Parameter Efficient Fine-tuning via Cross Block Orchestration for Segment Anything Model

Supplementary Material

A. Derivation of the Definition

Definition 4.1. (T-product) For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\mathcal{B} \in \mathbb{R}^{n_2 \times l \times n_3}$, the T-product $\mathcal{C} \in \mathbb{R}^{n_1 \times l \times n_3} = \mathcal{A} * \mathcal{B}$ is defined as:

$$\mathcal{C} = \mathcal{A} * \mathcal{B} = \mathtt{fold}(\mathtt{bcirc}(\mathcal{A}) \cdot \mathtt{unfold}(\mathcal{B})), \tag{S-1}$$

where

$$\mathtt{bcric}(\mathcal{A}) = \begin{bmatrix} \mathbf{A}^{(1)} & \mathbf{A}^{(n_3)} & \cdots & \mathbf{A}^{(2)} \\ \mathbf{A}^{(2)} & \mathbf{A}^{(1)} & \cdots & \mathbf{A}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{(n_3)} & \mathbf{A}^{(n_3-1)} & \cdots & \mathbf{A}^{(1)} \end{bmatrix}, \tag{S-2}$$

$$\mathtt{unfold}(\mathcal{A}) = [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \cdots, \mathbf{A}^{(n_3)}]^T, \tag{S-3}$$

$$\mathtt{fold}(\mathtt{unfold}(\mathcal{A})) = \mathcal{A}, \tag{S-4}$$

$\mathbf{A}^{(i)}$ denotes the i-th frontal slice $\mathcal{A}(:,:,i)$ of $\mathcal{A}$.

Derivation. According to [25], the block circulant matrix in Eq. (S-2) can be block diagonalized by using Discrete Fourier Transform (DFT) matrix $\mathbf{F}_{n_3}$ as:

$$(\mathbf{F}_{n_3} \circ \mathbf{I}_{n_1}) \cdot \mathtt{bcric}(\mathcal{A}) \cdot (\mathbf{F}_{n_3}^{-1} \circ \mathbf{I}_{n_1}) = \bar{\mathbf{A}} \tag{S-5}$$

where

$$\bar{\mathbf{A}} = \begin{bmatrix} \bar{\mathbf{A}}^{(1)} & & \\ & \ddots & \\ & & \bar{\mathbf{A}}^{(n_3)} \end{bmatrix} \in \mathbb{R}^{n_1 n_3 \times n_2 n_3} \tag{S-6}$$

is a block diagonal matrix and its i-th block $\bar{\mathbf{A}}^{(i)}$ is the i-th frontal slice of tensor $\bar{\mathcal{A}}$ which can be obtained by performing DFT of $\mathcal{A}$ along the 3-rd dimension, $\circ$ denotes the Kronecker product. According to the definition of the frontal-slice-wise product, the T-product in Eq. (S-1) is equivalent to the matrix-matrix product in the DFT domain. In mathematics, the DET of $\mathcal{A}$ is formulated as: $\bar{\mathcal{A}} = \mathtt{DFT}(\mathcal{A}) = \mathcal{A} \times_3 \mathbf{F}_{n_3}$. Similarly, the inverse DFT of $\bar{\mathcal{A}}$ is derived as: $\mathcal{A} = \mathtt{DFT}^{-1}(\bar{\mathcal{A}}) = \bar{\mathcal{A}} \times_3 \mathbf{F}_{n_3}^{-1}$. By the detailed theoretical analysis in [14], the DFT has been extended to a general invertible linear transform S with an invertible linear transform matrix $\mathbf{S}$. In mathematics, the invertible linear transform of $\mathcal{A}$ is formulated as: $\bar{\mathcal{A}} = \mathtt{S}(\mathcal{A}) = \mathcal{A} \times_3 \mathbf{S}$. Similarly, the inverse transform of $\bar{\mathcal{A}}$ is derived as: $\mathcal{A} = \mathtt{S}^{-1}(\bar{\mathcal{A}}) = \bar{\mathcal{A}} \times_3 \mathbf{S}$. ■

