Disrupting Model Merging: A Parameter-Level Defense Without Sacrificing Accuracy

Supplementary Material

A. Overview

In this Appendix, we provide additional details and results that complement the main text:

- **Section** B introduces various model merging methods.
- **Section** C describes the Transformer-based model architecture used in our experiments, focusing on the MLP and multi-head attention modules.
- **Section** D details the MLP permutation optimization.
- **Section** E offers a proof of equivalence under our random multi-head scaling, ensuring the model's functionality remains unchanged.
- **Section** F presents the complete algorithmic steps of PaRaMS.
- **Section** G provides dataset descriptions for the image classification tasks.
- **Section** H shows additional classification results not included in the main paper, illustrating further evidence of our method's effectiveness.
- **Section** I includes extended experiments on image generation, highlighting the robustness of our defense across diverse prompts and scenarios.
- **Section** J lists other comparative results between different merging methods.

B. Introductions of Merging Methods

Weight Average (WA). WA assumes an equal contribution of each task vector in merged models and merges task vectors of multiple models into a single one by simple averaging: $f(\tau_1, \ldots, \tau_n) = \frac{1}{n}\Sigma_{i=1}^{n}\tau_i$.

AdaMerging. AdaMerging is also based on the weighted sum to aggregate task vectors. Nevertheless, it assumes that the task vector of different layer (i.e, τ_i^ℓ) has different effects in merging, and proposed a layer-wise coefficients $\Lambda_i = \{\lambda_i^1, \ldots, \lambda_i^L\}$. Specifically, merging coefficients Λ_i are calculated based on the entropy on an unlabeled held-out dataset, and the merging algorithm is formalized as: $f^\ell(\tau_1, \ldots, \tau_n) = \lambda_i^\ell \Sigma_{i=1}^{n}\tau_i^\ell$ for the ℓ-th layer. Different from other merging algorithms, additional calculation is required on searching for Λ_i.

TIES-Merging (TIES). TIES is a plug-in for model merging methods, which resolve task conflicts in merging by TRIM, ELECT SIGN and MERGE on task vectors. The merging performs as: $f(\tau_1, \ldots, \tau_n) = \lambda\Sigma_{i=1}^{n}\phi(\tau_i)$ where ϕ is combined TIES operations and $\lambda = 0.3$ maximize the merging performance empirically, the same as TA.

Drop And REscale (DARE). DARE is also a plug-in for model merging methods like TIES. Following a drop and rescale flow, DARE first perform random drop on τ_i based on a drop rate p (i.e., setting their values to zeros), and rescales the remaining weights by $1/(1-p)$. DARE often retains or enhances the performance of model merging methods with even 90% task vectors removed.

C. Model Architecture

Our method is specially designed for Transformer-based architectures, in which each layer (often called a Transformer block) combines a multi-head attention submodule with a MLP. This design has become a core building block of modern deep learning models, including ViTs, CLIP models, Stable Diffusion models and LLMs such as LLaMA. Since the proposed method is closely related to the structure of MLP and Attention block, we briefly describe how the MLP and Attention submodules operate within each Transformer block.

MLP. The MLP submodule applies nonlinear transformations to each position's feature vector, often scaling dimensions in the hidden layer. Let $X \in \mathbb{R}^d$ be the feature vector at a single position (for simplicity, omitting batch and sequence dimensions). A typical two-layer MLP computes

$$\mathrm{MLP}(X) \;=\; W_2\,\sigma\big(W_1\,X + b_1\big) \;+\; b_2,$$

where $W_1 \in \mathbb{R}^{d_{\text{hidden}} \times d}$, $W_2 \in \mathbb{R}^{d \times d_{\text{hidden}}}$ (with biases b_1, b_2) are learnable parameters, and $\sigma(\cdot)$ is a nonlinear activation (e.g., GELU). This per-position feed-forward step enhances the network's expressive power without introducing dependencies across positions.

Structure of Multi-head Attention Block. Consider having h parallel attention heads, each with dimensionality d_k. Suppose the input sequence is represented by

$$x \in \mathbb{R}^{\text{seq} \times d_{\text{model}}}.$$

A linear mapping first produces $Q, K, V \in \mathbb{R}^{\text{seq}\times(h\times d_k)}$. We then split these along the last dimension into h parts:

$$\begin{aligned} Q &\rightarrow \big[Q_1, \ldots, Q_h\big], \\ K &\rightarrow \big[K_1, \ldots, K_h\big], \\ V &\rightarrow \big[V_1, \ldots, V_h\big], \end{aligned}$$

where each $Q_i, K_i, V_i \in \mathbb{R}^{\text{seq}\times d_k}$ corresponds to the i-th attention head. For the i-th head, the attention output is given by

$$\text{Attn}\big(Q_i, K_i, V_i\big) = \text{softmax}\Big(\frac{Q_i\, K_i^{\mathsf{T}}}{\sqrt{d_k}}\Big)\, V_i.$$

