Scaling Up 3D Kernels with Bayesian Frequency Re-parameterization for Medical Image Segmentation

Ho Hin Lee[1], Quan Liu[1], Shunxing Bao[1], Qi Yang[1], Xin Yu, Leon Y. Cai[3], Thomas Li[3], Yuankai Huo[1,2], Xenofon Koutsoukos[1], and Bennett A. Landman[2]

[1] Department of Computer Science, Vanderbilt University, Nashville, TN 37212, USA
[2] Department of Electrical and Computer Engineering, Vanderbilt University, Nashville TN 37212, USA
[3] Department of Biomedical Engineering, Vanderbilt University, Nashville TN 37212, USA

Abstract. With the inspiration of vision transformers, the concept of depth-wise convolution revisits to provide a large Effective Receptive Field (ERF) using Large Kernel (LK) sizes for medical image segmentation. However, the segmentation performance might be saturated and even degraded as the kernel sizes scaled up (e.g., $21 \times 21 \times 21$) in a Convolutional Neural Network (CNN). We hypothesize that convolution with LK sizes is limited to maintain an optimal convergence for locality learning. While Structural Re-parameterization (SR) enhances the local convergence with small kernels in parallel, optimal small kernel branches may hinder the computational efficiency for training. In this work, we propose RepUX-Net, a pure CNN architecture with a simple large kernel block design, which competes favorably with current network state-of-the-art (SOTA) (e.g., 3D UX-Net, SwinUNETR) using 6 challenging public datasets. We derive an equivalency between kernel re-parameterization and the branch-wise variation in kernel convergence. Inspired by the spatial frequency in the human visual system, we extend to vary the kernel convergence into element-wise setting and model the spatial frequency as a Bayesian prior to re-parameterize convolutional weights during training. Specifically, a reciprocal function is leveraged to estimate a frequency-weighted value, which rescales the corresponding kernel element for stochastic gradient descent. From the experimental results, RepUX-Net consistently outperforms 3D SOTA benchmarks with internal validation (FLARE: 0.929 to 0.944), external validation (MSD: 0.901 to 0.932, KiTS: 0.815 to 0.847, LiTS: 0.933 to 0.949, TCIA: 0.736 to 0.779) and transfer learning (AMOS: 0.880 to 0.911) scenarios in Dice Score. Both codes and pretrained models are available at: `https://github.com/MASILab/RepUX-Net`

Keywords: Bayesian Frequency Re-parameterization, Large Kernel Convolution, Medical Image Segmentation.

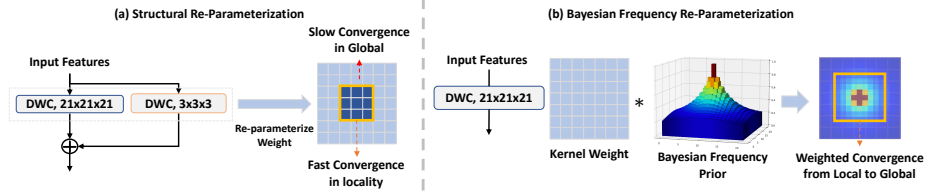arXiv:2303.05785v2 [eess.IV] 6 Jun 2023

Fig. 1: With the fast convergence in small kernels, SR merges the branches weights and enhances the locality convergence with respect to the kernel size (deep blue region), while the global convergence is yet to be optimal (light blue region). By adapting BFR, the learning convergence can rescale in an element-wise setting and distribute the learning importance from local to global.

1 Introduction

With the introduction of Vision Transformers (ViTs), CNNs have been greatly challenged as seen with the leading performance in multiple volumetric data benchmarks, especially for medical image segmentation [7, 8, 21, 23]. The key contribution of ViTs is largely credited to the large Effective Receptive Field (ERF) with a multi-head self-attention mechanism [6]. Note the attention mechanism is computationally unscalable with respect to the input resolutions [17,18]. Therefore, the concept of depth-wise convolution is revisited to provide a scalable and efficient feature computation with large ERF using large kernel sizes (e.g., $7 \times 7 \times 7$) [14, 18]. However, either from prior works or our experiments, the model performance becomes saturated or even degraded when the kernel size is scaled up in encoder blocks [4, 16]. We hypothesize that scaling up the kernel size in convolution may limit the optimal learning convergences across local to global scales. Recently, the feasibility of leveraging large kernel convolutions (e.g., 31×31 [4], 51×51 [16]) has been shown with natural image domain with Structural Re-parameterization (SR), which adapts Constant-Scale Linear Addition (CSLA) block (Fig. 2b) and re-parameterizes the large kernel weights during inference [4]. As convolutions with small kernel sizes converge more easily, the convergence of small kernel regions enhances in the re-parameterized weight, as shown in Fig. 1a. With such observation, we further ask: **Can we adapt variable convergence across elements of the convolution kernel during training, instead of regional locality only?**

