DC-Merge: Improving Model Merging with Directional Consistency

Han-Chen Zhang* Zi-Hao Zhou* Mao-Lin Luo Shimin Di
Min-Ling Zhang Tong Wei†

[1]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[2]Key Laboratory of Computer Network and Information Integration (Southeast University)
{hanchenzh, zhouzih, weit}@seu.edu.cn

Abstract

Model merging aims to integrate multiple task-adapted models into a unified model that preserves the knowledge of each task. In this paper, we identify that the key to this knowledge retention lies in maintaining the directional consistency of singular spaces between merged multi-task vector and individual task vectors. However, this consistency is frequently compromised by two issues: i) an imbalanced energy distribution within task vectors, where a small fraction of singular values dominate the total energy, leading to the neglect of semantically important but weaker components upon merging, and ii) the geometric inconsistency of task vectors in parameter space, which causes direct merging to distort their underlying directional geometry. To address these challenges, we propose DC-Merge, a method for directional-consistent model merging. It first balances the energy distribution of each task vector by smoothing its singular values, ensuring all knowledge components are adequately represented. These energy-balanced vectors are then projected onto a shared orthogonal subspace to align their directional geometries with minimal reconstruction error. Finally, the aligned vectors are aggregated in the shared orthogonal subspace and projected back to the original parameter space. Extensive experiments on vision and vision-language benchmarks show that DC-Merge consistently achieves state-of-the-art performance in both full fine-tuning and LoRA settings. The implementation code is available at https://github.com/Tobeginwith/DC-Merge.

1. Introduction

Pre-trained models are the foundation of modern machine learning systems [3, 4, 52, 71]. In practice, they are typically fine-tuned for specialization in specific tasks [17, 18, 28, 65]. A growing body of research has focused on *model merging* [37], which integrates multiple task-adapted models into a unified model while preserving each task's capability. Many methods have been proposed to improve the effectiveness of model merging by reducing sign conflicts [68], by aligning gradients [11], or through magnitude-based selection [12, 30, 45]. Despite its potential to enable efficient multi-task adaptation without retraining, existing approaches often suffer performance degradation after merging [37], especially when tasks originate from heterogeneous domains [11]. Recent studies aim to reduce the interference among different tasks [6, 20, 43], while the underlying mechanism of how task-specific capabilities are preserved after merging remains underexplored. A central question thus arises: *what property must be preserved to retain each task's ability after merging?*

Following Task Arithmetic (TA) [29], we define a task vector as the parameter difference between a fine-tuned model and its pre-trained counterpart. Each task vector can be decomposed via singular value decomposition (SVD) into a set of orthogonal *knowledge vectors*, each representing a distinct adaptation direction weighted by its singular value. We term each of these directions as a *knowledge component* and the corresponding singular values reflect the energy distributed across these components. We observe that task performance after model merging primarily depends on the *directional consistency* between the merged and original task vectors. Specifically, as long as the directions of the knowledge components are preserved, the merged model retains most task capabilities, even if their energy distribution changes. In contrast, slight directional deviations significantly degrade performance, indicating that maintaining directional consistency of knowledge components is crucial to maintaining task performance.

To quantify this consistency, we propose *directional similarity* (DirSim), which measures the consistency of directional geometry between two task vectors while discounting the influence of energy distribution. Unlike cosine similarity, which emphasizes consistency of high-energy compo-

*Equal contribution, †Corresponding author

nents, DirSim also accounts for the directional consistency of weaker yet semantically informative components. Empirically, DirSim shows a strongly positive correlation with post-merge task-wise performance, validating it as a reliable indicator of knowledge preservation.