The outputs of the h heads are then concatenated and projected via an output weight $W_O \in \mathbb{R}^{(h\times d_k)\times d_{\text{model}}}$:

$$\begin{aligned} &\text{Attention}\big(Q, K, V\big) = \\ &\Big[\text{Attn}\big(Q_1, K_1, V_1\big),\ \ldots,\ \text{Attn}\big(Q_h, K_h, V_h\big)\Big]\, W_O. \end{aligned}$$

D. MLP Permutation Optimization

Considering the optimization problem in Section Method:

$$\arg\max_{\eta_{\text{perm}}} \big\|\theta_{\text{pre}}^{\text{MLP}} - \eta_{\text{perm}}(\theta_{\text{def}}^{\text{MLP}})\big\|^2 = \arg\min_{\eta_{\text{perm}}}\ \theta_{\text{pre}}^{\text{MLP}} \cdot \eta_{\text{perm}}(\theta_{\text{def}}^{\text{MLP}}).$$

Which can be re-expressed in the following term in a 2-layer MLP:

$$\begin{aligned} \arg\min_{\eta_{\text{perm}}=\{P_i\}} \sum_{i=1}^{n} [&\langle W_{\text{premlp1}}^{(i)}, P_i W_{\text{defmlp1}}^{(i)} \rangle_F \\ &+ \langle W_{\text{premlp2}}^{(i)}, W_{\text{defmlp2}}^{(i)} P_i^{\mathsf{T}} \rangle_F], \end{aligned}$$

where $\langle A, B\rangle_F$ denotes the Frobenius inner productbetween real-valued matrices A and B. Hence, the optimization could be re-expressed and solved as a linear assignment problem.

E. Proof of Equivelance based on Random Scaling

First, scaling on Q_i and K_i keeps attention weights unchanged. Suppose we multiply Q_i by a diagonal matrix A_i and simultaneously multiply K_i by A_i^{-1}. Then

$$\frac{Q_i'\, K_i' T}{\sqrt{d_k}} \;=\; \frac{(Q_i\, A_i)\,\big(K_i\, A_i^{-1}\big)^{\mathsf{T}}}{\sqrt{d_k}} \;=\; \frac{Q_i\, K_i^{\mathsf{T}}}{\sqrt{d_k}},$$

ensuring that the attention score matrix and thus the softmax weights remain identical to the original.

Scaling V_i and the output projection W_O is also an inverse pair. We could multiply V_i by diagonal matrix B_i (possibly channelwise or headwise) and compensate by multiplying the corresponding block in the output projection by B^{-1}. Concretely, the single-head output keeps identical

$$\begin{aligned} &\text{Attention}\big(Q_i', K_i', V_i'\big) = \\ &\text{softmax}\Big(\frac{Q_i\, K_i^{\mathsf{T}}}{\sqrt{d_k}}\Big)\,(V_i B_i) B_i^{-1} W_O[:, i] = \\ &\text{Attention}\big(Q_i, K_i, V_i\big), \end{aligned}$$

hence, each head i can adopt

$$\begin{aligned} Q_i' &= Q_i\, A_i,\ \ K_i' = K_i\, A_i^{-1} \\ V_i' &= V_i\, B_i,\ \ W_O'[:, i] = B_i^{-1}\, W_O[:, i], \end{aligned}$$

so that the multi-head attention output ends up identical to its original form.

F. Algorithm of PaRaMS

The algorithm consists of 2 subblocks: Parameter Rearrangement for MLP block and Random Multi-head Scaling for attention block. The pseudocode for our method is shown in Algorithm 1.

Algorithm 1 PaRaMS

Input: Fine-tuned model θ_{def}, pretrained checkpoint θ_{pre}, scaling range $[s_{\min}, s_{\max}]$
Output: Modified model $\hat{\theta}$