In this work, we first derive and extend the theoretical equivalency of the weight optimization in the CSLA block. We observe that the kernel weight of each branch can be optimized with variable convergence using branch-specific learning rates. Furthermore, the ERF with SR is visualized to be more widely distributed from the center element to the global surroundings [4], demonstrating a similar behavior to the spatial frequency in the human visual system [13]. Inspired by the reciprocal characteristics of spatial frequency, we model the spatial frequency as a Bayesian prior to adapt variable convergence of each kernel element with stochastic gradient descent (Fig. 1b). Specifically, we compute a
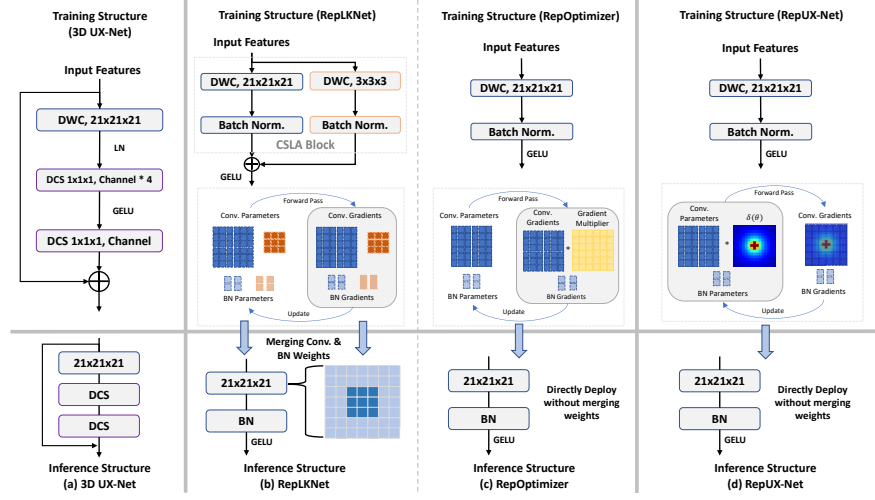
Fig. 2: Overview of RepUX-Net. Unlike performing SR to merge branches weight or performing GR within optimizers, we propose to multiply a Bayesian function δ and scale the element-wise learning importance in each large kernel. We then put the scaled weights back into the convolution layer for training.

scaling factor with respect to the distance from the kernel center and multiply the corresponding element for re-parameterization during training. Furthermore, we simplify the encoder block design into a plain convolution block only to minimize the computation burden in training and achieve State-Of-The-Art (SOTA) performance. We propose RepUX-Net, a pure 3D CNN with the large kernel size (e.g., $21 \times 21 \times 21$) in encoder blocks, to compete favorably with current SOTA segmentation networks. We evaluate RepUX-Net on supervised multi-organ segmentation with 6 different public volumetric datasets. RepUX-Net demonstrates significant improvement consistently across all datasets compared to all SOTA networks. We summarize our contributions as below:

- We propose RepUX-Net with better adaptation in large kernel convolution than 3D UX-Net, achieving SOTA performance in 3D segmentation. To our best knowledge, this is the first network that effectively leverages large kernel convolution with plain design in the encoder for 3D segmentation.
- We propose a novel theory-inspired re-parameterization strategy to scale the element-wise learning convergence in large kernels with Bayesian prior knowledge. To our best knowledge, this is the first re-parameterization strategy to adapt 3D large kernels in the medical domain.
- We leverage six challenging public datasets to evaluate RepUX-Net in 1) direct training and 2) transfer learning scenarios with 3D multi-organ segmentation. RepUX-Net achieves significant improvement consistently in both scenarios across all SOTA networks.