Despite its importance, directional consistency is often violated by two fundamental issues. First, the energy distribution of task vectors is imbalanced, where a few singular values capture most of the energy (as shown in Figure 1), causing the model to overemphasize on high-energy directions and thereby hindering generalization and directional geometry preservation. Second, directly merging task vectors in the original parameter space leads to basis misalignment: different tasks span heterogeneous low-rank subspaces whose orientations are not geometrically aligned. Consequently, the merged task vector fails to preserve the directional geometry of each task vector that characterizes the task's knowledge. To address these challenges, we propose a new method called *DC-Merge*, which explicitly enforces directional consistency between the merged multi-task vector and each original task vector. DC-Merge consists of two complementary modules: i) *energy smoothing* redistributes the singular values of each task vector to balance the energy distribution of its knowledge components, thereby preventing the merging process from overlooking weaker but semantically rich directions within each task vector. ii) *cover space merging* then projects all smoothed task vectors into a shared orthogonal subspace before aggregation, ensuring that merging occurs under a consistent cover basis without cross-task directional interference. Together, these modules preserve the task directional geometry during merging, enabling stable multi-task compatibility and strong generalization. Extensive experiments on both full fine-tuning (FFT) and LoRA [26] setups show that DC-Merge achieves state-of-the-art results on both vision and vision-language benchmarks while maintaining high directional consistency with original task vectors.

In summary, our key contributions are as follows:

- We correlate the model merging performance with a novel concept *directional consistency* between the merged multi-task vector and individual task vectors.
- We introduce DirSim, a new metric that isolates directional consistency from energy distribution effects. DirSim shows a strong positive correlation with the performance of merged model.
- We propose *DC-Merge*, a method that enhances directional consistency by first balancing energy distribution of task vectors and then merging them within a shared orthogonal subspace.
- Extensive experiments on vision and vision-language benchmarks demonstrate that DC-Merge achieves state-of-the-art performance in both FFT and LoRA settings.

2. Directional Consistency Matters

This section reveals the intrinsic imbalance of energy distribution across knowledge components and presents empirical evidence confirming the importance of directional consistency in model merging. We also introduce a new metric to quantify this consistency. Unless otherwise specified, the experiments in this section are based on a `ViT-B-32` visual encoder [15] under LoRA configuration.

2.1. Preliminary

Model Merging. Given a pre-trained parameter set $\boldsymbol{W}_0$ and a collection of fine-tuned models $\{\boldsymbol{W}_i\}_{i=1}^T$ obtained from distinct tasks, model merging seeks a merged parameter set $\widetilde{\boldsymbol{W}}$ that approximates the behavior of each $\boldsymbol{W}_i$ on its corresponding task.

Task Vectors. For a FFT model of the i-th task, the task vector is defined as $\Delta\boldsymbol{W}_i = \boldsymbol{W}_i - \boldsymbol{W}_0$, which captures the direction and magnitude of adaptation in the weight space [29]. In the LoRA paradigm, the task-specific update is parameterized explicitly as $\Delta\boldsymbol{W}_i^{\text{LoRA}} = \boldsymbol{B}_i\boldsymbol{A}_i$, where $\boldsymbol{A}_i \in \mathbb{R}^{r\times d}$ and $\boldsymbol{B}_i \in \mathbb{R}^{d\times r}$. Thus, LoRA directly produces a compact and structured low-rank task vector.

Unified View and Low-Rank Merging. Although FFT and LoRA differ in parameterization, they are inherently connected under a unified low-rank formulation. Empirically, FFT updates $\Delta\boldsymbol{W}_i$ tend to reside in a low-dimensional subspace [26] and can be well approximated by a truncated SVD as $\Delta\boldsymbol{W}_i \approx \boldsymbol{U}_i\boldsymbol{\Sigma}_i\boldsymbol{V}_i^\top$. From this perspective, LoRA explicitly constrains the parameter updates to a low-rank subspace, whereas FFT implicitly exhibits a similar low-rank structure that can be revealed through singular value analysis and compression. This insight bridges the two approaches under a unified low-rank adaptation paradigm.

