

# Rethinking Token Reduction with Parameter-Efficient Fine-Tuning in ViT for Pixel-Level Tasks

## Supplementary Material

### 6. Attention Distance Analysis

In this paper, we measure the average distance spanned by attention weights at different layers. This attention distance is analogous to receptive field size in CNNs [9]. Below, we outline the process for calculating the normalized mean attention distance. For simplicity, we omit the block index  $i$ . First, we compute the distance matrix based on  $\mathbf{P}$ . This involves calculating the distances of each token relative to all other tokens, resulting in a distance matrix  $\mathbf{D} \in \mathbb{R}^{p \times p}$ , where  $p$  denotes the length of the patch token. Following the principles of self-attention [9, 55], we first compute the query embedding  $\mathbf{Q}$  and key embedding  $\mathbf{K}$  of the patch tokens  $\mathbf{P}$ . Next, we obtain the attention weights, denoted as  $\mathbf{A}$  using the formula  $\mathbf{A} = \mathbf{Q}\mathbf{K}^\top$ . We represent the attention weights for each head as  $\mathbf{A}_j$ . Subsequently, we derive the weighted distance matrix  $\mathbf{W}_j$  by calculating  $\mathbf{W}_j = \mathbf{D} \odot \mathbf{A}_j$ . Finally, for the mean attention distance of  $j^{th}$  head, we can use the following formula:

$$d_j = \frac{1}{p} \sum_{m=1}^p \sum_{n=1}^p (\mathbf{W}_j)_{m,n} \quad (15)$$

To compute the normalized mean attention distance, we apply min-max normalization to  $\mathbf{d} = [d_j]_{j=1}^N$ , where  $N$  denotes the number of heads in the attention.

### 7. Mutual Information Analysis

Normalized Mutual Information is used to measure the attention collapse [47]. Let  $p_Q(q)$  and  $p_K(k)$  be the spatial distribution of query embeddings  $\mathbf{Q}$  and key embeddings  $\mathbf{K}$  and assume that these query tokens are uniformly distributed since a single query token is given for each spatial coordinate. That is  $p_Q(q) = \frac{1}{N}$ . Our goal is to measure the mutual information of the  $p_Q(q)$  and  $p_K(k)$ .

$$I(q; k) = \sum p_{QK}(q, k) \log \frac{p_{QK}(q, k)}{p_Q(q)p_K(k)}, \quad (16)$$

where  $p_{QK}(q, k) = \pi(k|q)p_Q(q)$  represents the joint distribution of  $p_Q(q)$  and  $p_K(k)$  and  $\pi(k|q)$  denotes the conditional distribution, which is the attention weights after softmax normalization. Since  $p_Q(q)$  is constant,  $p_K(k) = p_{QK}(q, k) = \pi(k|q)p_Q(q)$ . Then, we get normalized mutual information  $I_{norm}$  by

$$I_{norm} = \frac{I(q; k)}{\sqrt{H(q)H(k)}}, \quad (17)$$

where  $H(q)$  and  $H(k)$  are the entropy of  $p_Q(q)$  and  $p_K(k)$ , respectively.

### 8. Fourier Analysis

Following [46, 47], let  $\mathbf{P}$  be patch token sequence. We begin by applying the fast Fourier transform (FFT) to  $\mathbf{P}$ , followed by a conversion to obtain the log amplitude:

$$\delta = \log |\text{FFT}(\mathbf{P})| \quad (18)$$

Subsequently, we extract the half-diagonal components, denoted as  $\delta'$ . The relative log amplitudes are then computed as follows:

$$\Delta = \delta' - \delta'_{max}, \quad (19)$$

where  $\delta'_{max}$  represents the maximum amplitude, identified as the first element of  $\delta'$ .

### 9. Decoder

The application of PEFT to complex downstream tasks necessitates the incorporation of a decoder that introduces non-linearity, in contrast to a simple linear layer in [44]. In this work, we aim to demonstrate the capacity of PEFT methods to effectively transfer knowledge and the inductive bias introduced by our approach. To mitigate the effects of inductive bias associated with intricately designed decoders, such as those based on convolutional architectures like UperNet [60], we implement a MLP-based decoder. The structure of our decoder can be represented as follows:

$$\text{Decoder}(\cdot) = \text{MLP}(\text{Up}(\text{MLP}(\cdot))), \quad (20)$$

where Up denotes an upsampling operation using bilinear interpolation.

### 10. Loss Functions

The total loss function in our approach is a composite of three components: the segmentation loss  $\mathcal{L}_{seg}$ , the distillation loss  $\mathcal{L}_{cos}$ , and a regularization term  $\mathcal{L}_{rate}$ , which controls the mask rate  $k$ . The overall loss is defined as:

$$\mathcal{L} = \mathcal{L}_{seg} + \lambda_1 \mathcal{L}_{rate} + \lambda_2 \mathcal{L}_{cos} \quad (21)$$

where  $\lambda_1$  and  $\lambda_2$  are hyperparameters that balance the contributions of the  $\mathcal{L}_{rate}$  and  $\mathcal{L}_{cos}$ .

The segmentation loss  $\mathcal{L}_{seg}$  combines binary cross-entropy (BCE) and DICE to regularize the generation of segmentation masks.

$$\mathcal{L}_{seg} = \lambda_{bce} \mathcal{L}_{bce} + \lambda_{dice} \mathcal{L}_{dice}, \quad (22)$$

where  $\lambda_{bce}$ , and  $\lambda_{dice}$  are hyperparameters that balance the contributions of the BCE and DICE loss. The BCE loss  $\mathcal{L}_{bce}$  and DICE loss  $\mathcal{L}_{dice}$  are defined as:

$$\mathcal{L}_{bce} = -\frac{1}{H \times W} \sum_{m=1}^H \sum_{n=1}^W \left( \left( \mathbf{Y} \log \hat{\mathbf{Y}} \right)_{m,n} + \left( (\mathbf{1} - \mathbf{Y}) \log (\mathbf{1} - \hat{\mathbf{Y}}) \right)_{m,n} \right), \quad (23)$$

$$\mathcal{L}_{dice} = 1 - \frac{\epsilon + 2 \sum_{m=1}^H \sum_{n=1}^W \left( \mathbf{Y} \odot \hat{\mathbf{Y}} \right)_{m,n}}{\epsilon + \sum_{m=1}^H \sum_{n=1}^W \left( \mathbf{Y} + \hat{\mathbf{Y}} \right)_{m,n}}, \quad (24)$$

where  $\mathbf{Y}$  and  $\hat{\mathbf{Y}} \in \mathbb{R}^{H \times W}$  represent the ground truth and predicted segmentation maps, respectively, with values in the range  $[0, 1]$ . The term  $\epsilon$  is a small constant added to prevent division by zero.

We directly adopt the Ada loss from [65] to control the mask rate.

$$\mathcal{L}_{rate} = \left( \frac{1}{N \times h \times w} \sum_{i=1}^N \sum_{m=1}^h \sum_{n=1}^w (\mathbf{M}_P^i)_{m,n} - k \right)^2 \quad (25)$$

where  $\mathbf{M}_P^i \in \mathbb{R}^{H' \times W'}$  denotes the mask at  $i^{\text{th}}$  transformer block for patch tokens,  $N$  represent the total number of transformer blocks, and  $k$  is the target mask rate.