2   Related Works

Weights Re-parameterization: SR is a methodology of equivalently converting model structures via transforming the parameters in kernel weights. For example, RepVGG demonstrates to construct one extra ResNet-style shortcut as a 1×1 convolution, parallel to 3×3 convolution during training [5]. Such parallel branch design is claimed to enhance the learning efficiency during training, in which the 1x1 branch is then merged into the parallel 3×3 kernel via a series of linear transformation in the inference stage. OREPA further adds more parallel branches with linear scaling modules to enhance training efficiency [10]. Inspired by the parallel branches design, RepLKNet is proposed to scale up the 2D kernel size (e.g., 31x31) with a 3x3 convolution as the parallel branch [4]. SLaK further extends the kernel size to 51x51 by decomposing the large kernel into two rectangular parallel kernels with sparse groups and training the model with dynamic sparsity [16]. However, the proposed models' FLOPs remain at a high-level with the parallel branch design and demonstrates to have a trade-off between model performance and training efficiency. To tackle the trade-off, RepOptimizer provides an alternative to re-parameterize the back-propagate gradient, instead of the structural parameters of kernel weights, to enhance the training efficiency with plain convolution block design [3]. Significant efforts have been demonstrated to enlarge the 2D kernel size in the natural image domain, while limited studies have been proposed for 3D kernels in medical domain. As 3D kernels have a larger number of parameters than 2D, it is challenging to directly leverage the parallel branch design and maintain an optimal convergence of learning large kernel convolution without trading off the computation efficiency significantly.

3   Methods

Instead of changing the gradient dynamics during training [3], we introduce RepUX-Net, a pure 3D CNN architecture that performs element-wise scaling in large kernel weights to enhance the learning convergence and effectively adapts large receptive field for volumetric segmentation. To design such behavior, we adapt a two-step pipeline: 1) we define the theoretical equivalency of variable learning convergence in convolution branches; 2) we simulate the behavior of spatial frequency to re-weight the learning importance of each element in kernels for stochastic gradient descent. Note the theoretical derivation depends on the optimization with first-order gradient-driven optimizer (e.g., SGD, AdamW) [3].

3.1   Variable Learning Convergence in Multi-Branch Design

From Figure 2, the learning convergence of the large kernel convolution can be improved by either adding up the encoded outputs of parallel branches weighted by diverse scales with SR (RepLKNet [4]) or performing Gradient Re-parameterization (GR) by multiplying with constant values (RepOptimizer [3]) in a Single Operator (SO). Inspired by the concepts of SR and GR, we extend

the equivalency proof in RepOptimizer to adapt variable learning convergence in branches. Here, we only showcase the conclusion with two convolutions and two constant scalars as the scaling factors for simplicity. The complete proof of equivalency is demonstrated in Supplementary 1.1. Let $\{\alpha_L, \alpha_S\}$ and $\{W_L, W_S\}$ be the two constant scalars and two convolution kernels (Large & Small) respectively. Let X and Y be the input and output features, the CSLA block is formulated as $Y_{CSLA} = \alpha_L(X \star W_L) + \alpha_S(X \star W_S)$, where $\star$ denotes as convolution. For SO blocks, we train the plain structure parameterized by W' and $Y_{SO} = X \star W'$. Let i be the number of training iterations, we ensure that $Y^{(i)}{}_{CSLA} = Y^{(i)}{}_{SO}, \forall i \geq 0$ and derive the stochastic gradient descent of parallel branches as follows:

$$\alpha_L W_{L(i+1)} + \alpha_S W_{S(i+1)} = \alpha_L W_{L(i)} - \lambda_L \alpha_L \frac{\partial \mathcal{L}}{\partial W_{L_i}} + \alpha_S W_{S(i)} - \lambda_S \alpha_S \frac{\partial \mathcal{L}}{\partial W_{S_i}}, \ (1)$$

where $\mathcal{L}$ is the objective function; λ_L and λ_S are the Learning Rate (LR) of each branch respectively. We observe that the optimization of each branch can be different by adjusting the branch-specific LR. The locality convergence in large kernels enhance with the quick convergence in small kernels. Additionally from our experiments, a significant improvement is demonstrated with different branch-wise LR using SGD (Table 2). With such observation, we further hypothesize that **the convergence of each large kernel element can be optimized differently by linear scaling with prior knowledge**.

3.2   Bayesian Frequency Re-parameterization (BFR)

With the visualization of ERF in RepLKNet [4], the diffused distribution (from local to global) in ERF demonstrates similar behavior with the spatial frequency in the human visual system [13]. High spatial frequency (small ERF) allows to refine and sharpen details with high acuity, while global details are demonstrated with low spatial frequency. Inspired by the reciprocal characteristics in spatial frequency, we first generate a Bayesian prior distribution to model the spatial frequency by computing a reciprocal distance function between each element and the central point of the kernel weight as follows:

$$d(x, y, z, c) = \sqrt{(x - c)^2 + (y - c)^2 + (z - c)^2}$$
$$\delta(x_k, y_k, z_k, c, \alpha) = \frac{\alpha}{d(x_k, y_k, z_k, c) + \alpha} \tag{2}$$

where k and c are the element and central index of the kernel weight, α is the hyperparameter to control the shape of the generated frequency distribution. Instead of adjusting the LR in parallel branches, we propose to re-parameterize the convolution weights by multiplying the scaling factor δ to each kernel element and apply a static LR λ for stochastic gradient descent in single operator setting as follows:

$$W'_{i+1} = \delta W'_i - \lambda \frac{\partial L}{\partial \delta W'_i} \tag{3}$$

With the multiplication with δ, each element in the kernel weight is rescaled with respect to the frequency level and allow to converge differently with a static LR in stochastic gradient descent. Such design demonstrates to influence the weighted convergence diffused from local to global in theory, thus tackling the limitation of enhancing the local convergence only in branch-wise setting.