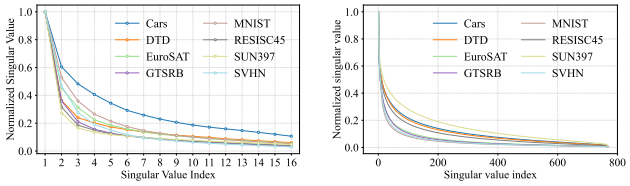
Building on this unified view, model merging can be viewed as an operation on these low-rank matrices. The merging process thus involves: i) extracting low-rank representations for each task vector, ii) merging these low-rank matrices to effectively integrate multi-task knowledge, and iii) constructing the final model by combining the pre-trained weights and merged multi-task vector through $\widetilde{\boldsymbol{W}} = \boldsymbol{W}_0 + \Delta\widetilde{\boldsymbol{W}}$.

2.2. Balanced Energy Enhances Generalization

A New Perspective on Task Vectors. We first provide a new interpretation of task vectors from the viewpoint of their intrinsic low-rank structure. Given a task vector $\Delta\boldsymbol{W}$ with rank $r \ll d$, we can decompose it using SVD:

$$\Delta\boldsymbol{W} = \sum_{i=1}^{r} \sigma^i \boldsymbol{u}^i \boldsymbol{v}^{i\top}, \tag{1}$$

where $\{\boldsymbol{u}^i\}_{i=1}^r$ and $\{\boldsymbol{v}^i\}_{i=1}^r$ are the left and right singular vectors, respectively, and $\{\sigma_i\}_{i=1}^r$ are the correspond-



(a) LoRA (b) FFT

Figure 1. The singular value distribution of task vectors averaged across all layers. We normalize each singular value by the largest one within each dataset to eliminate the magnitude discrepancy among different datasets.

ing singular values. From this decomposition, each term $\sigma^i \boldsymbol{u}^i \boldsymbol{v}^{i\top}$ can be regarded as a specific *knowledge vector* for task adaptation, and the adaptation direction $\boldsymbol{u}^i \boldsymbol{v}^{i\top}$ is defined as a *knowledge component*. The singular value σ^i quantifies the degree to which the corresponding knowledge component is utilized, thus the distribution of $\boldsymbol{\sigma} = (\sigma^1, \ldots, \sigma^r)$ can be interpreted as the *energy distribution* across knowledge components.

Observation: Singular Values Follow a Long-Tailed Distribution. An important empirical observation arises when analyzing the low-rank task vectors obtained from practical training. As shown in Figure 1, the decomposed knowledge vectors $\{\sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\top\}_{i=1}^r$ usually follow a long-tailed energy distribution, where a small fraction of knowledge components dominate the total energy, indicating that the knowledge captured by the task vector is inherently imbalanced. This intrinsic imbalance leads to a potential drawback in model behavior: as a small number of knowledge components dominate the task adaptation, the model tends to overfit to specific patterns while neglecting weaker but semantically important components.

To further illustrate this phenomenon, Figure 2a presents the transfer performance in eight tasks, contrasting the results before and after balancing the internal knowledge of each task vector. The *diagonal elements* of each heatmap represent the degree to which the original task's capability is preserved, while the *off-diagonal elements* measure the zero-shot transferability to other tasks, reflecting how well the knowledge generalizes beyond its training domain. As shown in Figure 2a (left), directly using the original low-rank task vector obtained from fine-tuning causes severe performance degradation on unrelated tasks, with many off-diagonal entries significantly suppressed, implying that over-concentrated energy distribution harms cross-task generalization. In contrast, Figure 2a (right) displays the results of *energy-balanced task vectors* constructed by a simple averaging strategy:

$$\Delta \overline{\boldsymbol{W}} = \left(\frac{\sum_{i=1}^r \sigma^i}{r} \right) \left(\sum_{i=1}^r \boldsymbol{u}^i \boldsymbol{v}^{i\top} \right). \tag{2}$$

The diagonal elements are quite close to 1.0, indicating that the vast majority of task capability is preserved, and the off-diagonal elements increase notably, suggesting improved zero-shot transfer and multi-task compatibility.