For distillation, we use cosine similarity to align the representations of the fine-tuned and pre-trained models. The logits from both models are processed through an MLP-based distillation head, yielding  $\dot{\mathbf{X}}^N$  and  $\ddot{\mathbf{X}}^N$ , where  $\dot{\mathbf{X}}^N, \ddot{\mathbf{X}}^N \in \mathbb{R}^{(l+p) \times d}$ . The distillation loss is defined as:

$$\mathcal{L}_{cos} = 1 - \sum_{i=1}^{l+p} \frac{\dot{\mathbf{X}}_i^N \cdot \ddot{\mathbf{X}}_i^N}{\|\dot{\mathbf{X}}_i^N\| \cdot \|\ddot{\mathbf{X}}_i^N\|} \quad (26)$$

## 11. Implementation Details

**Baselines.** We compare our method with several approaches, including linear probing, fine-tuning of the decoder only, and VPT [26], which inserts learnable tokens into the hidden states of each transformer block. Additionally, we consider AdaptFormer [4], which adds trainable low-rank MLP layers in parallel to the FFN layer within a transformer block, and LoRA [23], which incorporates trainable low-rank linear layers alongside the frozen linear weights. Finally, we include EVP [35], which integrates high-frequency priors with parallel adapters in the transformer blocks.

Table 4 presents the implementation details for training binary segmentation tasks.

**Salient object segmentation.** We utilize four widely recognized datasets for salient object segmentation: DUTS [57],

ECSSD [62], SOD [43], and HKU-IS [31]. The DUTS dataset comprises 10,553 training images and 5,019 testing images. The ECSSD dataset includes 1,000 testing images, while the SOD dataset consists of 300 testing images. Additionally, the HKU-IS dataset contains 4,447 testing images. All methods are trained on the training set of DUTS [57] and evaluated on the testing sets of DUTS, ECSSD, SOD, and HKU-IS.

**Defocus blur detection.** Following the methodology presented in [50], we conduct training on the CUHK dataset [50], which consists of a total of 704 samples exhibiting partial defocus. The network is trained using 604 images from the CUHK dataset, with testing performed on the remaining images.

**Camouflaged object segmentation.** To assess our methods, we select four commonly utilized datasets for camouflaged object segmentation. The COD10K dataset [14] comprises 3,040 training samples and 2,026 testing samples. The CAMO dataset [28] provides 1,000 images for training and 250 for testing. The NC4K dataset [41] contains 4,121 testing samples, while the CHAMELEON dataset [52] includes 76 testing images. In alignment with [14], we train our methods on the training sets of COD10K and CAMO, and we evaluate their performance on the testing sets of COD10K, CAMO, NC4K, and CHAMELEON.

**Polyp segmentation.** For polyp segmentation, we employ three datasets: Kvasir [25], ETIS [51], and CVC-ColonDB [54]. The Kvasir dataset includes 1,100 training images and 196 testing images. The ETIS dataset contains 196 images, while CVC-ColonDB comprises 612 images. Our training is conducted on the training set of Kvasir, with evaluation performed using the testing sets of ETIS, CVC-ColonDB, and Kvasir.

**Skin lesion segmentation.** For skin lesion segmentation, we focus on the ISIC 2017 dataset [6], which provides 2,000 training images and 600 testing images.

**Semantic segmentation.** We utilize two foundational datasets for semantic segmentation: ADE20K [66] and Cityscapes [7]. Following [44], we conduct training and testing on ADE20K and Cityscapes, respectively. As presented in Table 5, we train for a total of 8 epochs on ADE20K and 48 epochs on Cityscapes, thereby ensuring that the number of iterations remains approximately consistent across both datasets.

## 12. More Experiment Results

### 12.1. Ablation Study

**TR placement.** Attention mechanisms incur notable computational overhead for long sequences. Our ablation study in Tab. 6 evaluates the effects of TR placement, demonstrating that pre-attention TR application significantly degrades pixel-level task performance. For instance,  $F_\beta^w$  scores on

Table 4. Experimental settings for binary segmentation datasets. We train all methods using the same hyperparameters.

| Configuration          | DUTS | CUHK | COD10K+CAMO       | Kvasir | ISIC 2017 |
|------------------------|------|------|-------------------|--------|-----------|
| Optimizer              |      |      | AdamW [40]        |        |           |
| Base learning rate     |      |      | 1.5e-4            |        |           |
| Weight decay           |      |      | 1e-4              |        |           |
| Batch size             |      |      | 10                |        |           |
| Learning rate schedule |      |      | Cosine decay [39] |        |           |
| Warmup epochs          | 4    | 10   | 10                | 10     | 5         |
| Training epochs        | 16   | 40   | 50                | 40     | 20        |

Table 5. Experimental settings for semantic segmentation datasets. We train all methods using the same hyperparameters.

| Configuration          | ADE20K            | Cityscapes |
|------------------------|-------------------|------------|
| Optimizer              | AdamW [40]        |            |
| Base learning rate     | 1.5e-4            |            |
| Weight decay           | 1e-4              |            |
| Batch size             | 4                 |            |
| Learning rate schedule | Cosine decay [39] |            |
| Warmup epochs          | 1                 | 6          |
| Training epochs        | 8                 | 48         |

Table 6. Ablation study on TR placement and different backbones. The rows highlighted in gray represent our method (default setting). **Bold** values denote the optimal performance under TR.

| Methods      |                   | DUTS                   |                       |                     | Kvasir                 |                       |                     |
|--------------|-------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|
|              |                   | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ |
| TR Placement | No TR             | .906                   | .937                  | .946                | .935                   | .957                  | .967                |
|              | Attention         | .878                   | .926                  | .937                | .865                   | .921                  | .933                |
|              | FFN               | <b>.895</b>            | <b>.933</b>           | <b>.944</b>         | <b>.938</b>            | <b>.960</b>           | <b>.975</b>         |
| Backbones    | iBOT-Base [67]    | .826                   | .893                  | .905                | .925                   | .957                  | .968                |
|              | iBOT-Large [67]   | .847                   | .904                  | .917                | .941                   | .964                  | .973                |
|              | EVA02-Base [16]   | .883                   | .923                  | .934                | .953                   | .967                  | .979                |
|              | EVA02-Large [16]  | .901                   | .934                  | .943                | <b>.965</b>            | <b>.974</b>           | <b>.984</b>         |
|              | DINOv2-Base [44]  | .895                   | .933                  | .944                | .938                   | .960                  | .975                |
|              | DINOv2-Large [44] | <b>.908</b>            | <b>.940</b>           | <b>.949</b>         | .958                   | .969                  | .980                |

DUTS decline to 0.878 (vs. 0.895 when TR is applied to FFN), with analogous performance reductions observed on the Kvasir dataset. This aligns with [65]’s findings in image classification. Future work should explore effective TR application in attention for pixel-level tasks without compromising performance.

**Different backbones.** To assess the generalization and scalability of our method, we conduct experiments on backbones with diverse attention properties (e.g. DINOv2 [44], EVA02 [16], iBOT [67]). As shown in Tab. 6, our method achieves consistent performance across these backbones. Furthermore, when scaling to larger variants, performance improves with model size, empirically demonstrating scalability.

## 12.2. Segmentation Results

**Comparison with recent TR methods.** To evaluate our method against existing TR approaches for pixel-level tasks, we conduct comparisons in Tab. 7. We evaluate recent TR methods specialized for distinct pixel-level tasks, including DTMFormer [58] in medical image segmentation, DoViT [37] in semantic segmentation, and SViT [36] in instance or semantic segmentation. Experiments are performed on established benchmarks, including Kvasir and ISIC 2017 for medical image segmentation, and ADE20K and Cityscapes for semantic segmentation, ensuring consistency with prior experimental setups.

**Semantic Segmentation.** In Table 8, our methods demonstrate superior performance relative to DyT on both ADE20K and Cityscapes. Notably, our approach outperforms AdaptFormer, a method that does not incorporate TR. This advantage may stem from our method’s ability to maintain the diversity of attention while the high-frequency components aggregated by the token compensator positively influence multi-class segmentation. This effect is particularly significant on ADE20K, which includes 151 classes, in contrast to Cityscapes, which has only 34 classes. Additionally, our methods achieve performance comparable to that of PEFT methods without TR on the Cityscapes dataset.