3.3   Model Architecture

The backbone of RepUX-Net is based on 3D UX-Net [14], which comprises multiple volumetric convolution blocks that directly utilize 3D patches and leverage skip connections to transfer hierarchical multi-resolution features for end-to-end optimization. Inspired by [15], we choose a kernel size of $21 \times 21 \times 21$ for Depth-Wise Convolution (DWC-21) as the optimal choice without significant trade-off between model performance and computational efficiency in 3D. We further simplify the block design as a plain convolution block design to minimize the computational burden from additional modules. The encoder blocks in layers l and $l+1$ are defined as follows:

$$\hat{z}^l = \text{GeLU}(\text{DWC-21}(\text{BN}(z^{l-1}))), \ \hat{z}^{l+1} = \text{GeLU}(\text{DWC-21}(\text{BN}(z^l))) \quad (4)$$

where $\hat{z}_l$ and $\hat{z}_{l+1}$ are the outputs from the DWC layer in each depth level; BN denotes as the batch normalization layer.

4   Experimental Setup

Datasets We perform experiments on six public datasets for volumetric segmentation, which comprise with 1) Medical Segmentation Decathlon (MSD) spleen dataset [1], 2) MICCAI 2017 LiTS Challenge dataset (LiTS) [2], 3) MICCAI 2019 KiTS Challenge dataset (KiTS) [9], 4) NIH TCIA Pancreas-CT dataset (TCIA) [20], 5) MICCAI 2021 FLARE Challenge dataset (FLARE) [19], and 6) MICCAI 2022 AMOS challenge dataset (AMOS) [12]. More details of each dataset (including data split for training and inference) are described in Supplementary Material (SM) Table 1.

Implementation We evaluate RepUX-Net with three different scenarios: 1) internal validation with direct supervised learning, 2) external validation with the unseen datasets, and 3) transfer learning with pretrained weights. All preprocessing and training details including baselines, are followed with [14] for benchmarking. For external validations, we leverage the AMOS-pretrained weights to evaluate 4 unseen datasets. In summary, we evaluate the segmentation performance of RepUX-Net by comparing current SOTA networks in a fully-supervised setting. Furthermore, we perform ablation studies to investigate the effect on Bayesian frequency distribution with different scales generated by α and the variability of branch-wise learning rates with first-order gradient optimizers (e.g., SGD, AdamW) for volumetric segmentation. Dice similarity coefficient is leveraged as an evaluation metric to measure the overlapping regions between the model predictions and the manual ground-truth labels.

Table 1: Comparison of SOTA approaches on the five different testing datasets. (*: $p < 0.01$, with Paired Wilcoxon signed-rank test to all baseline networks)

| Methods | #Params | FLOPs | Internal Testing FLARE | | | | | External Testing | | | |
			Spleen	Kidney	Liver	Pancreas	Mean	MSD Spleen	KiTS Kidney	LiTS Liver	TCIA Pancreas
nn-UNet [11]	31.2M	743.3G	0.971	0.966	0.976	0.792	0.926	0.917	0.829	0.935	0.739
TransBTS [22]	31.6M	110.4G	0.964	0.959	0.974	0.711	0.902	0.881	0.797	0.926	0.699
UNETR [8]	92.8M	82.6G	0.927	0.947	0.960	0.710	0.886	0.857	0.801	0.920	0.679
nnFormer [23]	149.3M	240.2G	0.973	0.960	0.975	0.717	0.906	0.880	0.774	0.927	0.690
SwinUNETR [7]	62.2M	328.4G	0.979	0.965	0.980	0.788	0.929	0.901	0.815	0.933	0.736
3D UX-Net (k=7) [14]	53.0M	639.4G	0.981	0.969	0.982	0.801	0.934	0.926	0.836	0.939	0.750
3D UX-Net (k=21) [14]	65.9M	757.6G	0.980	0.968	0.979	0.795	0.930	0.908	0.808	0.929	0.720
RepOptimizer [3]	65.8M	757.4G	0.981	0.969	0.981	0.822	0.937	0.913	0.833	0.934	0.746
3D RepUX-Net (Ours)	65.8M	757.4G	**0.984**	**0.970**	**0.983**	**0.837**	**0.944***	**0.932***	**0.847***	**0.949***	**0.779***