$\hat{\theta} \leftarrow \theta_{\text{def}}$
Step 1: MLP Parameter Rearrangement
for each MLP layer ℓ **do**
  $P^{(\ell)} \leftarrow \arg\max_P \Big\| P^{(\ell)}\big(\theta_{\text{def}}^{(\ell)}\big) - \theta_{\text{pre}}^{(\ell)}\Big\|^2$
  $W_1^{(\ell)} \leftarrow P^{(\ell)}\, W_1^{(\ell)}$
  $W_2^{(\ell)} \leftarrow W_2^{(\ell)}\,\big(P^{(\ell)}\big)^{\top}$
  $b_1^{(\ell)} \leftarrow P^{(\ell)}\, b_1^{(\ell)}$
end for
Step 2: Random Multi-Head Scaling
for each layer j containing multi-head attention **do**
  for each attention head i **do**
    $\mathbf{a}_i \sim \mathcal{U}(s_{\min}, s_{\max})^{d_k}$
    $A_i \leftarrow \text{diag}(\mathbf{a}_i) \in \mathbb{R}^{d_k\times d_k}$
    $Q_{j,i} \leftarrow Q_{j,i} A_i$
    $K_{j,i} \leftarrow K_{j,i} A_i^{-1}$
    $\mathbf{b}_i \sim \mathcal{U}(s_{\min}, s_{\max})^{d_k}$
    $B_i \leftarrow \text{diag}(\mathbf{b}_i) \in \mathbb{R}^{d_k\times d_k}$
    $V_{j,i} \leftarrow V_{j,i} B_i$
    $W_{\text{out}}[j, :, i] \leftarrow B_i^{-1} W_{\text{out}}[j, :, i]$
  end for
end for
return $\hat{\theta}$

G. Dataset Discriptions used in Image Classification Task

Cars. Cars dataset comprises high-resolution images of cars, with a focus on fine-grained vehicle classification. It contains about 16,000 images spanning 196 car models, making it a benchmark for fine-grained recognition tasks.

RESISC45. RESISC45 is a remote sensing image dataset containing 31,500 images from 45 scene classes (e.g., airports, industrial areas, harbors). Each class has 700 images, facilitating the study of aerial scene classification.

SVHN. SVHN dataset features real-world digit images extracted from Google Street View. It includes over 600,000 labeled digits, commonly used for digit recognition under challenging, cluttered backgrounds.

GTSRB. GTRSB consists of over 50,000 images covering 43 traffic sign classes. It is widely used to evaluate classification performance in real-world traffic scenarios.

MNIST. MNIST is a classic dataset of handwritten digit images (0–9), comprising 70,000 grayscale images (60,000 for training, 10,000 for testing). It remains a foundational benchmark for evaluating basic image classification methods.

EuroSAT. EuroSAT is a satellite image dataset derived from Sentinel-2 data, covering 10 land-use and land-cover classes (e.g., forest, residential). It contains 27,000 labeled images, serving as a testbed for remote-sensing scene classification.

DTD. DTD Dataset includes 5,640 images of texture patterns grouped into 47 classes (e.g., banded, porous, grid). It focuses on texture-centric classification and is often used to assess a model's ability to capture fine-grained visual attributes.

H. Other Results on Image Classification

H.1. Evaluation when more than two models are merged

We further investigate the scenario where more than two models are merged simultaneously with task arithmetic. Specifically, we evaluate the performance of merging 2 to 7 finetuned models at once, examining whether our defense remains effective when merging with more than 2 models. When merging with more than 2 models, the common scaling coefficient is usually set as $\lambda = 0.3$, we follow this setting. In Figure 1, we can observe performance of MMP- is at least larger than 60% (green and blue line) while the performance of MMP+ is at most less than 10% (orange and red line), this means our proposal is still effective with different number of merged model. The Benign curves (ViT-B-32 in blue, ViT-L-14 in green) show that as the number of merged models increases, the average accuracy gradually declines (e.g., from around 85% to 63% on ViT-B-32, and from around 90% to 77% on ViT-L-14). In contrast, the Protected curves (ViT-B-32 in orange, ViT-L-14 in red) remain consistently low (around 6–7%), indicating that once our defense is applied, merging multiple models fails to preserve any meaningful performance. This result demonstrates that our protection method continues to be effective even in scenarios where more than two models are merged.

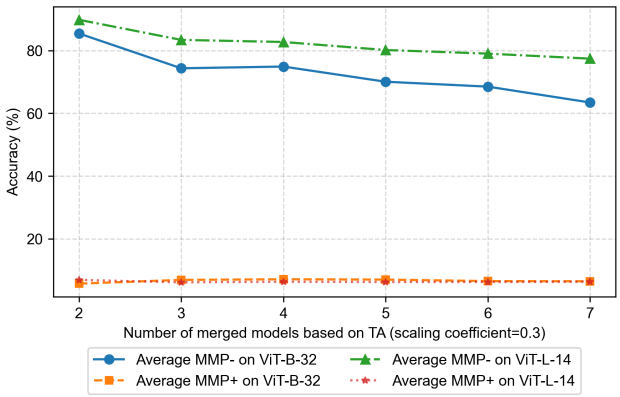

Figure 1. The classification accuracy of MMP-/MMP+ (merged by TA) with different numbers of merged models (2 to 7) Here, defender's model is trained on MNIST, and the other six tasks as free-riders' task) with a scaling coefficient of 0.3, evaluated on ViT-B-32 and ViT-L-14.