**Salient object segmentation.** Table 9 presents the results of our evaluation on salient object segmentation (SOS). Our method demonstrates superior performance compared to DyT across all SOS datasets, while achieving performance levels comparable to methods that do not utilize TR. This suggests that our approach effectively enhances fundamental segmentation capabilities.

**Camouflaged object segmentation.** In Table 10, we demonstrate the performance of our method on four camouflaged object segmentation (COS) datasets. Our approach achieves superior performance relative to DyT across all datasets, showcasing its effectiveness in handling more complex scenes. Furthermore, it attains performance levels comparable to those of methods that do not incorporate TR. This finding suggests that our method significantly enhances segmentation capabilities in challenging scenarios

Table 7. Comparison with full fine-tuning (full FT) TR methods.

| Methods       |                | Total<br>Params.<br>(M) | Trainable<br>Params. (M) /<br>Ratio (%) | Semantic Segmentation |                | Medical                |                       |                     |                        |                       |                     |
|---------------|----------------|-------------------------|---|-----------------------|----------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|
|               |                |                         |   | ADE20K [66]           | Cityscapes [7] | Kvasir [25]            |                       | ISIC 2017 [6]       |                        |                       |                     |
|               |                |                         |   | mIOU (%)              | mIOU (%)       | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ |
| Full FT w/ TR | DTMFormer [58] | 33.6                    | 33.6 / 100.0%                           | -                     | -              | .870                   | .926                  | .941                | .776                   | .830                  | .821                |
|               | DoViT [37]     | 88.5                    | 88.5 / 100.0%                           | 46.5                  | <b>76.4</b>    | -                      | -                     | -                   | -                      | -                     | -                   |
|               | SViT [36]      | 48.0                    | 48.0 / 100.0%                           | 44.8                  | 76.2           | -                      | -                     | -                   | -                      | -                     | -                   |
| PEFT w/ TR    | DyT [65]       | 86.7                    | 1.2 / 1.4%                              | 49.5                  | 50.6           | .897                   | .944                  | .963                | .843                   | <b>.868</b>           | <b>.884</b>         |
|               | Ours           | 87.9                    | 2.6 / 2.9%                              | <b>52.6</b>           | 67.3           | <b>.938</b>            | <b>.960</b>           | <b>.975</b>         | .871                   | .862                  | .881                |

Table 8. Quantitative and efficiency comparison of semantic segmentation using parameter-efficient fine-tuning (PEFT), both without and with token reduction (TR).

| Methods |                 | FLOPs (G) | mIOU        |                |             |
|---------|-----------------|-----------|-------------|----------------|-------------|
|         |                 |           | ADE20K [66] | Cityscapes [7] | Avg.        |
| w/o TR  | Linear          | 117.08    | 45.9        | 60.9           | 54.1        |
|         | VPT [26]        | 122.52    | 49.2        | 61.8           | 55.5        |
|         | LoRA [23]       | 118.90    | 50.7        | 61.6           | 56.2        |
|         | AdaptFormer [4] | 118.70    | <b>51.5</b> | <b>68.8</b>    | <b>60.2</b> |
| w/ TR   | DyT [65]        | 97.87     | 49.5        | 50.6           | 50.0        |
|         | Ours            | 107.55    | <b>52.6</b> | <b>67.3</b>    | <b>60.0</b> |

associated with camouflaged objects.

**Polyp segmentation.** In Table 11, we present the performance of our method on several medical segmentation datasets. Our approach demonstrates superior performance compared to DyT across all datasets, highlighting its effectiveness in addressing the complexities inherent in medical images. Notably, our method outperforms all PEFT methods without TR including AdaptFormer on Kvasir and CVC-ColonDB. This finding indicates the robustness and adaptability of our methods in medical segmentation tasks.

### 12.3. VTAB-1K Results

To evaluate the performance of our method on classification tasks and its adaptation capabilities when training data is limited, we employ the VTAB-1K [64] benchmark. Following the approach outlined in [65], we utilize a Vision Transformer (ViT) pretrained on ImageNet-21k, maintaining the same training scheme. In contrast to our segmentation configuration, we adopt a significantly reduced rank setting for this classification task, specifically configuring the adapter rank at 6 and the token compensator rank at 2.

Results are summarized in Tab. 12. For TR-PEFT methods on natural images, our approach achieves one first-rank and one second-rank result, outperforming DyT, which ranks second in both instances. In specialized domains, AdaptFormer demonstrates superior performance across most datasets, while our method and DyT achieve comparable performance. Conversely, in structured domains, our method exhibits limitations due to limited dataset scale and significant domain gaps between structured images and pre-training data, necessitating larger training datasets than DyT

to achieve comparable performance.

## 13. More Visualization

### 13.1. Masked Tokens

We provide additional visualizations to illustrate the effects of our token reduction methods across various scenes, including simple, complex, and medical contexts. Fig. 6 displays the effects of our token reduction in a simple scene, exemplified by the DUTS dataset. Fig. 7 demonstrates the effects in a complex scene. Fig. 8 and Fig. 9 showcase the impact of our token reduction methods in polyp segmentation and skin lesion segmentation, respectively.

Our method progressively masks tokens along object boundaries as network layers increase. In Fig. 6, we observed that our approach tends to mask inner object regions while preserving boundary integrity. In other words, our method retains only the boundary information of objects, thereby achieving a relatively higher mask ratio within a single image. In Fig. 7, Fig. 8, and Fig. 9, the masking strategy effectively identifies concealed objects that require fine-grained processing, demonstrating the effectiveness of our approach. This strategy thereby enhances the model’s confidence in target identification across various segmentation tasks.

### 13.2. Segmentation Maps

To further demonstrate the effectiveness of our method, we present additional visualizations of segmentation maps across diverse scenarios, including simple, complex, and medical contexts. Specifically, Fig. 10 illustrates segmentation results in a simple scene using the DUTS dataset as a representative example. Fig. 11 highlights segmentation performance in complex scenes, while Fig. 12 showcases results in medical applications, specifically polyp segmentation and skin lesion segmentation.

As depicted in Fig. 10, our method achieves superior performance in simple scenes. The segmented regions generated by our approach exhibit smoother boundaries and finer details, such as continuous thin lines, compared to competing methods. Additionally, the segmentation maps produced by our method contain minimal noise, which can be



Table 9. Quantitative comparison of salient object segmentation.

| Methods |                 | DUTS [57]              |                       |                     | ECSSD [62]             |                       |                     | SOD [43]               |                       |                     | HKU-IS [31]            |                       |                     |
|---------|-----------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|
|         |                 | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ |
| w/o TR  | Linear          | .658                   | .841                  | .832                | .777                   | .892                  | .875                | .715                   | .810                  | .785                | .741                   | .881                  | .879                |
|         | Decoder Only    | .756                   | .879                  | .889                | .849                   | .916                  | .921                | .787                   | .834                  | .837                | .817                   | .906                  | .924                |
|         | VPT [26]        | .896                   | .931                  | .938                | .934                   | .951                  | .958                | .830                   | .826                  | .824                | .914                   | .940                  | .954                |
|         | AdaptFormer [4] | <b>.906</b>            | <b>.937</b>           | <b>.946</b>         | <b>.943</b>            | <b>.955</b>           | <b>.962</b>         | <b>.865</b>            | <b>.850</b>           | <b>.859</b>         | <b>.926</b>            | <b>.946</b>           | <b>.960</b>         |
|         | LoRA [23]       | .897                   | .932                  | .942                | .927                   | .949                  | .954                | .854                   | .851                  | .855                | .911                   | .940                  | .954                |
|         | EVP [35]        | .887                   | .929                  | .935                | .930                   | .949                  | .955                | .846                   | .836                  | .832                | .912                   | .940                  | .953                |
| w/ TR   | DyT [65]        | .859                   | .921                  | .930                | .902                   | .938                  | .944                | .837                   | .849                  | .855                | .886                   | .931                  | .945                |
|         | Ours            | .895                   | .933                  | .944                | .924                   | <b>.948</b>           | <b>.958</b>         | <b>.860</b>            | <b>.851</b>           | <b>.863</b>         | .908                   | <b>.943</b>           | <b>.960</b>         |
|         | Ours†           | <b>.901</b>            | <b>.935</b>           | <b>.947</b>         | <b>.932</b>            | <b>.948</b>           | .957                | .856                   | .850                  | .857                | <b>.917</b>            | .942                  | .958                |