Revisiting Task Vector Similarity from a Knowledge Decomposition Perspective. To further analyze the underlying reason behind better generalization capabilities after balancing the energy distribution, we revisit the cosine similarity between task vectors [29] from the perspective of knowledge decomposition. We argue that this metric can be interpreted as the *expressive capacity* of one task vector to represent another, i.e., how well the knowledge of task t can be linearly reconstructed by task s.

Proposition 1 *Given the knowledge vector decompositions of two task vectors $\Delta \boldsymbol{W}_s$ and $\Delta \boldsymbol{W}_t$, their cosine similarity can be equivalently expressed as*

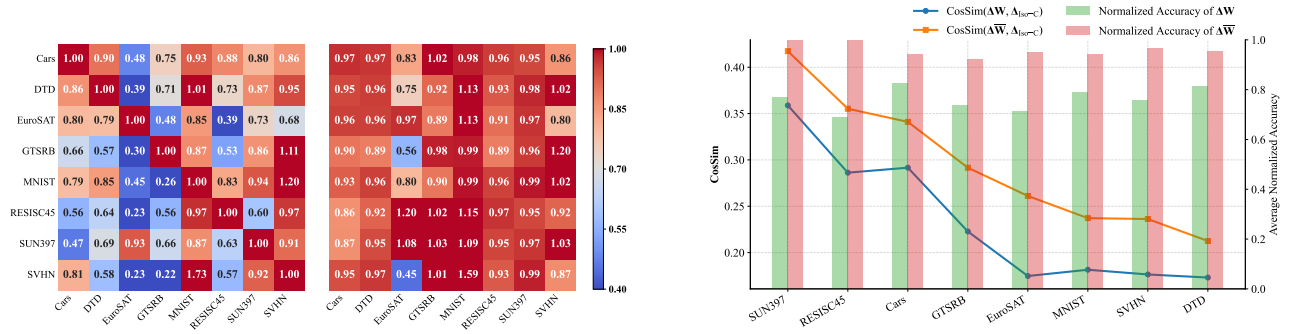
$$\begin{aligned} \mathrm{CosSim}(\Delta \boldsymbol{W}_s, \Delta \boldsymbol{W}_t) &= \frac{\langle \Delta \boldsymbol{W}_s, \Delta \boldsymbol{W}_t \rangle}{\|\Delta \boldsymbol{W}_s\|_F \, \|\Delta \boldsymbol{W}_t\|_F} \\ &= \frac{\boldsymbol{\sigma}_s \, \boldsymbol{R}(s,t) \, (\boldsymbol{\sigma}_t)^\top}{\|\boldsymbol{\sigma}_s\|_2 \, \|\boldsymbol{\sigma}_t\|_2}, \end{aligned} \tag{3}$$

where $\boldsymbol{R}(s,t) \in \mathbb{R}^{n \times m}$ is defined entry-wise as:

$$\boldsymbol{R}_{i,j}(s,t) = \left(\boldsymbol{u}_s^i\right)^\top \boldsymbol{u}_t^j \, \left(\boldsymbol{v}_t^j\right)^\top \boldsymbol{v}_s^i. \tag{4}$$

Remark. The matrix $\boldsymbol{R}(s,t)$ measures the directional consistency between the two knowledge bases. Each entry $\boldsymbol{R}_{i,j}(s,t) = (\boldsymbol{u}_s^i)^\top \boldsymbol{u}_t^j \, (\boldsymbol{v}_t^j)^\top \boldsymbol{v}_s^i$ quantifies how the j-th knowledge component of task t can be projected onto the i-th knowledge component of task s. Therefore, $\boldsymbol{R}(s,t)$ can be interpreted as a projection operator that expresses the knowledge geometry of task t in the basis of task s. From this perspective, the overall cosine similarity $\mathrm{CosSim}(\Delta \boldsymbol{W}_s, \Delta \boldsymbol{W}_t) = \frac{\boldsymbol{\sigma}_s \boldsymbol{R}(s,t) (\boldsymbol{\sigma}_t)^\top}{\|\boldsymbol{\sigma}_s\|_2 \|\boldsymbol{\sigma}_t\|_2}$ can be viewed as the weighted aggregation of these projections, reflecting how effectively task s can represent or reconstruct the knowledge of task t. When the rank m of $\Delta \boldsymbol{W}_t$ is higher than the rank n of $\Delta \boldsymbol{W}_s$, this interpretation is particularly intuitive: $\Delta \boldsymbol{W}_s$ spans a lower-dimensional subspace that attempts to encode the richer knowledge geometry of task t. Thus, a higher CosSim indicates stronger expressiveness of task s with respect to task t. We leave further analysis of this perspective in Appendix C.