B. Learning Algorithm of HL

Parametrization & Forward Propagation. The hypercomplex net (HL) differs from the real-valued linear net (i.e., a linear projection head) in the generation of its parameters $\mathbf{W} \in \mathbb{R}^{V \times V}$. In the HL framework, we define $\widetilde{H_a}$ and $\widetilde{H_b}$ as the two weights of an element H_i, as follows:

$$\widetilde{H_a} = a_0 1 + a_1 j_1 + \cdots + a_{N-1} j_{N-1} \tag{S-7}$$

$$\widetilde{H_b} = b_0 1 + b_1 j_1 + \cdots + b_{N-1} j_{N-1}. \tag{S-8}$$

Then, we can update H_i via `Hamilton product`, which is formulated as follows:

$$\begin{aligned} H_i =& \widetilde{H_a} \otimes \widetilde{H_b} \\ =& (a_0 b_0 + \cdots + a_0 b_{N-1} j_{N-1}) 1 + \\ & (a_1 b_0 + \cdots + a_1 b_{N-1} j_{N-1}) j_1 + \\ & \cdots \\ & (a_{N-1} b_0 + \cdots + a_{N-1} b_{N-1} j_{N-1}) j_{N-1}. \end{aligned} \tag{S-9}$$

Denote $c_0 = a_0 b_0 + \cdots + a_0 b_{N-1}$, $c_1 = a_1 b_0 + \cdots + a_1 b_{N-1}$, $\cdots$, $c_{N-1} = a_{N-1} b_0 + \cdots + a_{N-1} b_{N-1}$. Following the specific rule in [7], we amalgamate these coefficients into a real-valued matrix, subsequently deriving $\mathbf{W}$ as follows:

$$\mathbf{W} = \begin{bmatrix} c_0 & -c_1 & \cdots & c_{N-1} \\ c_1 & c_2 & \cdots & c_0 \\ \cdots & \cdots & \cdots & \cdots \\ c_{N-1} & -c_0 & \cdots & c_{N-2}, \end{bmatrix} \tag{S-10}$$

where N is usually less than or equal to V, $\mathbf{C}_i \in \mathbb{R}^{\frac{V}{N} \times \frac{V}{N}}$. Considering $\mathbf{M}$ and $\widetilde{\mathbf{M}}$ as the input and output respectively, the forward propagation through the linear projection head $\mathcal{F}$, parameterized by $\mathbf{W}$, is formulated as follows:

$$\widetilde{\mathbf{M}} = \mathcal{F}(\mathbf{M}; \mathbf{W}). \tag{S-11}$$

Backward Propagation. In the backward propagation process of the HL, we need to update each weight. To this end, we define the gradient *w.r.t.* a loss $\mathcal{L}$ for each weight as $\Delta_{\widetilde{H_a}} = \frac{\partial \mathcal{L}}{\partial \widetilde{H_a}}$, $\Delta_{\widetilde{H_b}} = \frac{\partial \mathcal{L}}{\partial \widetilde{H_b}}$, respectively. Then,

$$\Delta_{\widetilde{H_a}} = \frac{\partial \mathcal{L}}{\partial \widetilde{H_a}^{1}} + \frac{\partial \mathcal{L}}{\partial \widetilde{H_a}^{j_1}} + \cdots + \frac{\partial \mathcal{L}}{\partial \widetilde{H_a}^{j_{N-1}}}, \tag{S-12}$$

$$\Delta_{\widetilde{H_b}} = \frac{\partial \mathcal{L}}{\partial \widetilde{H_b}^{1}} + \frac{\partial \mathcal{L}}{\partial \widetilde{H_b}^{j_1}} + \cdots + \frac{\partial \mathcal{L}}{\partial \widetilde{H_b}^{j_{N-1}}}, \tag{S-13}$$

where each term is then computed by applying the chain rule. According to the standard hyper-complex backward propagation rule [17], each weight is updated as follows:

$$\widetilde{H_a} = \widetilde{H_a} - \lambda \Delta_{\widetilde{H_a}}, \quad \widetilde{H_b} = \widetilde{H_b} - \lambda \Delta_{\widetilde{H_b}}, \tag{S-14}$$

where the parameter λ is correlated with the learning rate.