H.2. Evaluation TA performance under different scaling coefficient

Since the performance of TA is affected by a merge coefficient, we then measure whether different coefficients affect our proposal. By adjusting the coefficient from 0.3 to 0.8, we observe how the performance of MMP- and MMP+ evolves when combining multiple specialized models, as shown in Figure 2. Here, the defender is finetuned on EuroSAT, and the free-rider consists of six other tasks. In the figure, we can observe that the performance of MMP- (lines in different shades of blue) is at least larger than 60%, and the performance of MMP+ (lines in different shades of red) is at most less than 25%. This demonstrates the effectiveness of our method in preventing free-riders from gaining the specialized capabilities of the defender's model, regardless of the TA scaling coefficient.

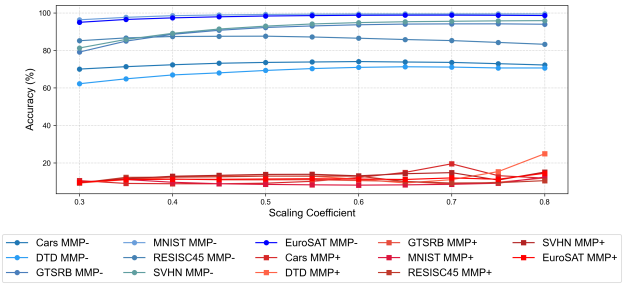

Figure 2. The classification accuracy of MMP-/MMP+ (merged by task arithmetic) with coefficient 0.3 to 0.8. Here, the defenser's model is trained on EuroSAT, and we have six different free-rider tasks. Here, model is ViT-B-32.

Table 1. Ablation study based on ViT-L-14. Defender: Cars. Free-rider: other six tasks. Merging method: TA (scaling coefficient=0.8).

Setting	Average accuracy (%) of MMP+
Rearrangement Only	5.90
Scaling Only	10.92
PaRaMS	4.99

H.3. Ablation Study

Our method comprises two key modules, MLP parameter rearrangement and random multi-head scaling. They jointly disrupt model merging while preserving the model's functionality. To examine how each module individually contributes to this disruption, we perform an ablation study by omitting one module at a time. Specifically, we compare the avarage accuracy of MMP+ based on ViT-L-14 and TA (scaling coefficient=0.8). Results are shown in Table 1.

Table 1 shows that even with a single module (Rearrangement Only or Scaling Only), the MMP+ accuracy still exhibits significant degradation compared to benign merging scenarios. This indicates that either MLP rearrangement or random multi-head scaling alone can effectively disrupt model merging, thereby providing flexibility when only one type of parameter manipulation is applicable in merging. Nonetheless, employing both modules (PaRaMS) yields an even stronger defense, as it further increases the parameter distance across both MLP and attention modules. We thus conclude that each module individually contributes substantially to disrupting model merging, yet their combination provides a more robust protection.

H.4. Computation Cost Comparison

We further compare the computational costs associated with the original model merging, our proposed defense (PaRaMS), and an ultimate adaptive attack based on knowledge distillation (KD). This experiment is conducted on a Windows 11 platform equipped with AMD Ryzen 9 9950X CPU and Nvidia A6000 Ada GPU. For the KD setting, we fine-tune the model for 50 epochs anda a 128 batch size. The average computation time across seven datasets is presented in Table 2.

Table 2. Average computational cost comparison of merging based on TA, applying PaRaMS and performing knowledge distillation as adaptive method.

Setting/Model	ViT-B-32	ViT-L-14
Task Arithmetic	1.59s	5.08s
PaRaMS	57.32s	173.05s
Knowledge Distillation	3.82h	11.42h
Finetune From Pretrain	2.57h	7.83h

As shown in Table 2, PaRaMS is computationally lightweight, requiring only 57.32 seconds on average for ViT-B-32 and 173.05 seconds for ViT-L-14. In comparison, the standard model merging based on Task Arithmetic is extremely efficient, taking less than 10 seconds, thus significantly lowering the barrier for potential free-riders, and highlighting the risks of intellectual property (IP) infringement associated with lightweight merging methods. On the other hand, the adaptive attack via KD incurs substantially higher computational costs (near 4 hours for ViT-B-32 and 12 hours for ViT-L-14). Moreover, it requires access to the defender's private training data, which is typically unavailable since model publishers rarely share their proprietary datasets publicly. Therefore, the KD approach is not practically feasible in most open-source scenarios.

I. Other Results on Image Generation

We further show several sets of generated images from UMP-, UMP+, free-rider, MMP- and MMP+. For Figure 3, the defender is an Anime-based SD1.5 model, and the free-rider is a reality-Europe SD1.5 model, both from HuggingFace.

