SwinUNETR-V2	**0.6353**(0.1688)	0.8457(0.0567)	**0.7405** (0.1128)	**0.6203**(0.2012)	**0.8001**(0.0802)	**0.4805**(0.2973)	**0.6403**(0.1887)

Table 5. Dice and HD95 on WORD test set of the variations of SwinUNETR-V2.

	SwinUNETR	Swin-Var-Bot	Swin-Var-IR	Swin-Var-Res	Swin-Var-Down	SwinUNETR-V2
Dice (↑)	0.8683	0.8685	0.8713	0.8713	0.8687	0.8751
HD95 (↓)	5.94	5.18	6.64	5.3	14.04	5.17

score and 95% Hausdorff Distance (hd95) are shown in Table 1 and Table 2. We don't have the original baseline results for statistical testing (we reproduced some baseline results but the results are lower than reported), so we report the standard deviation of our methods. SwinUNETR has 62.5M parameters/295 GFlops and SwinUNETR-V2 has 72.8M parameters/320 GFlops. The baseline parameters/flops can be found in [14].

FLARE 2021 Result. We use the 5-fold cross-validation data split and baseline scores from [14]. Following [14], the five trained models are evaluated on 20 held-out test scans, and the average dice scores (not model ensemble) are shown in Table 3. We can see our SwinUNETR-V2 surpasses all the baseline methods by a large margin.

MSD Results. For MSD datasets, we perform 5-fold cross-validation and ran the baseline experiments with our codebase using exactly the same hyperparameters as mentioned. nnunet2D/3D baseline experiments are performed using nnunet's original codebase[2] since it has its own automatic hyperparameter selection. The test dice score and standard deviation (averaged over 5 fold) are shown in Table 4. We did not do any post-processing or model ensembling, thus there can be a gap between the test values and online MSD leaderboard values. We didn't compare with leaderboard results because the purpose of the experiments is to make fair comparisons, while not resorting to additional training data/pretraining, postprocessing, or model ensembling.

Variations of SwinUNetR-V2 In this section, we investigate other variations of adding convolutions into swin transformer. We follow Fig. 1 and investigate the (2.b) and (2.c) schemes, as well as the inverted convolution block. As for the (2.a) of parallel branches, it increases the GPU memory usage for 3D medical image too much and

[2] https://github.com/MIC-DKFZ/nnUNet.

we keep it for future investigation. As shown in Fig. 3, we investigate 1) Swin-Var-Bot (2.b scheme): Replacing the top 2 stages of swin transformer with ResConv block, and keeping the bottom two stages using swin blocks. 2) Swin-Var-IR: Using inverted residual blocks (with 3D depthwise convolution) instead of ResConv blocks. 3) Swin-Var-Res (2.c scheme): Instead of only adding Resconv blocks at the beginning of each stage, we create a new swin transformer block which all starts with this ResConv block, like the MOAT [31] work. 4) Swin-Var-Down: the patch merging is replaced by convolution with stride 2 like nnFormer [35]. We perform the study on the WORD dataset, and the mean test Dice and HD95 scores are shown in Table 5. We can see that adding convolution at different places does affect the performances, and the SwinUNETR-V2 design is the optimal on WORD test set.

4 Discussion and Conclusion

In this paper, we propose a new 3D medical image segmentation network SwinUNETR-V2. For some tasks, we found the original SwinUNETR with pure transformer backbones (or other ViT-based models) may have inferior performance and training stability than CNNs. To improve this, our core intuition is to combine convolution with window-based self-attention. Although existing window-based attention already has a convolution-like inductive bias, it is still not good enough for learning local details as convolutions. We tried multiple combination strategies as in Table 5 and found our current design most effective. By only adding one ResConv block at the beginning of each resolution level, the features can be well-regularized while not too constrained by the convolution inductive bias, and the computation cost will not increase by a lot. Extensive experiments are performed on a variety of challenging datasets, and SwinUNETR-V2 achieved promising improvements. The optimal combination of swin transformer and convolution still lacks a clear principle and theory, and we can only rely on trial and error in designing new architectures. We will apply the network to active challenges for more evaluation.