Table 10. Quantitative comparison of camouflaged object segmentation.

| Methods |                 | COD10K [14]            |                       |                     | CAMO [28]              |                       |                     | NC4K [41]              |                       |                     | CHAMELEON [52]         |                       |                     |
|---------|-----------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|
|         |                 | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ |
| w/o TR  | Linear          | .657                   | .860                  | .866                | .707                   | .853                  | .852                | .750                   | .889                  | .891                | .746                   | .892                  | .888                |
|         | Decoder Only    | .769                   | .888                  | .918                | .796                   | .886                  | .904                | .831                   | .907                  | .925                | .825                   | .908                  | .924                |
|         | VPT [26]        | .816                   | .903                  | .936                | .850                   | .908                  | .933                | .869                   | .918                  | .941                | .860                   | <b>.926</b>           | .951                |
|         | AdaptFormer [4] | <b>.826</b>            | <b>.906</b>           | <b>.941</b>         | <b>.866</b>            | <b>.911</b>           | <b>.939</b>         | <b>.879</b>            | <b>.922</b>           | <b>.946</b>         | <b>.873</b>            | <b>.926</b>           | <b>.952</b>         |
|         | LoRA [23]       | .814                   | .897                  | .935                | .852                   | .905                  | .934                | .867                   | .913                  | .939                | .855                   | .920                  | .944                |
|         | EVP [35]        | .803                   | .900                  | .933                | .832                   | .901                  | .921                | .859                   | .916                  | .937                | .847                   | .917                  | .938                |
| w/ TR   | DyT [65]        | .796                   | .894                  | .934                | .839                   | .901                  | .932                | .851                   | .910                  | .937                | .834                   | .909                  | .943                |
|         | Ours            | <b>.810</b>            | <b>.899</b>           | .935                | .842                   | .902                  | .932                | .853                   | .914                  | .939                | .834                   | .915                  | .936                |
|         | Ours†           | .808                   | <b>.899</b>           | <b>.936</b>         | <b>.848</b>            | <b>.905</b>           | <b>.933</b>         | <b>.859</b>            | <b>.915</b>           | <b>.939</b>         | <b>.860</b>            | <b>.924</b>           | <b>.952</b>         |

attributed to the integration of a learnable mask mechanism that effectively suppresses irrelevant regions.

In complex scenes (Fig. 11), our method maintains its advantage, particularly in preserving the continuity of thin structures—a challenge where alternative approaches often produce fragmented results. For medical applications (Fig. 12), our method demonstrates higher confidence levels in delineating abnormal regions. Notably, in cases involving ambiguous abnormal areas (e.g., the second row of Fig. 12), our approach maintains clearer distinctions compared to other methods, which exhibit uncertainty in such regions.

## 14. Limitations and Future Work

While the proposed method demonstrates improved efficiency in FFN computations through TR, two primary limitations remain. First, it does not directly address the computational overhead of attention mechanisms in long-sequence tasks. Our experiments in Tab. 6 show that applying token reduction before attention layers degrades performance in pixel-level tasks, as evidenced by  $F_{\beta}^w$  scores of 0.878

compared to 0.895 when token reduction is applied to FFN on DUTS. This aligns with prior work [65], which identified similar challenges in image classification, indicating that token reduction disrupts spatial dependencies essential for attention-based feature refinement. Second, the method demonstrates limited adaptability in structured image domains such as medical or satellite imagery, where significant domain gaps exist between pretraining data and target tasks. Achieving performance comparable to DyT in these domains requires substantially larger training datasets, underscoring a dependency on data scale that may limit practicality in resource-constrained settings.

Future research should focus on optimizing the integration of token reduction with attention mechanisms to minimize computational cost in pixel-level tasks. This could involve developing TR techniques that preserve attention dynamics without significantly compromising performance. Additionally, exploring the method’s adaptability to distant image domains is essential. Further evaluation should extend to multimodal understanding and generation, to assess the broader applicability and robustness of TR strategies. Addressing these challenges could advance the de-

Table 11. Quantitative comparison of polyp segmentation.

| Methods |                   | Kvasir [25]            |                       |                     | ETIS [62]              |                       |                     | CVC-ColonDB [54]       |                       |                     |
|---------|-------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|------------------------|-----------------------|---------------------|
|         |                   | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ | $F_{\beta}^w \uparrow$ | $S_{\alpha} \uparrow$ | $E_{\phi} \uparrow$ |
| w/o TR  | Linear            | .482                   | .782                  | .747                | .327                   | .726                  | .680                | .535                   | .816                  | .787                |
|         | Decoder Only      | .723                   | .870                  | .872                | .531                   | .774                  | .725                | .737                   | .863                  | .858                |
|         | VPT [26]          | .891                   | .939                  | .951                | .587                   | .814                  | .778                | .817                   | .900                  | .906                |
|         | AdaptFormer [4]   | .935                   | <b>.957</b>           | <b>.967</b>         | <b>.719</b>            | <b>.873</b>           | <b>.877</b>         | <b>.864</b>            | <b>.912</b>           | <b>.930</b>         |
|         | LoRA [23]         | <b>.936</b>            | .956                  | .966                | .641                   | .839                  | .839                | .850                   | .909                  | .929                |
|         | EVP [35]          | .857                   | .934                  | .943                | .618                   | .846                  | .831                | .806                   | .898                  | .907                |
| w/ TR   | DyT [65]          | .897                   | .944                  | .963                | .665                   | .859                  | .861                | .827                   | .900                  | .919                |
|         | Ours              | .938                   | <b>.960</b>           | <b>.975</b>         | .681                   | .864                  | <b>.871</b>         | .862                   | <b>.917</b>           | <b>.936</b>         |
|         | Ours <sup>†</sup> | <b>.940</b>            | .959                  | .974                | <b>.691</b>            | <b>.868</b>           | .862                | <b>.865</b>            | .912                  | .929                |

Table 12. Quantitative comparison of polyp segmentation.