In these generated images, UMP- (the unprotected single model) and UMP+ (the protected single model) both faithfully capture the prompt details—such as "Angel Goku" standing before the Eiffel Tower or "Naruto Uzumaki" in Rome—indicating that our defense does not degrade the protected model's original generative performance. Conversely, MMP- (the benign merged model) successfully blends concepts from both the defender and free-rider, as shown by coherent scenes like "Crayon Shin-chan eating spaghetti in an European restaurant."

Once protection is applied, however, MMP+ (the protected merged model) fails to inherit the defender's specialized knowledge, resulting in heavily distorted or random artifacts. For instance, the final column in each row often appears corrupted or nonsensical, demonstrating that the free-rider cannot exploit the protected model's fine-tuned capabilities through merging. Overall, these results underscore

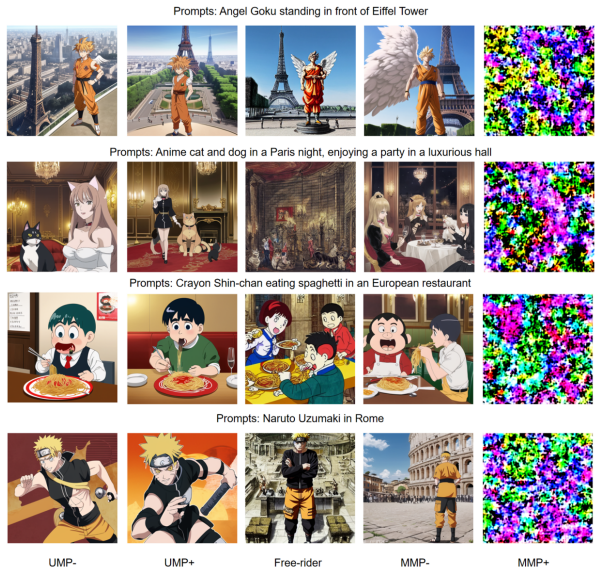

Figure 3. Generated images from UMP-, UMP+, MMP-, MMP+. Each row is related to one prompt set.

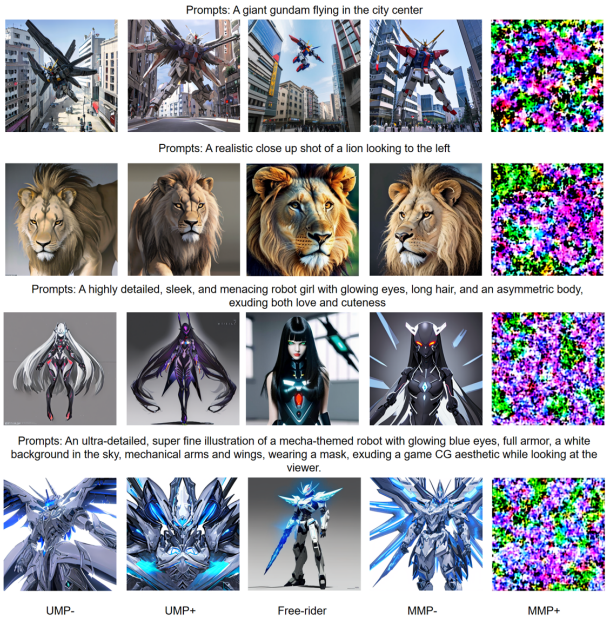

Figure 4. Generated images from UMP-, UMP+, MMP-, MMP+. Each row is related to one prompt set.

that our defense preserves the defender's performance while rendering merged outputs unusable.

We also show generation examples of Animated-Gundam style and realistic style in Figure 4.

In these examples, UMP-/UMP+ specializes in an anime–mecha aesthetic (e.g., "giant gundam" and "robot girl"), while the free-rider focuses on more realistic imagery (e.g., "close-up shot of a lion"). The UMP- and UMP+ both render their respective prompts with high fidelity—indicating our defense does not degrade the defender's own mecha–anime style. Meanwhile, the free-rider model excels at realism, as seen in the lion images.

For MMP-, the outputs successfully blend the anime mecha concepts with the free-rider's realistic style, producing coherent "hybrid" results. However, once the defender is protected, MMP+ fails to inherit the mecha–anime capabilities and instead generates heavily corrupted or noisy outputs. This underscores that while our defense preserves the defender's fine-tuned strengths, it prevents any merged model from cheaply acquiring those specialized skills.

J. Comprehensive Results of Different Merging Methods

We show one set of results on TA without DARE based on Vit-B-32 in the paper, and overall average accuracy of eight merging settings on three architectures. Here we show the other seven settings of ViT-B-32 in the following.

The tables show a same result that our method efficiently disturbs model merging in all the following settings on all datasets.