References

1. Antonelli, M., et al.: The medical segmentation decathlon. Nat. Commun. **13**(1), 1–13 (2022)
2. Cao, H., et al.: Swin-Unet: Unet-like pure transformer for medical image segmentation. arXiv preprint arXiv:2105.05537 (2021)
3. Chen, J., et al.: TransUNet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 (2021)
4. Chen, Y., et al.: Mobile-Former: Bridging mobileNet and transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5270–5279 (2022)
5. Dai, Z., Liu, H., Le, Q.V., Tan, M.: CoAtNet: marrying convolution and attention for all data sizes. Adv. Neural Inf. Process. Syst. **34**, 3965–3977 (2021)
6. Dosovitskiy, A., et al.: An image is worth 16×16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
7. d'Ascoli, S., Touvron, H., Leavitt, M.L., Morcos, A.S., Biroli, G., Sagun, L.: ConViT: improving vision transformers with soft convolutional inductive biases. In: International Conference on Machine Learning, pp. 2286–2296. PMLR (2021)

8. Guo, J., et al.: CMT: convolutional neural networks meet vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12175–12185 (2022)
9. Hatamizadeh, A., et al.: UNETR: transformers for 3D medical image segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574–584 (2022)
10. He, Y., Yang, D., Roth, H., Zhao, C., Xu, D.: DiNTS: differentiable neural network topology search for 3D medical image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5841–5850 (2021)
11. Huang, Z. et al.: Revisiting nnU-Net for iterative pseudo labeling and efficient sliding window inference. In: Ma, J., Wang, B. (eds.) Fast and Low-Resource Semi-supervised Abdominal Organ Segmentation. FLARE 2022. Lecture Notes in Computer Science. vol. 13816. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-23911-3_16
12. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods **18**(2), 203–211 (2021)
13. Ji, Y., et al.: AMOS: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation. arXiv preprint arXiv:2206.08023 (2022)
14. Lee, H.H., Bao, S., Huo, Y., Landman, B.A.: 3D UX-Net: A large kernel volumetric convnet modernizing hierarchical transformer for medical image segmentation. arXiv (2022)
15. Li, X., et al.: The state-of-the-art 3d anisotropic intracranial hemorrhage segmentation on non-contrast head CT: the instance challenge. arXiv preprint arXiv:2301.03281 (2023)
16. Liu, Z., et al.: Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)
17. Luo, X.: Word: a large scale dataset, benchmark and clinical applicable study for abdominal organ segmentation from CT image. Med. Image Anal. **82**, 102642 (2022)
18. Ma, J., et al.: Fast and low-GPU-memory abdomen CT organ segmentation: the flare challenge. Med. Image Anal. **82**, 102616 (2022)
19. Myronenko, A., Siddiquee, M.M.R., Yang, D., He, Y., Xu, D.: Automated head and neck tumor segmentation from 3D PET/CT. arXiv preprint arXiv:2209.10809 (2022)
20. Peng, Z., et al.: Conformer: local features coupling global representations for visual recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 367–376 (2021)
21. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
22. Siddiquee, M.M.R., Yang, D., He, Y., Xu, D., Myronenko, A.: Automated segmentation of intracranial hemorrhages from 3D CT. arXiv preprint arXiv:2209.10648 (2022)
23. Tang, Y., et al.: Self-supervised pre-training of swin transformers for 3D medical image analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20730–20740 (2022)
24. Valanarasu, J.M.J., Oza, P., Hacihaliloglu, I., Patel, V.M.: Medical transformer: gated axial-attention for medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12901, pp. 36–46. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87193-2_4
25. Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B., Shlens, J.: Scaling local self-attention for parameter efficient visual backbones. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12894–12904 (2021)

26. Wang, W., et al.: 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)
27. Wang, W., Chen, C., Ding, M., Yu, H., Zha, S., Li, J.: TransBTS: multimodal brain tumor segmentation using transformer. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12901, pp. 109–119. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87193-2_11
28. Wu, H., et al.: CvT: introducing convolutions to vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 22–31 (2021)
29. Xia, Z., Pan, X., Song, S., Li, L.E., Huang, G.: Vision transformer with deformable attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4794–4803 (2022)
30. Xie, Y., Zhang, J., Shen, C., Xia, Y.: CoTr: efficiently bridging CNN and transformer for 3D medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12903, pp. 171–180. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87199-4_16
31. Yang, C., et al.: MOAT: alternating mobile convolution and attention brings strong vision models. arXiv preprint arXiv:2210.01820 (2022)
32. Yang, D., et al.: T-AutoML: automated machine learning for lesion segmentation using transformers in 3d medical imaging. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3962–3974 (2021)
33. Zhang, Y., Liu, H., Hu, Q.: TransFuse: fusing transformers and CNNs for medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12901, pp. 14–24. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87193-2_2
34. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 633–641 (2017)
35. Zhou, H.Y., Guo, J., Zhang, Y., Yu, L., Wang, L., Yu, Y.: nnFormer: Interleaved transformer for volumetric segmentation. arXiv preprint arXiv:2109.03201 (2021)