|   | Natural     |             |             |             |             |             |             | Specialized |             |             |             | Structured  |             |             |             |             |             |             |             | Avg.        |
|---|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|   | CIFAR-100   | Caltech101  | DTD         | Flowers102  | Pets        | SVHN        | SUN397      | Camelyon    | EuroSAT     | Resisc45    | Retinopathy | Clevr-Count | Clevr-Dist  | DMLab       | KITTI-Dist  | dSpr-Loc    | dSpr-Ori    | sNORB-Azim  | sNORB-Elev  |             |
| <i>Parameter-efficient fine-tuning</i>                      |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |
| AdaptFormer   | 70.8        | 91.2        | <b>70.5</b> | <b>98.8</b> | <b>90.9</b> | <b>86.6</b> | <b>54.3</b> | 83.0        | <b>95.8</b> | <b>84.4</b> | <b>76.3</b> | 81.9        | 64.3        | 49.3        | <b>80.3</b> | 76.3        | 45.7        | 31.7        | 41.1        | 72.3        |
| LoRA  | 67.1        | 91.4        | 69.4        | <b>98.8</b> | 90.4        | 85.3        | 54.0        | 84.9        | 95.3        | <b>84.4</b> | 73.6        | <b>82.9</b> | <b>69.2</b> | 49.8        | 78.5        | 75.7        | 47.1        | 31.0        | <b>44.0</b> | 72.3        |
| VPT   | <b>78.8</b> | 90.8        | 65.8        | 98.0        | 88.3        | 78.1        | 49.6        | 81.8        | 96.1        | 83.4        | 68.4        | 68.5        | 60.0        | 46.5        | 72.8        | 73.6        | 47.9        | 32.9        | 37.8        | 69.4        |
| <i>Parameter-efficient fine-tuning with token reduction</i> |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |             |
| DyT   | 70.4        | 94.2        | 68.6        | 98.0        | 90.3        | 86.5        | 51.5        | <b>87.1</b> | 95.3        | 84.2        | 72.2        | 79.2        | 60.8        | 51.0        | 79.9        | 79.7        | <b>55.1</b> | <b>34.0</b> | 40.9        | <b>72.5</b> |
| Ours  | 68.3        | <b>94.4</b> | 68.8        | 98.6        | 89.9        | 84.7        | 52.2        | <b>87.1</b> | 95.7        | 83.8        | 72.6        | 81.1        | 60.7        | <b>51.1</b> | 79.9        | <b>83.0</b> | 50.9        | 30.0        | 43.3        | 72.4        |

velopment of efficient and accurate vision architectures for diverse real-world applications.

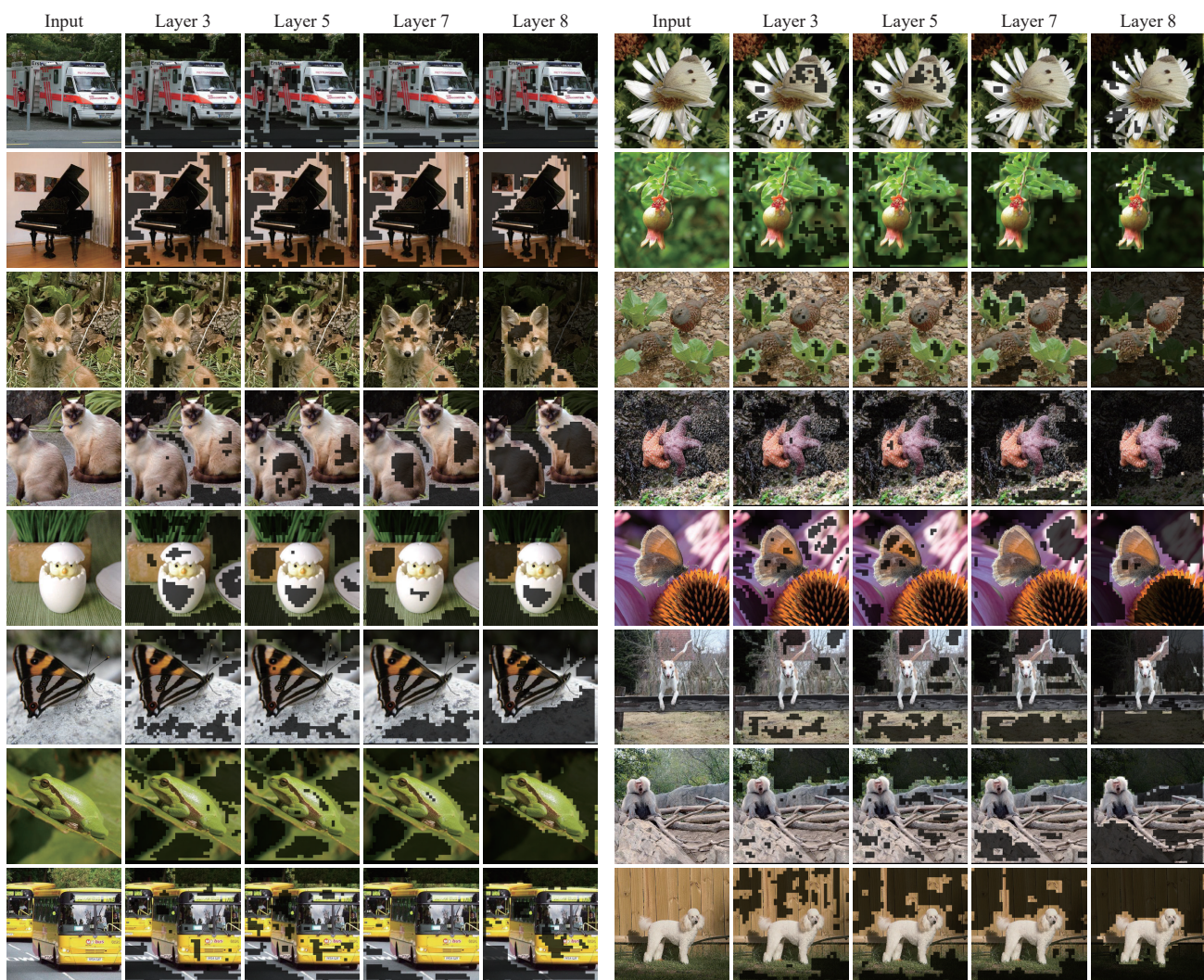


Figure 6. Visualization of the effects of our token reduction on DUTS dataset.