Table 2: Ablation studies with quantitative Comparison on Block Designs with/out frequency modeling using different optimizer

Optimizer	Main Branch	Para. Branch	BFR	Train Steps	Main LR	Para. LR	Mean Dice
SGD	$21 \times 21 \times 21$	$\times$	$\times$	40000	0.0003	$\times$	0.898
AdamW	$21 \times 21 \times 21$	$\times$	$\times$	40000	0.0001	$\times$	0.906
SGD	$21 \times 21 \times 21$	$3 \times 3 \times 3$	$\times$	40000	0.0003	0.0006	0.917
AdamW	$21 \times 21 \times 21$	$3 \times 3 \times 3$	$\times$	40000	0.0001	0.0001	0.929
AdamW	$21 \times 21 \times 21$	$\times$	$\checkmark$	40000	0.0001	$\times$	**0.938**
SGD	$21 \times 21 \times 21$	$3 \times 3 \times 3$	$\times$	60000	0.0003	0.0006	0.930
AdamW	$21 \times 21 \times 21$	$3 \times 3 \times 3$	$\times$	60000	0.0001	0.0001	0.938
AdamW	$21 \times 21 \times 21$	$\times$	$\checkmark$	60000	0.0001	$\times$	**0.944**

5   Results

Different Scenarios Evaluations. Table 1 shows the result comparison of current SOTA networks on medical image segmentation in a volumetric setting. With our designed convolutional blocks as the encoder backbone, RepUX-Net demonstrates the best performance across all segmentation task with significant improvement in Dice score (FLARE: 0.934 to 0.944, AMOS: 0.891 to 0.902). Furthermore, RepUX-Net demonstrates the best generalizability consistently with a significant boost in performance across 4 different external datasets (MSD: 0.926 to 0.932, KiTS: 0.836 to 0.847, LiTS: 0.939 to 0.949, TCIA: 0.750 to 0.779). Furthermore, from Figure 2A, RepUX-Net demonstrates the quickest convergence rate in training with AMOS datasets from scratch. For transfer learning scenario, the performance of RepUX-Net significantly outperforms the current SOTA networks with mean Dice of 0.911 (1.22% enhancement), as shown in Table 2. RepUX-Net demonstrates its capabilities across the generalizability of unseen datasets and transfer learning ability. The qualitative representations (in SM Figure 1) further provides additional confidence of the quality improvement in segmentation predictions with RepUX-Net.

Ablation studies with block designs & optimizers. With the plain convolution design, a mean dice score of 0.906 is demonstrated with AdamW optimizer

Table 3: Evaluations on the AMOS testing split in different scenarios.(*: $p <$ 0.01, with Paired Wilcoxon signed-rank test to all baseline networks)