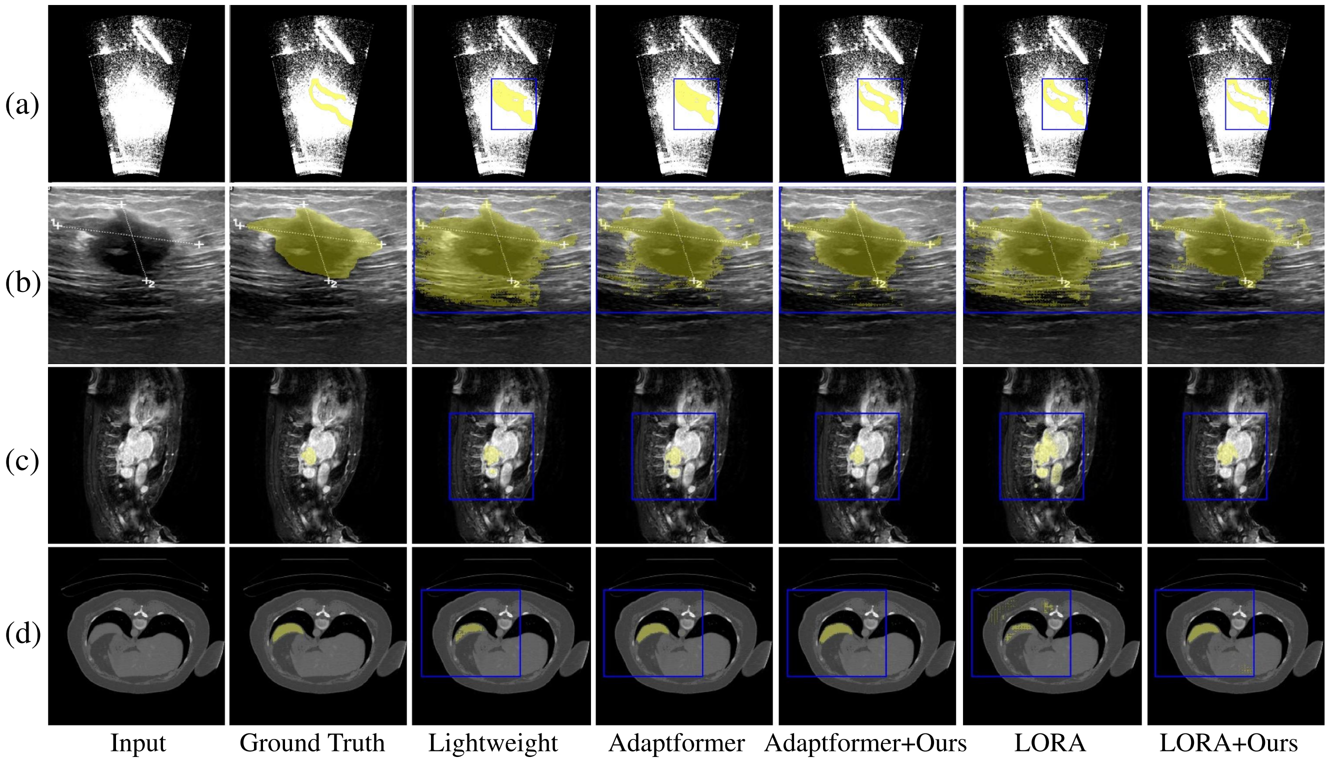
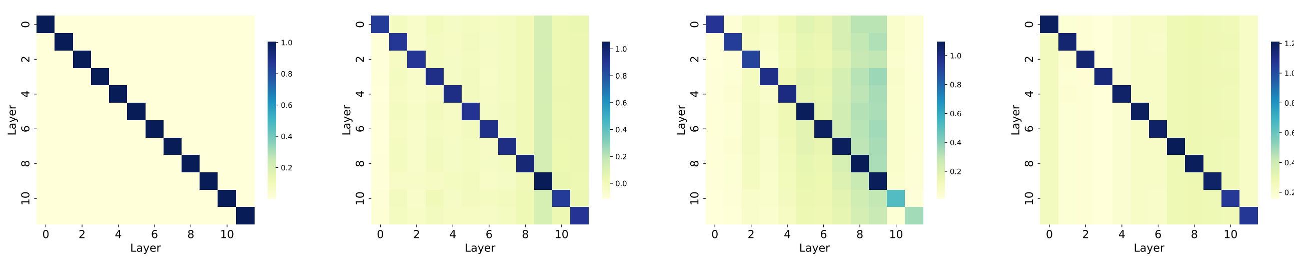


Figure S-1. **Qualitative segmentation results on four datasets**, i.e., (a) remote sensing image segmentation on SONAR dataset [19], (b) medical image segmentation on BRAST [1] dataset, (c) medical image segmentation on MOMO [2] dataset, and (d) medical image segmentation on SPLEN [2] dataset. "Lightweight": freezes all the backbone parameters and only tunes SAM's lightweight mask decoder.