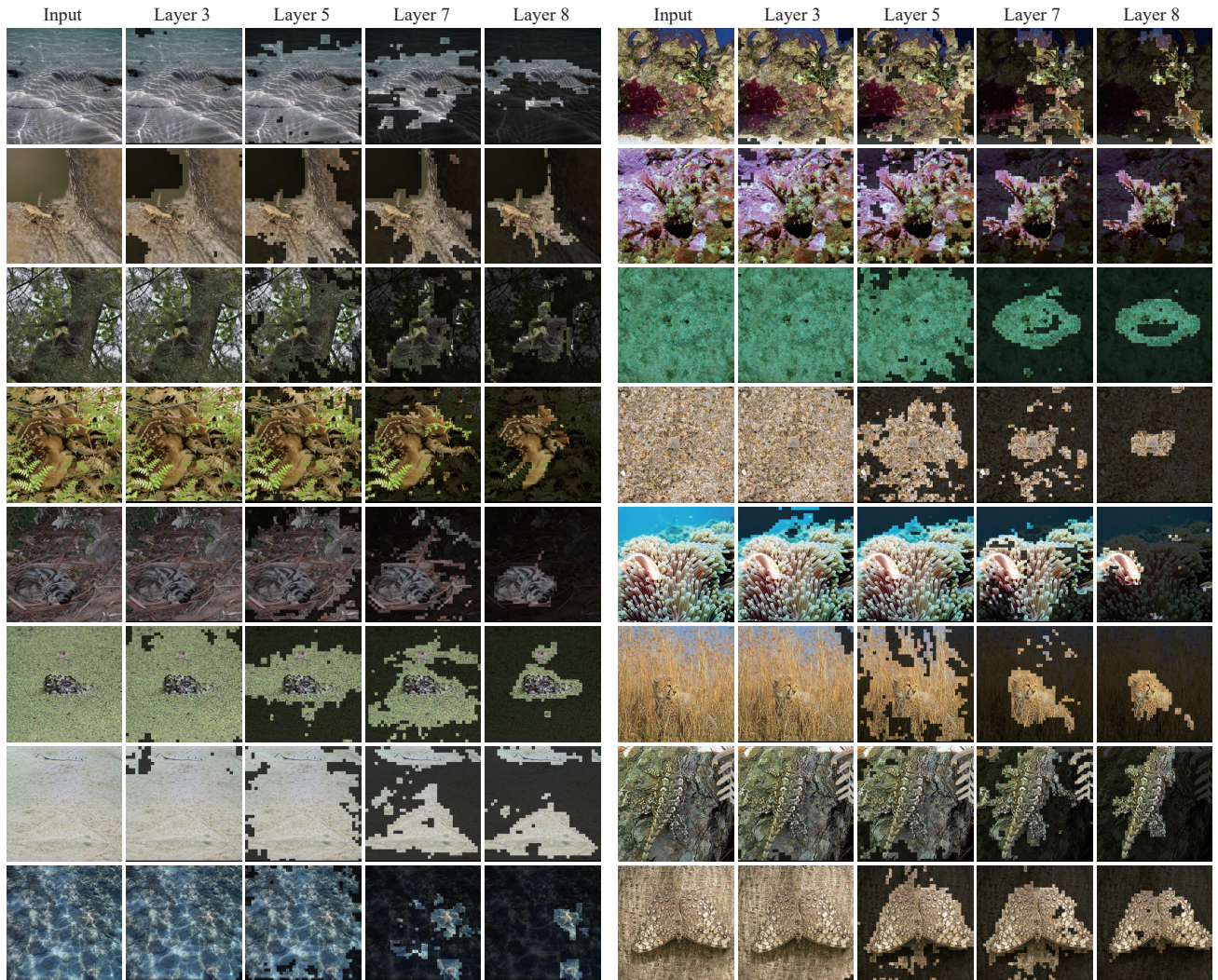


Figure 7. Visualization of the effects of our token reduction on COD10K and CAMO dataset.

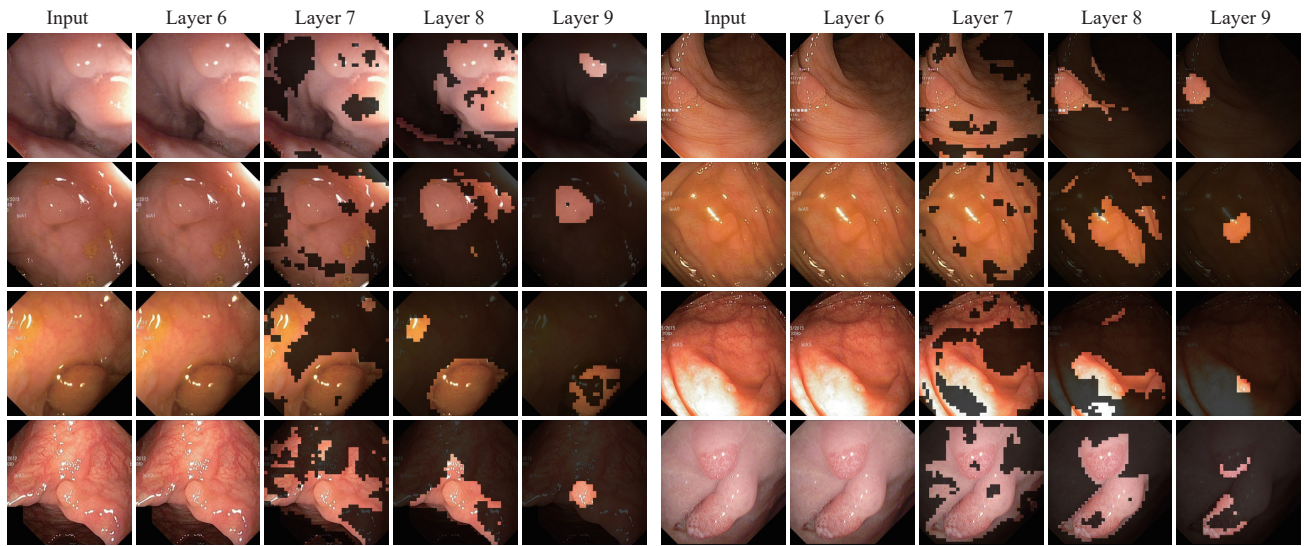


Figure 8. Visualization of the effects of our token reduction on Kvasir dataset.

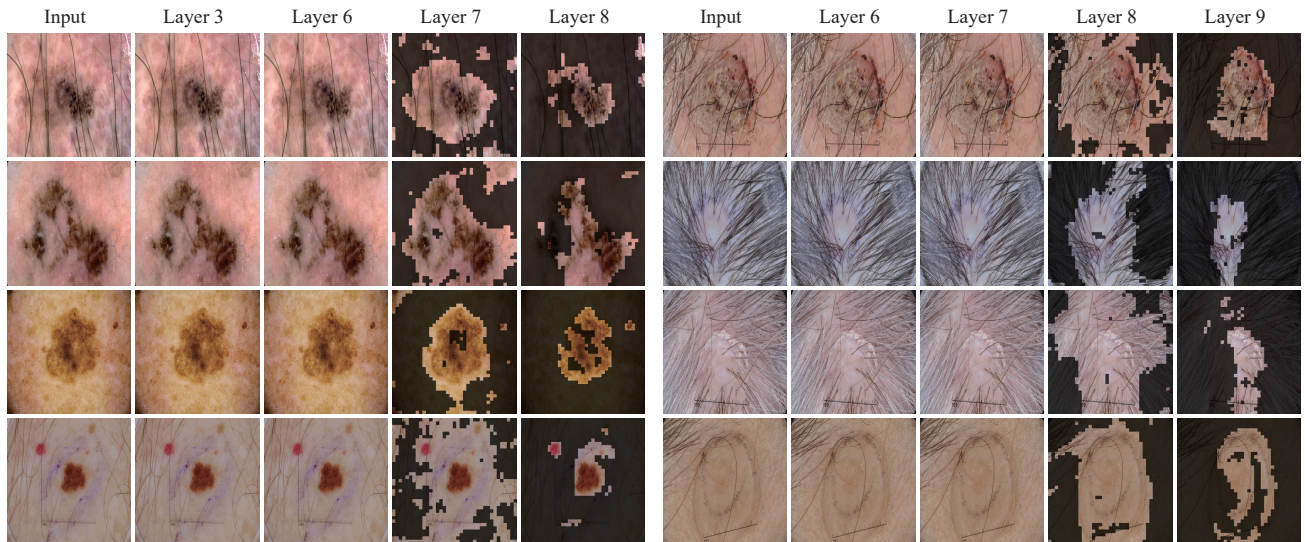


Figure 9. Visualization of the effects of our token reduction on ISIC 2017 dataset.





Figure 10. Visualization of the effects of our token reduction on ISIC 2017 dataset.