Table 3. The classification accuracy of MMP-/MMP+ (merged by TA with DARE) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32 ($\lambda = 0.8$).

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	69.93/93.95 0.52/2.84	71.99/99.80 0.55/9.93	67.65/97.30 0.60/8.36	67.53/98.53 0.47/3.35	67.62/99.67 0.52/9.00	70.17/76.70 0.39/1.65
	RESISC45	93.95/69.93 2.33/0.56	*NA*	83.14/99.90 2.60/3.83	90.51/96.82 1.95/7.41	90.02/97.86 1.76/0.82	90.49/99.57 1.22/11.16	93.22/72.45 2.11/2.13
	EuroSAT	99.80/71.99 9.15/0.53	99.90/83.14 9.33/2.32	*NA*	98.11/95.89 11.13/9.00	98.11/93.33 10.65/3.80	96.98/99.48 19.04/9.66	99.72/70.48 12.44/1.81
	SVHN	97.30/67.65 9.57/0.47	96.82/90.51 9.23/1.79	95.89/98.11 11.46/8.78	*NA*	94.64/94.18 8.53/2.22	92.27/99.40 9.96/10.88	96.95/67.77 9.21/2.34
	GTSRB	98.53/67.53 1/59/0.62	97.86/90.02 1.47/2.57	93.33/98.11 1.85/9.31	94.18/96.64	*NA*	91.96/99.35 2.03/9.24	98.18/67.39 2.41/2.61
	MNIST	99.67/67.62 9.71/0.47	99.57/90.49 9.51/2.19	99.48/96.98 8.54/11.50	99.40/92.27 9.82/9.07	99.35/91.96 9.72/3.28	*NA*	99.67/70.21 10.36/2.18
	DTD	76.70/70.17 2.55/0.45	72.45/93.22 2.23/1.73	70.48/99.72 1.76/11.93	67.77/96.95 2.18/8.02	67.39/98.18 2.07/1.69	70.21/99.67 2.18/9.94	*NA*

Table 4. The classification accuracy of MMP-/MMP+ (merged by SA) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	73.20/92.87 0.46/2.68	73.60/99.30 0.46/9.24	71.87/95.36 0.46/7.56	73.42/96.89 0.46/3.09	73.00/99.41 0.46/9.58	74.28/72.07 0.46/2.13
	RESISC45	92.87/73.20 2.62/0.51	*NA*	87.60/98.93 2.27/10.69	90.92/94.66 2.94/8.11	91.11/95.33 3.05/1.73	91.10/99.43 3.73/9.24	92.14/70.11 3.06/2.39
	EuroSAT	99.30/73.60 9.17/0.52	98.93/87.60 12.93/2.52	*NA*	97.67/92.97 13.85/13.79	97.11/92.24 10.94/2.14	97.96/99.18 8.52/9.84	99.15/69.31 13.00/2.13
	SVHN	95.36/71.87 9.69/0.36	94.66/90.92 9.69/2.57	92.97/97.67 9.69/9.67	*NA*	94.74/94.83 9.69/3.16	92.50/99.37 9.69/9.82	95.36/70.53 9.69/2.82
	GTSRB	96.89/73.42 3.09/0.44	95.33/91.11 3.09/2.52	92.24/97.11 3.18/14.54	94.83/94.74 3.09/8.19	*NA*	93.25/99.12 3.17/8.95	96.78/70.85 3.09/2.13
	MNIST	99.41/73.00 9.26/0.47	99.43/91.10 8.61/2.10	99.18/97.96 9.78/13.80	99.37/92.50 9.82/9.70	99.12/93.25 9.82/5.87	*NA*	99.45/69.10 9.82/2.29
	DTD	72.07/74.28 1.81/0.62	70.11/92.14 1.76/2.44	69.31/99.15 1.76/19.67	70.53/95.36 1.65/14.74	70.85/96.78 1.86/2.91	69.10/99.45 1.91/11.02	*NA*