(a) Initialization (b) TRCAN [8] (c) NWPU [4–6] (d) ADOME [16]

Figure S-2. **Visualization of Relation Matrix (RM) on three datasets.** Initialization: we initialize the RM as a diagonal matrix.

C. Extension Experiments

As shown in Table. S-1, our method is easily plugged into various PEFT methods and achieves higher accuracy with very few extra parameters.

Visualizations. we present more visual comparisons of our representative segmentation examples with those from two baseline models, i.e., LORA [9] and Adaptformer [3], as shown in Fig. S-1. These results further underscore the enhanced precision in segmentation achieved by our methods. Fig. S-2 shows the different distribution *w.r.t.* various scenarios. The above result suggests that our relation matrix captures valuable cross-block information.

D. Datasets and Hyper-parameters

Datasets. To validate the effectiveness of our SAM-COBOT, we conduct experiments on fine-tuning SAM [11] to 10 datasets.

(1) COCO2017 (COCO) [13]. The COCO dataset comprises 118,287 natural images in the training set and 5,000 natural images in the validation set for natural image segmentation. We fine-tune SAM on the training set and evaluate its efficacy on the validation set.

(2) TRASHCAN (TRCAN) [8]. The TRCAN dataset consists of 6008 underwater trash images in the training set and 1204 underwater trash images in the validation set for natu-

Method	Param(M)	ADOME	NWPU	TRCAN
LST	7.91	86.5 ± 0.2	80.9 ± 0.1	70.7 ± 0.2
VPT	0.10	87.7 ± 0.2	81.8 ± 0.2	71.5 ± 0.1
Attention-tuning	28.44	90.8 ± 0.1	84.9 ± 0.1	74.0 ± 0.1
AT+Ours	28.47	**91.0** ± 0.1	**85.2** ± 0.2	**74.3** ± 0.1
SSF	0.27	88.5 ± 0.3	81.9 ± 0.1	73.0 ± 0.2
SSF+Ours	0.34	**90.6** ± 0.5	**82.8** ± 0.2	**73.5** ± 0.1
BitFit	0.10	86.3 ± 0.1	80.6 ± 0.1	72.1 ± 0.1
BitFit+Ours	0.17	**89.7** ± 0.5	**82.0** ± 0.1	**73.1** ± 0.1

Table S-1. **Additional comparisons with various PEFT methods, e.g., LST [20], VPT [10], Attention-tuning [21], SSF [12], BitFit [23], on three datasets.**

ral image segmentation. We fine-tune SAM on the training set and evaluate its efficacy on the validation set.

(3) NWPU VHR-10 (NWPU) [4–6]. The NWPU dataset comprises 650 images for remote sensing image segmentation. As recommended in [6], we allocate 70% of the images for fine-tuning and the remaining 30% for evaluation.

(4) SAR Ship Detection Dataset (SSDD) [24]. The SSDD dataset comprises 812 SAR Ship images in the training set and 348 SAR Ship images in the validation set for remote sensing image segmentation. We fine-tune SAM on the training set and evaluate its efficacy on the validation set.

(5) Marine Debris dataset (SONAR) [22]. The SONAR dataset comprises 1000 marine debris images for training, 251 marine debris images for validating and 617 marine debris images for testing. We fine-tune SAM on the training set and evaluate its efficacy on the testing set.

(6) CT Abdominal organ (ADOME) [16]. The ADOME dataset comprises 50 labeled 3D CT images for medical image segmentation. Following [15], we split 80% of the image slices for fine-tuning and 20% for testing.

(7) Spleen (SPLEN) [2] & **(8) Cardiac (MOMO)** [2]. The SPLEN dataset contains 61 3D CT volumes for spleen segmentation and the MOMO dataset contains 30 3D Monomodal MRI volumes for left atrium segmentation, they are both from the Medical Segmentation Decathlon challenge [2]. Following default setting [2], we split 80% of the image slices for fine-tuning and 20% for testing.

(9) Breast Ultrasound (BRAST) [1]. The breast ultrasound dataset contains 210 malignant breast ultrasound images. Following default setting [1], we split 80% of the image slices for fine-tuning and 20% for testing.

(10) Segrap (SEGRAP) [18] The SEGRAP dataset contains 120 3D CT scans for gross target volume of nasopharynx (GTVnx) segmentation. Following default setting [18], we split 80% of the image slices for fine-tuning and 20% for testing.