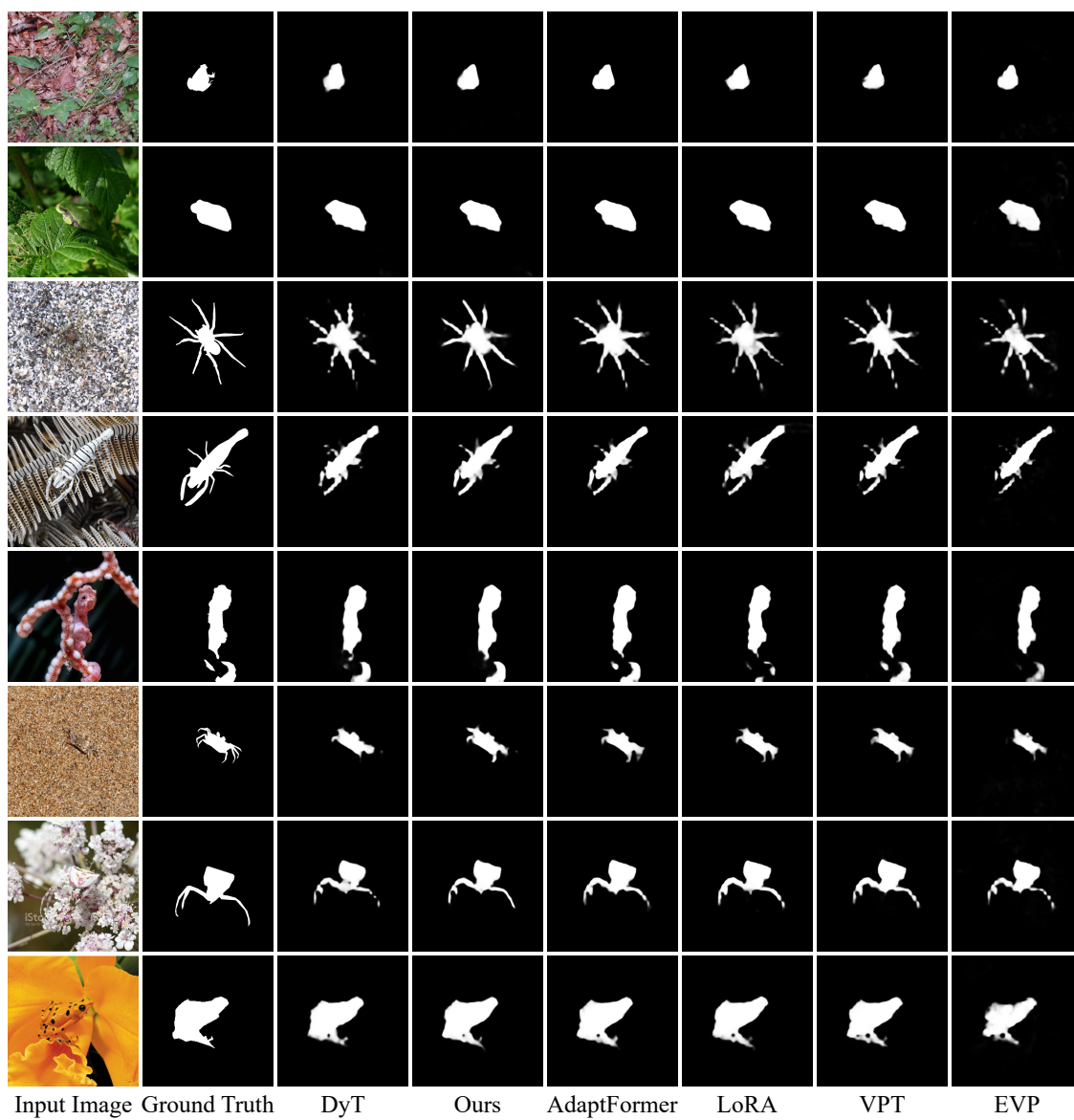


Figure 11. Visualization of the effects of our token reduction on ISIC 2017 dataset.

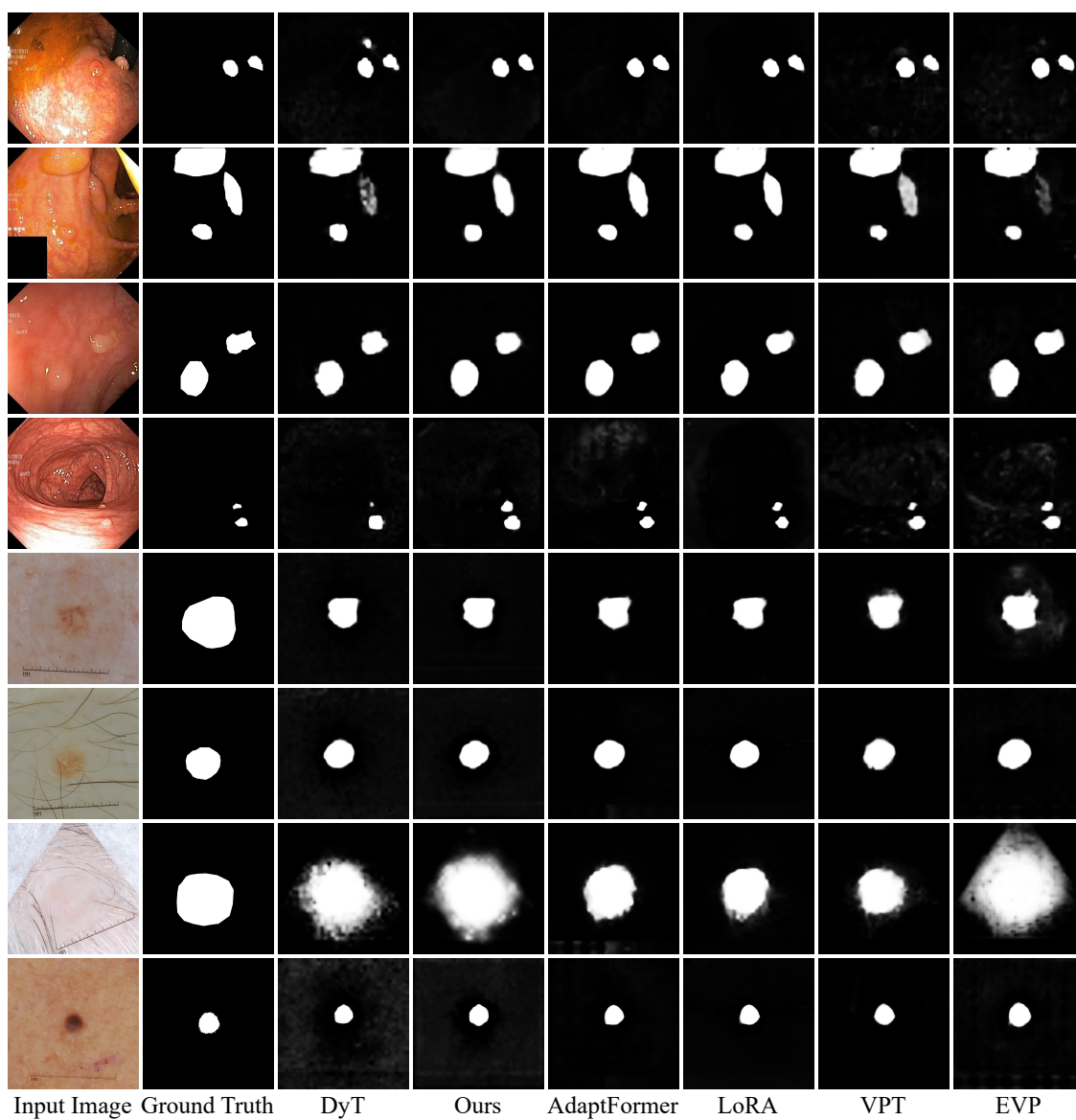


Figure 12. Visualization of the effects of our token reduction on ISIC 2017 dataset.