Table 5. The classification accuracy of MMP-/MMP+ (merged by SA with DARE) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	73.34/92.97 0.42/2.21	73.36/99.28 0.47/8.22	71.81/95.33 0.51/9.45	73.62/96.90 0.61/2.38	72.89/99.43 0.44/9.81	73.78/71.81 0.63/1.81
	RESISC45	92.97/73.34 2.10/0.56	*NA*	87.52/98.91 2.14/4.56	90.92/94.55 2.37/7.58	91.06/95.29 1.94/1.96	90.95/99.45 2.51/9.41	91.94/69.63 2.27/2.82
	EuroSAT	99.28/73.36 12.57/0.58	98.91/87.52 11.41/2.86	*NA*	97.65/92.96 9.26/7.32	97.19/92.14 6.59/2.83	97.93/99.20 9.26/10.21	99.07/68.99 8.20/1.76
	SVHN	95.33/71.81 6.69/0.47	94.55/90.92 7.54/2.52	92.96/97.65 9.56/10.76	*NA*	94.70/94.81 14.11/3.10	92.39/99.34 8.75/9.50	95.39/70.32 7.76/2.02
	GTSRB	96.90/73.62 2.71/0.56	95.29/91.06 2.13/2.10	92.14/97.19 2.97/7.94	94.81/94.70 2.12/7.58	*NA*	93.26/99.09 3.44/11.31	96.75/70.69 2.15/1.60
	MNIST	99.43/72.89 10.44/0.52	99.45/90.95 9.67/2.51	99.20/97.93 9.05/11.22	99.34/92.39 9.31/8.23	99.09/93.26 9.72/2.92	*NA*	99.45/68.83 8.04/2.93
	DTD	71.81/73.78 2.18/0.51	69.63/91.94 2.45/2.62	68.99/99.07 2.23/6.65	70.32/95.39 2.34/8.68	70.69/96.75 1.97/1.19	68.83/99.45 2.02/9.73	*NA*

Table 6. The classification accuracy of MMP-/MMP+ (merged by TIES) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	75.96/93.35 0.53/2.11	76.11/99.11 0.65/9.24	75.28/96.27 0.58/9.22	75.95/96.60 0.58/3.09	76.43/99.57 0.57/8.92	75.46/74.57 0.62/2.18
	RESISC45	93.35/75.96 3.22/0.52	*NA*	89.44/98.41 2.75/19.61	92.27/96.10 2.86/7.85	92.86/95.76 3.05/3.08	92.62/99.60 3.02/9.85	92.78/73.56 3.16/2.29
	EuroSAT	99.11/76.11 11.19/0.58	98.41/89.44 12.37/1.84	*NA*	96.96/95.39 11.24/7.53	96.87/92.61 12.26/3.79	98.15/99.51 11.39/8.33	98.63/72.93 11.93/2.13
	SVHN	96.27/75.28 9.18/0.47	96.10/92.27 9.17/2.11	95.39/96.96 9.16/10.93	*NA*	95.56/95.17 9.18/4.81	93.89/99.36 9.16/8.92	96.43/73.09 9.16/2.45
	GTSRB	96.60/75.95 2.95/0.53	95.76/92.86 3.00/3.41	92.61/96.87 2.96/9.19	95.17/95.56 2.90/9.56	*NA*	94.61/99.44 3.01/10.02	97.17/74.52 2.89/1.86
	MNIST	99.57/76.43 9.82/0.58	99.60/92.62 9.82/2.52	99.51/98.15 9.82/10.13	99.36/93.89 9.82/9.69	99.44/94.61 9.82/2.88	*NA*	99.59/72.82 9.82/2.13
	DTD	74.57/75.46 2.13/0.60	73.56/92.78 2.13/2.43	72.93/98.63 2.13/18.54	73.09/96.43 2.13/6.43	74.52/97.17 2.13/3.09	72.82/99.59 2.13/9.74	*NA*

Table 7. The classification accuracy of MMP-/MMP+ (merged by TIES with DARE) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	75.76/90.84 0.57/2.40	76.16/98.78 0.50/11.69	75.67/95.44 0.57/8.83	76.12/95.55 0.46/2.36	76.32/99.53 0.51/10.33	75.03/71.33 0.71/2.07
	RESISC45	90.84/75.76 2.13/0.57	*NA*	92.63/97.13 2.70/10.81	94.89/93.90 2.11/7.55	95.06/89.15 2.10/2.08	95.02/99.54 2.21/9.73	95.30/66.17 2.05/2.18
	EuroSAT	98.78/76.16 11.76/0.47	97.13/92.63 9.09/3.05	*NA*	99.50/89.06 9.06/8.95	99.56/82.42 11.41/1.35	99.65/99.12 5.57/9.03	99.81/64.47 7.48/2.13
	SVHN	95.44/75.67 12.92/0.53	93.90/94.89 9.12/2.51	89.06/99.50 8.53/9.39	*NA*	96.81/87.63 8.92/2.08	96.25/99.05 10.11/10.10	97.26/62.45 8.77/1.86
	GTSRB	95.55/76.12 2.58/0.51	89.15/95.06 1.80/1.97	82.42/99.56 0.80/7.83	87.63/96.81 1.94/8.84	*NA*	98.27/98.63 2.53/10.73	98.73/61.70 1.38/2.77
	MNIST	99.53/76.32 9.62/0.49	99.54/95.02 9.16/1.98	99.12/99.65 11.07/5.93	99.05/96.25 9.48/8.03	98.63/98.27 10.52/2.06	*NA*	99.67/60.53 9.33/2.29
	DTD	71.33/75.03 2.02/0.47	66.17/95.30 2.45/2.56	64.47/99.81 2.82/11.98	62.45/97.26 2.18/8.31	61.70/98.73 1.70/2.15	60.53/99.67 2.02/8.79	*NA*