	Spleen	R. Kid	L. Kid	Gall.	Eso.	Liver	Stom.	Aorta	IVC	Panc.	RAG	LAG	Duo.	Blad.	Pros.	Avg
Train From Scratch Scenario																
Methods																
nn-UNet	0.951	0.919	0.930	0.845	0.797	0.975	0.863	0.941	0.898	0.813	0.730	0.677	0.772	0.797	0.815	0.850
TransBTS	0.930	0.921	0.909	0.798	0.722	0.966	0.801	0.900	0.820	0.702	0.641	0.550	0.684	0.730	0.679	0.783
UNETR	0.925	0.923	0.903	0.777	0.701	0.964	0.759	0.887	0.821	0.687	0.688	0.543	0.629	0.710	0.707	0.740
nnFormer	0.932	0.928	0.914	0.831	0.743	0.968	0.820	0.905	0.838	0.725	0.678	0.578	0.677	0.737	0.596	0.785
SwinUNETR	0.956	0.957	0.949	0.891	0.820	0.978	0.880	0.939	0.894	0.818	0.800	0.730	0.803	0.849	0.819	0.871
3D UX-Net (k=7)	0.966	0.959	0.951	0.903	0.833	0.980	0.910	0.950	0.913	0.830	0.805	0.756	**0.846**	0.897	0.863	0.890
3D UX-Net (k=21)	0.963	0.959	0.953	**0.921**	0.848	0.981	0.903	0.953	0.910	0.828	0.815	0.754	0.824	0.900	0.878	0.891
RepOptimizer	0.968	**0.964**	0.953	0.903	0.857	0.981	0.915	0.950	0.915	0.826	0.802	0.756	0.813	0.906	0.867	0.892
RepUX-Net (Ours)	**0.972**	0.963	**0.964**	0.911	**0.861**	**0.982**	**0.921**	**0.956**	**0.924**	**0.837**	**0.818**	**0.777**	0.831	**0.916**	**0.879**	**0.902***
Transfer Learning Scenario																
Methods																
nn-UNet	0.965	0.959	0.951	0.889	0.820	0.980	0.890	0.948	0.901	0.821	0.785	0.739	0.806	0.869	0.839	0.878
TransBTS	0.885	0.931	0.916	0.817	0.744	0.969	0.837	0.914	0.855	0.724	0.630	0.566	0.704	0.741	0.650	0.792
UNETR	0.926	0.936	0.918	0.785	0.702	0.969	0.788	0.893	0.828	0.732	0.717	0.554	0.658	0.683	0.722	0.762
nnFormer	0.935	0.904	0.887	0.836	0.712	0.964	0.798	0.901	0.821	0.734	0.665	0.587	0.641	0.744	0.714	0.790
SwinUNETR	0.959	0.960	0.949	0.894	0.827	0.979	0.899	0.944	0.899	0.828	0.791	0.745	0.817	0.875	0.841	0.880
3D UX-Net (k=7)	0.970	0.967	0.961	0.923	0.832	0.984	0.920	0.951	0.914	0.856	0.825	0.739	0.853	0.906	0.876	0.900
3D UX-Net (k=21)	0.969	0.965	0.962	0.910	0.824	0.982	0.918	0.949	0.915	0.850	0.823	0.740	0.843	0.905	0.877	0.898
RepOptimizer	0.967	0.967	0.957	0.908	0.847	0.983	0.913	0.945	0.914	0.838	0.825	0.780	0.836	0.915	0.864	0.897
RepUX-Net	**0.973**	**0.968**	**0.965**	**0.933**	**0.865**	**0.985**	**0.930**	**0.960**	**0.923**	**0.859**	**0.829**	**0.793**	**0.869**	**0.918**	**0.891**	**0.911***

and perform slightly better than that with SGD. With the additional design of a parallel small kernel branch, the segmentation performance significantly improved (SGD: 0.898 to 0.917, AdamW: 0.906 to 0.929) with the optimized parallel branch LR using SR. The performance is further enhanced (SGD: 0.917 to 0.930, AdamW: 0.929 to 0.937) without being saturated with the increase of the training steps. By adapting BFR, the segmentation performance outperforms the parallel branch design significantly with a Dice score of 0.944.

Effectiveness on Different Frequency Distribution. From Figure 2 in SM, RepUX-Net demonstrates the best performance when $\alpha = 1$, while comparable performance is demonstrated in both $\alpha = 0.5$ and $\alpha = 8$. A possible family of Bayesian distributions (different shapes) may need to further optimize the learning convergence of kernels across each channel.

Limitations. The shape of the generated Bayesian distribution is fixed across all kernel weights with an unlearnable distance function. Each channel in kernels is expected to extract variable features with different distributions. Exploring different families of distributions to rescale the element-wise convergence in kernels will be our potential future direction.

6 Conclusion

We introduce RepUX-Net, the first 3D CNN adapting extreme large kernel convolution in encoder network for medical image segmentation. We propose to model the spatial frequency in the human visual system as a reciprocal function, which generates a Bayesian prior to rescale the learning convergence of each ele-

ment in kernel weights. By introducing the frequency-guided importance during training, RepUX-Net outperforms current SOTA networks on six challenging public datasets via both direct training and transfer learning scenarios.