## References

- [1] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023. 1, 3
- [2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 1
- [3] Mengzhao Chen, Wenqi Shao, Peng Xu, Mingbao Lin, Kaipeng Zhang, Fei Chao, Rongrong Ji, Yu Qiao, and Ping Luo. Diffrate: Differentiable compression rate for efficient vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17164–17174, 2023. 1, 3
- [4] 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. 3, 4, 7, 2, 5, 6
- [5] Ming-Ming Cheng and Deng-Ping Fan. Structure-measure: A new way to evaluate foreground maps. *IJCV*, 129(9): 2622–2638, 2021. 6
- [6] Noel CF Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 168–172. IEEE, 2018. 3, 6, 7, 2, 4
- [7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 4
- [8] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations*, 2024. 4
- [9] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 4
- [11] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, 2017. 6
- [12] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *IJCAI*. AAAI Press, 2018. 6
- [13] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *CVPR*, 2020. 3, 6, 8
- [14] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE TPAMI*, 44(10): 6024–6042, 2022. 3, 6, 7, 2, 5
- [15] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. *arXiv preprint arXiv:2211.07636*, 2022. 1
- [16] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, page 105171, 2024. 1, 3
- [17] Shengxi Gui, Shuang Song, Rongjun Qin, and Yang Tang. Remote sensing object detection in the deep learning era—a review. *Remote Sensing*, 16(2):327, 2024. 3
- [18] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2022. 3
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4
- [20] Ruozhen He, Qihua Dong, Jiaying Lin, and Rynson W.H. Lau. Weakly-supervised camouflaged object detection with scribble annotations. *AAAI*, 37(1):781–789, 2023. 8
- [21] Charles Herrmann, Richard Strong Bowen, and Ramin Zabih. Channel selection using gumbel softmax. In *European conference on computer vision*, pages 241–257. Springer, 2020. 5
- [22] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. 1, 3, 4
- [23] 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. 1, 3, 7, 2, 4, 5, 6
- [24] Jian Hu, Jiayi Lin, Shaogang Gong, and Weitong Cai. Relax image-specific prompt requirement in sam: A single generic prompt for segmenting camouflaged objects. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12511–12518, 2024. 8
- [25] Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas De Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *MultiMedia modeling: 26th international conference, MMM 2020, Daejeon, South Korea, January 5–8, 2020, proceedings, part II* 26, pages 451–462. Springer, 2020. 3, 6, 7, 2, 4

- [26] 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 (ECCV)*, 2022. 1, 3, 7, 2, 4, 5, 6
- [27] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 1, 3
- [28] Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabran network for camouflaged object segmentation. *CVIU*, 184:45–56, 2019. 6, 7, 2, 5
- [29] Cheng Lei, Jie Fan, Xinran Li, Tianzhu Xiang, Ao Li, Ce Zhu, and Le Zhang. Towards real zero-shot camouflaged object segmentation without camouflaged annotations. *arXiv preprint arXiv:2410.16953*, 2024. 8
- [30] Tao Lei, Junwen Bai, Siddhartha Brahma, Joshua Ainslie, Kenton Lee, Yanqi Zhou, Nan Du, Vincent Zhao, Yuxin Wu, Bo Li, et al. Conditional adapters: Parameter-efficient transfer learning with fast inference. *Advances in Neural Information Processing Systems*, 36:8152–8172, 2023. 1, 3, 4
- [31] Guanbin Li and Yizhou Yu. Visual saliency based on multi-scale deep features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5455–5463, 2015. 2, 5
- [32] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022. 1, 2, 3
- [33] Mingbao Lin, Mengzhao Chen, Yuxin Zhang, Chunhua Shen, Rongrong Ji, and Liujuan Cao. Super vision transformer. *International Journal of Computer Vision*, 131(12): 3136–3151, 2023. 1
- [34] Ting Liu, Xuyang Liu, Liangtao Shi, Zunnan Xu, Siteng Huang, Yi Xin, and Qunjun Yin. Sparse-Tuning: Adapting vision transformers with efficient fine-tuning and inference. *arXiv preprint arXiv:2405.14700*, 2024. 1, 2, 3, 8
- [35] Weihuang Liu, Xi Shen, Chi-Man Pun, and Xiaodong Cun. Explicit visual prompting for low-level structure segmentations. In *CVPR*, pages 19434–19445, 2023. 3, 7, 2, 5, 6
- [36] Yifei Liu, Mathias Gehrig, Nico Messikommer, Marco Cannici, and Davide Scaramuzza. Revisiting token pruning for object detection and instance segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2658–2668, 2024. 3, 4
- [37] Yang Liu, Qiang Zhou, Jing Wang, Zhibin Wang, Fan Wang, Jun Wang, and Wei Zhang. Dynamic token-pass transformers for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1827–1836, 2024. 3, 4
- [38] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1
- [39] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 3
- [40] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6, 3
- [41] Yunqiu Lyu, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *CVPR*, 2021. 2, 5
- [42] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *CVPR*, pages 248–255, 2014. 6
- [43] Vida Movahedi and James H Elder. Design and perceptual validation of performance measures for salient object segmentation. In *2010 IEEE computer society conference on computer vision and pattern recognition-workshops*, pages 49–56. IEEE, 2010. 2, 5
- [44] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 1, 2, 6, 3
- [45] Youwei Pang, Xiaoqi Zhao, Tian-Zhu Xiang, Lihe Zhang, and Huchuan Lu. Zoomnext: A unified collaborative pyramid network for camouflaged object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 8
- [46] Namuk Park and Songkuk Kim. How do vision transformers work? In *International Conference on Learning Representations*, 2022. 2, 6, 1
- [47] Namuk Park, Wonjae Kim, Byeongho Heo, Taekyung Kim, and Sangdoo Yun. What do self-supervised vision transformers learn? In *International Conference on Learning Representations*, 2023. 1, 2
- [48] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *EACL*, pages 487–503, 2021. 3
- [49] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021. 1, 3, 4
- [50] Jianping Shi, Li Xu, and Jiaya Jia. Discriminative blur detection features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2965–2972, 2014. 6, 7, 2
- [51] Juan Silva, Aymeric Histace, Olivier Romain, Xavier Dray, and Bertrand Granado. Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer. *International journal of computer assisted radiology and surgery*, 9:283–293, 2014. 2
- [52] Przemysław Skurowski, Hassan Abdulameer, J Błaszczyk, Tomasz Depta, Adam Kornacki, and P Kozieł. Animal



- camouflage analysis: Chameleon database. *Unpublished manuscript*, 2(6):7, 2018. [2](#), [5](#)
- [53] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, 2002. [2](#)
- [54] Nima Tajbakhsh, Suryakanth R Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE transactions on medical imaging*, 35(2):630–644, 2015. [2](#), [6](#)
- [55] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. [1](#)
- [56] Di Wang, Jing Zhang, Bo Du, Gui-Song Xia, and Dacheng Tao. An empirical study of remote sensing pretraining. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–20, 2023. [3](#)
- [57] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, 2017. [6](#), [7](#), [2](#), [5](#)
- [58] Zhehao Wang, Xian Lin, Nannan Wu, Li Yu, Kwang-Ting Cheng, and Zengqiang Yan. Dtmformer: Dynamic token merging for boosting transformer-based medical image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5814–5822, 2024. [3](#), [4](#)
- [59] W.Liu, X.Shen, C.-M.Pun, and X.Cun. Explicit visual prompting for universal foreground segmentations. *arXiv preprint arXiv:2305.18476*, 2023. [3](#)
- [60] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision*. Springer, 2018. [1](#)
- [61] Zhenda Xie, Zigang Geng, Jingcheng Hu, Zheng Zhang, Han Hu, and Yue Cao. Revealing the dark secrets of masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14475–14485, 2023. [1](#)
- [62] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1155–1162, 2013. [2](#), [5](#), [6](#)
- [63] Hongxu Yin, Arash Vahdat, Jose Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. [1](#)
- [64] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019. [4](#)
- [65] Wangbo Zhao, Jiasheng Tang, Yizeng Han, Yibing Song, Kai Wang, Gao Huang, Fan Wang, and Yang You. Dynamic tuning towards parameter and inference efficiency for vit adaptation. *arXiv preprint arXiv:2403.11808*, 2024. [1](#), [3](#), [4](#), [6](#), [7](#), [8](#), [2](#), [5](#)
- [66] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. [2](#), [4](#)
- [67] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations (ICLR)*, 2022. [1](#), [3](#)