Table 8. The classification accuracy of MMP-/MMP+ (merged by AdaMerging) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	74.01/90.67 0.61/2.52	73.72/96.81 0.57/21.74	73.51/83.82 0.56/11.25	73.71/95.69 0.55/3.04	74.56/96.51 0.57/9.68	73.80/66.12 0.67/2.23
	RESISC45	90.67/74.01 2.81/0.55	*NA*	87.29/97.09 2.60/13.06	90.29/83.99 2.57/7.64	89.10/95.46 2.90/2.14	90.51/96.77 2.68/9.51	90.59/66.17 2.84/2.93
	EuroSAT	96.81/73.72 9.26/0.51	97.09/87.29 9.26/2.52	*NA*	95.91/82.77 9.20/11.74	95.00/95.15 9.24/3.80	96.50/96.32 9.26/8.35	96.76/66.33 9.26/1.12
	SVHN	83.82/73.51 7.82/0.55	83.99/90.29 7.70/2.43	82.77/95.91 7.68/13.17	*NA*	87.63/95.91 7.94/3.09	87.77/98.22 7.79/9.58	85.15/66.81 7.64/2.29
	GTSRB	95.69/73.71 1.46/0.55	95.46/89.10 1.70/3.16	95.15/95.00 1.49/11.41	95.91/87.63 1.54/8.10	*NA*	95.72/96.58 1.47/9.97	95.66/66.54 1.74/2.13
	MNIST	96.51/74.56 6.68/0.44	96.77/90.51 5.32/2.78	96.32/96.50 6.84/11.81	98.22/87.77 7.53/19.11	96.58/95.72 8.28/6.03	*NA*	96.73/66.38 8.37/2.18
	DTD	66.12/73.80 1.91/0.36	66.17/90.59 2.07/4.00	66.33/96.76 1.97/8.41	66.81/85.15 2.13/12.69	66.54/95.66 2.34/3.09	66.38/96.73 1.86/8.92	*NA*

Table 9. The classification accuracy of MMP-/MMP+ (merged by AdaMerging with DARE) on $\mathcal{T}_{\text{def}}$ and $\mathcal{T}_{\text{fr}}$ on ViT-B-32.

MMP- Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$ MMP+ Accuracy (%) on $\mathcal{T}_{\text{def}}$/$\mathcal{T}_{\text{fr}}$		$\mathcal{T}_{\text{fr}}$						
		Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
$\mathcal{T}_{\text{def}}$	Cars	*NA*	73.85/90.41 0.51/1.87	73.90/96.83 0.56/9.28	73.27/83.97 0.50/8.70	73.68/95.63 0.34/3.06	74.37/96.55 0.47/11.63	73.97/66.33 0.49/2.34
	RESISC45	90.41/73.85 1.94/0.50	*NA*	87.30/96.91 2.33/9.15	90.30/83.87 2.60/9.69	89.10/95.49 2.40/3.26	90.32/96.72 2.02/9.63	90.70/66.38 3.17/2.61
	EuroSAT	96.83/73.90 13.59/0.51	96.91/87.30 9.48/1.97	*NA*	95.89/83.01 13.43/13.14	94.94/94.96 11.30/1.84	96.56/96.40 7.91/10.28	96.70/66.12 15.44/1.97
	SVHN	83.97/73.27 9.70/0.50	83.87/90.30 8.09/2.19	83.01/95.89 9.75/11.30	*NA*	87.80/95.95 9.15/1.36	87.83/98.21 7.24/11.10	84.95/66.97 8.55/1.86
	GTSRB	95.63/73.68 1.59/0.56	95.49/89.10 1.36/1.81	94.96/94.94 3.09/9.33	95.95/87.80 2.12/7.62	*NA*	95.62/96.62 1.76/9.40	95.61/66.44 2.89/3.09
	MNIST	96.55/74.37 10.28/0.46	96.72/90.32 9.82/1.17	96.40/96.56 10.10/11.11	98.21/87.83 8.51/7.99	96.62/95.62 10.12/2.16	*NA*	96.77/66.38 9.52/1.44
	DTD	66.33/73.97 2.23/0.44	66.38/90.70 2.29/2.11	66.12/96.70 2.13/9.04	66.97/84.95 2.13/9.04	66.44/95.61 2.29/2.57	66.38/96.77 2.39/9.10	*NA*