Hyper-parameters. We set $r = 4$ (i.e., rank) in LoRA [9] and $r = 16$ (i.e., dimension of hidden space) in Adaptformer [3], when employing these methods as our baseline models. For the suprasphere introduced intra-block enhancement module, we set $N = 4$. Regarding the additional parameters introduced in the two modules of SAM-COBOT, we follow VPT [10], and search for a superior hyper-parameter in terms of learning rate from a learning rate list: $\{1.25\times10^{-6}, 1.25\times10^{-5}, 1.25\times10^{-4}\}$ for medical image segmentation, and $\{10^{-4}, 10^{-3}, 10^{-2}\}$ for other scenarios.

References

[1] Walid Al-Dhabyani, Mohammed Gomaa, Hussien Khaled, and Aly Fahmy. Dataset of breast ultrasound images. *Data in brief*, page 104863, 2020. 2, 3

[2] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. The medical segmentation decathlon. *Nature communications*, page 4128, 2022. 2, 3

[3] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 2, 3

[4] Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS journal of photogrammetry and remote sensing*, 117:11–28, 2016. 2, 3

[5] Gong Cheng, Junwei Han, Peicheng Zhou, and Lei Guo. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. *ISPRS Journal of Photogrammetry and Remote Sensing*, 98: 119–132, 2014.

[6] Gong Cheng, Peicheng Zhou, and Junwei Han. Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(12):7405–7415, 2016. 2, 3

[7] Thomas Hawkins. Hypercomplex numbers, lie groups, and the creation of group representation theory. *Archive for History of Exact Sciences*, 8:243–287, 1972. 1

[8] Jungseok Hong, Michael Fulton, and Junaed Sattar. Trashcan: A semantically-segmented dataset towards visual detection of marine debris. *arXiv preprint arXiv:2007.08097*, 2020. 2

[9] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 2, 3

[10] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727, 2022. 3

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the*

IEEE/CVF International Conference on Computer Vision (ICCV), 2023. 2

[12] Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. *Advances in Neural Information Processing Systems*, pages 109–123, 2022. 3

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 2

[14] Canyi Lu, Xi Peng, and Yunchao Wei. Low-rank tensor completion with a new tensor nuclear norm induced by invertible linear transforms. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5996–6004, 2019. 1

[15] Jun Ma and Bo Wang. Segment anything in medical images. *arXiv preprint arXiv:2304.12306*, 2023. 3

[16] Jun Ma, Yao Zhang, Song Gu, Cheng Zhu, Cheng Ge, Yichi Zhang, Xingle An, Congcong Wang, Qiyuan Wang, Xin Liu, Shucheng Cao, Qi Zhang, Shangqing Liu, Yunpeng Wang, Yuhui Li, Jian He, and Xiaoping Yang. Abdomenct-1k: Is abdominal organ segmentation a solved problem? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6695–6714, 2022. 2, 3

[17] Tohru Nitta. A quaternary version of the back-propagation algorithm. In *Proceedings of ICNN'95-International Conference on Neural Networks*, pages 2753–2756, 1995. 1

[18] SegRap2023 Challenge. Segmentation of organs-at-risk and gross tumor volume of npc for radiotherapy planning. https://segrap2023.grand-challenge.org/, 2023. 3

[19] Deepak Singh and Matias Valdenegro-Toro. The marine debris dataset for forward-looking sonar semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3741–3749, 2021. 2

[20] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Lst: Ladder side-tuning for parameter and memory efficient transfer learning. *Advances in Neural Information Processing Systems*, 35:12991–13005, 2022. 3

[21] Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Jakob Verbeek, and Hervé Jégou. Three things everyone should know about vision transformers. In *European Conference on Computer Vision*, pages 497–515. Springer, 2022. 3

[22] Lin Wang, Xiufen Ye, Liqiang Zhu, Weijie Wu, Jianguo Zhang, Huiming Xing, and Chao Hu. When sam meets sonar images. *arXiv preprint arXiv:2306.14109*, 2023. 3

[23] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021. 3

[24] Tianwen Zhang, Xiaoling Zhang, Jianwei Li, Xiaowo Xu, Baoyou Wang, Xu Zhan, Yanqin Xu, Xiao Ke, Tianjiao Zeng, Hao Su, et al. Sar ship detection dataset (ssdd): Official release and comprehensive data analysis. *Remote Sensing*, 13(18):3690, 2021. 3

[25] Zemin Zhang, Gregory Ely, Shuchin Aeron, Ning Hao, and Misha Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3842–3849, 2014. 1