References

1. Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B.A., Litjens, G., Menze, B., Ronneberger, O., Summers, R.M., et al.: The medical segmentation decathlon. Nature communications **13**(1), 4128 (2022)
2. Bilic, P., Christ, P., Li, H.B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., Szeskin, A., Jacobs, C., Mamani, G.E.H., Chartrand, G., et al.: The liver tumor segmentation benchmark (lits). Medical Image Analysis **84**, 102680 (2023)
3. Ding, X., Chen, H., Zhang, X., Huang, K., Han, J., Ding, G.: Re-parameterizing your optimizers rather than architectures. arXiv preprint arXiv:2205.15242 (2022)
4. Ding, X., Zhang, X., Han, J., Ding, G.: Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11963–11975 (2022)
5. 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)
6. 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)
7. Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H.R., Xu, D.: Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In: International MICCAI Brainlesion Workshop. pp. 272–284. Springer (2022)
8. 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)
9. Heller, N., McSweeney, S., Peterson, M.T., Peterson, S., Rickman, J., Stai, B., Tejpaul, R., Oestreich, M., Blake, P., Rosenberg, J., et al.: An international challenge to use artificial intelligence to define the state-of-the-art in kidney and kidney tumor segmentation in ct imaging. (2020)
10. Hu, M., Feng, J., Hua, J., Lai, B., Huang, J., Gong, X., Hua, X.S.: Online convolutional re-parameterization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 568–577 (2022)
11. 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. Nature methods **18**(2), 203–211 (2021)
12. Ji, Y., Bai, H., Yang, J., Ge, C., Zhu, Y., Zhang, R., Li, Z., Zhang, L., Ma, W., Wan, X., et al.: Amos: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation. arXiv preprint arXiv:2206.08023 (2022)
13. Kulikowski, J.J., Marčelja, S., Bishop, P.O.: Theory of spatial position and spatial frequency relations in the receptive fields of simple cells in the visual cortex. Biological cybernetics **43**(3), 187–198 (1982)
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 preprint arXiv:2209.15076 (2022)

15. Li, H., Nan, Y., Del Ser, J., Yang, G.: Large-kernel attention for 3d medical image segmentation. arXiv preprint arXiv:2207.11225 (2022)
16. Liu, S., Chen, T., Chen, X., Chen, X., Xiao, Q., Wu, B., Pechenizkiy, M., Mocanu, D., Wang, Z.: More convnets in the 2020s: Scaling up kernels beyond 51x51 using sparsity. arXiv preprint arXiv:2207.03620 (2022)
17. 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)
18. 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)
19. Ma, J., Zhang, Y., Gu, S., Zhu, C., Ge, C., Zhang, Y., An, X., Wang, C., Wang, Q., Liu, X., Cao, S., Zhang, Q., Liu, S., Wang, Y., Li, Y., He, J., Yang, X.: Abdomenct-1k: Is abdominal organ segmentation a solved problem? IEEE Transactions on Pattern Analysis and Machine Intelligence (2021). https://doi.org/10.1109/TPAMI.2021.3100536
20. Roth, H.R., Lu, L., Farag, A., Shin, H.C., Liu, J., Turkbey, E.B., Summers, R.M.: Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part I 18. pp. 556–564. Springer (2015)
21. Tang, Y., Yang, D., Li, W., Roth, H.R., Landman, B., Xu, D., Nath, V., Hatamizadeh, A.: 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)
22. Wang, W., Chen, C., Ding, M., Yu, H., Zha, S., Li, J.: Transbts: Multimodal brain tumor segmentation using transformer. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 109–119. Springer (2021)
23. 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)

7 Supplementary Material

7.1 Derivation of Variable Convergence in Multi-Branch Design

The parallel structural design is referred to the CSLA block. Each branch only comprises on differentiable linear operator with trainable parameters (e.g., Convolution (Conv), Fully-Connected (FC) layer, scaling layer) and no training-time non-linearity. We begin with a simple case where the CSLA block has two parallel Conv kernels with same dimensions in kernel weights by padding and scaled by constant values. Let α_L, α_S and W_L, W_S be the constant scalars and the weights of two Conv kernels (Large & Small), and X and Y be the input and the output features. The computation flow of the CSLA block is formulated as following:

$$Y_{CSLA} = \alpha_L(X * W_L) + \alpha_S(X * W_S), \tag{5}$$

where $*$ denotes convolution. To optimize the gradient in parallel structure as a Single Operator (SO), we first derive the gradient flow scenario in a single operator structure, which is parameterized by $W^{'}$ as follows:

$$Y_{SO} = X * W^{'} \tag{6}$$

Our goal is to ensure generating same outputs in both multi-branch and single operator setting $Y_{CSLA} = Y_{SO}$ during training. Since only linear operations are performed within branches, we derive the kernel weights relationship as follows:

$$W^{'} = \alpha_L W_{L(i)} + \alpha_S W_{S(i)}. \tag{7}$$

With the above theoretical relationship in kernel weights, we apply the stochastic gradient descent rule and update the parallel branches gradient as follows:

$$W^{'}_{i+1} = W^{'}_i - \lambda \frac{\partial \mathcal{L}}{\partial W^{'}_i} \tag{8}$$

$$\alpha_L W_{L(i+1)} + \alpha_S W_{S(i+1)} = \alpha_L W_{L(i)} + \alpha_S W_{S(i)} - \lambda(\alpha_L \frac{\partial \mathcal{L}}{\partial W_{L_i}} + \alpha_S \frac{\partial \mathcal{L}}{\partial W_{S_i}}) \tag{9}$$

where $\mathcal{L}$ is the differentiable loss function, i is the index number of training iterations and λ is the Learning Rate (LR). We further expand equation 5 and observe that each conv branch can be optimized with different convergence rate by adjusting the corresponding LR as follows:

$$\alpha_L W_{L(i+1)} + \alpha_S W_{S(i+1)} = \alpha_L W_{L(i)} - \lambda_L \alpha_L \frac{\partial \mathcal{L}}{\partial W_{L_i}} + \alpha_S W_{S(i)} - \lambda_S \alpha_S \frac{\partial \mathcal{L}}{\partial W_{S_i}}, \tag{10}$$

where λ_L and λ_S are the LR for the large kernel branch and small kernel branch respectively. After training, the small kernel parameters are merged onto the central point of the large kernels, which is equivalent to enhance the learning convergence in locality with single operator setting.

Table 4: Complete overview of six public MICCAI challenge datasets

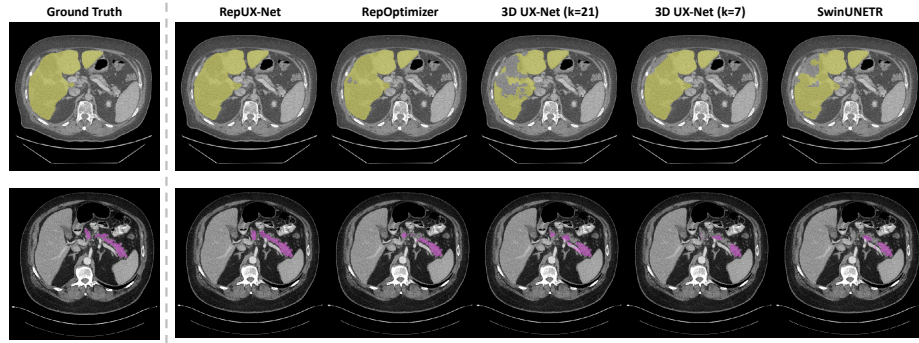| Challenge	FLARE	AMOS	MSD	KiTS	LiTS	TCIA
Imaging Modality	Multi-Contrast CT	Multi-Contrast CT	Venous CT	Arterial CT	Venous CT	Venous CT
Anatomical Region	Abdomen	Abdomen	Spleen	Kidney	Liver	Pancreas
Sample Size	361	200	41	300	131	89
Anatomical Label	Spleen, Kidney, Liver, Pancreas	Spleen, Left & Right Kidney, Gall Bladder, Esophagus, Liver, Stomach, Aorta, Inferior Vena Cava (IVC) Pancreas, Left & Right Adrenal Gland (AG), Duodenum	Spleen	Kidney, Tumor	Liver, Tumor	Pancreas
Data Splits	5-Fold Cross-Validation (Internal) Train: 272 / Validation: 69 / Test: 20	1-Fold (Internal) Train: 160 / Validation: 20 / Test: 20	All (External) Test: 41	Test: 300	Test: 131	Test: 89

Fig. 3: Qualitative Representations of organ segmentation in LiTS and TCIA datasets
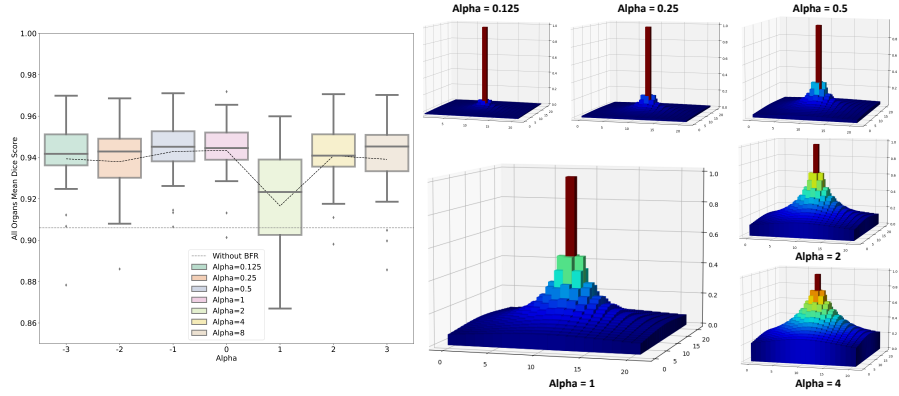

Fig. 4: Quantitative evaluation of the ablation study with different frequency distribution.