Energy-Balanced Knowledge Components Enhance Multi-Task Performance. Building upon the cosine similarity analysis above, we can now explain why energy-balanced task vectors achieve stronger multi-task capability. When singular values $\boldsymbol{\sigma}$ are highly skewed, the corresponding task vectors collapse onto a few dominant knowledge components, limiting the span of the subspace and thereby reducing its expressive coverage over other tasks. In contrast, balancing the energy distribution across knowledge components prevents representational collapse and enlarges



(a) (b)

Figure 2. **(a)** Comparison of cross-task transferability before (left) and after (right) balancing the energy distribution. The diagonal elements represent the relative performance of each task with respect to its fine-tuned model, while the off-diagonal elements indicate the relative transfer performance on other tasks, normalized against their zero-shot baseline. **(b)** Each task vector's cosine similarity with multi-task vector (lines) and average normalized transfer accuracy of each task vector (bars). We compare original task vectors and their energy-balanced counterparts. The energy-balanced task vectors achieve higher CosSim with the multi-task vector (we use $\mathbf{\Delta}_{\text{Iso-C}}$ [46] for simplicity) and better cross-task generalization, indicating that balancing the energy distribution across knowledge components enhances multi-task expressiveness.

the subspace, thereby enhancing its ability to encode multiple tasks.

To verify this, we compute the cosine similarity between the single-task vectors and multi-task vector, and compare it with that of the energy-balanced task vectors obtained by Eq. (2). As shown in Figure 2b, the balanced vectors consistently exhibit higher cosine similarity with the multi-task model vector. Empirically, this aligns with higher average normalized transfer accuracy. These results suggest that energy smoothing improves the task vector's ability to represent multi-task knowledge.

2.3. Measuring Directional Consistency

Directional Knowledge Similarity. We now delve into the underlying mechanism of task knowledge preservation. The cosine similarity can be factorized into two components: i) matrix $\boldsymbol{R}(s,t)$ quantifies the directional consistency between the two knowledge components, and ii) singular values $\boldsymbol{\sigma}_s, \boldsymbol{\sigma}_t$ encode the importance of these knowledge components. While both terms contribute to the overall similarity, we argue that the preservation of task ability primarily depends on the *directional consistency* rather than on the energy distribution. Empirically, this observation is supported by the fact that energy-balanced task vectors can largely maintain the performance of the original task vector, which implies that as long as the relative directions of knowledge vectors are preserved, the model can retain most of its learned behavior.

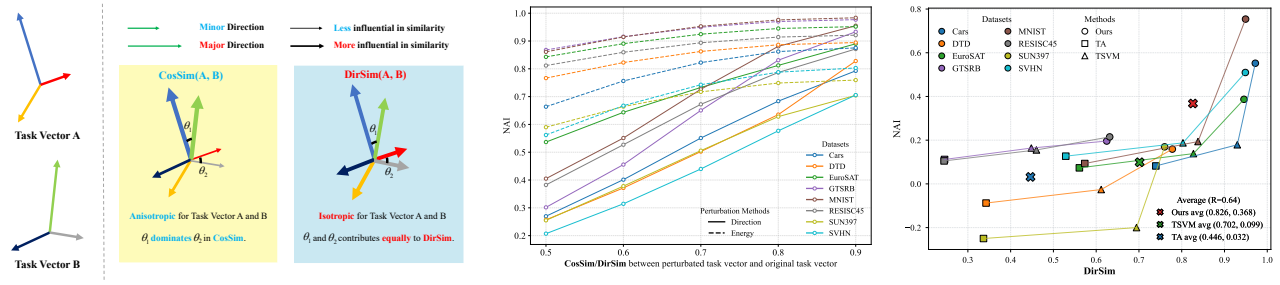
To isolate this directional factor and quantify it explicitly, we propose a new similarity metric that removes the influence of energy distribution by uniformizing it with $\bar{\boldsymbol{\sigma}}_s = \frac{\mathbf{1}_n}{\sqrt{n}}, \bar{\boldsymbol{\sigma}}_t = \frac{\mathbf{1}_m}{\sqrt{m}}$. Substituting back to the cosine similarity leads to a purely directional consistency measure:

$$\begin{aligned} \text{DirSim}(\Delta \boldsymbol{W}_s, \Delta \boldsymbol{W}_t) &\triangleq \bar{\boldsymbol{\sigma}}_s \, \boldsymbol{R}(s,t) \, \left(\bar{\boldsymbol{\sigma}}_t\right)^\top \\ &= \frac{1}{\sqrt{nm}} \, \mathbf{1}_n^\top \, \boldsymbol{R}(s,t) \, \mathbf{1}_m. \end{aligned} \tag{5}$$

DirSim equally considers the similarity between every pair of directions, whereas CosSim is subject to the similarity among the dominant directions (Figure 3a). A higher value DirSim implies that the two task vectors share more directionally consistent knowledge components and that one can better represent the other within its knowledge basis.

To verify the above claim, we conduct controlled perturbation experiments and analyze how the retained task performance correlates with both DirSim and CosSim. As illustrated in Figure 3b, performance decreases as DirSim declines, indicating that directional inconsistency leads to substantial performance loss. In contrast, when only energy is redistributed while the directions remain aligned, performance remains largely stable despite notable changes in CosSim. These results provide strong empirical evidence that preserving knowledge directions is the key to maintaining task ability, while variation of energy distribution has a relatively minor effect. We leave implementation details in Appendix E.3.

Directional Similarity for Post-merge Task-wise Performance. Building upon the previous analysis, we argue that preserving the directions of knowledge components is the key factor for retaining each task's performance during model merging. To quantify this, one might consider computing the directional similarity (DirSim) between each task vector $\Delta \boldsymbol{W}_i$ and the merged task vector $\Delta \widetilde{\boldsymbol{W}}$. However, $\Delta \widetilde{\boldsymbol{W}}$ aggregates multiple task vectors, introducing directional redundancy that artificially deflates DirSim.



(a) (b) (c)

Figure 3. **(a)** Comparison of CosSim and DirSim. DirSim considers the similarity between every pair of directions equally, whereas CosSim mainly focuses on the similarity among the dominant directions while ignoring the minor ones. **(b)** Empirical validation of the importance of preserving directional geometry. Solid line: task performance vs. DirSim under random directional perturbations; dashed line: task performance vs. CosSim under energy distribution perturbations. **(c)** Correlation of task-wise performance with projected DirSim. The overall correlation is positive (Pearson $R = 0.64$) and per-method averages follow the same trend. Similar patterns persist under larger task scales as illustrated in Figure 7. We utilize Normalized Accuracy Improvement (NAI) [46] to measure task-wise performance in **(b)** and **(c)**.

To address this, we consider the task-specific activation of the merged vector by projecting it onto the low-rank feature subspace. Empirical evidence shows that features from a given task i collapse into a low-rank subspace $\boldsymbol{U}_i$ [13, 19]. We thus project the merged multi-task vector onto the low-rank feature subspace to acquire the task-activated part:

$$\Delta\widetilde{\boldsymbol{W}}_i \;=\; \boldsymbol{U}_i\boldsymbol{U}_i^{\top}\,\Delta\widetilde{\boldsymbol{W}}. \tag{6}$$

We then compute $\mathrm{DirSim}(\Delta\boldsymbol{W}_i, \Delta\widetilde{\boldsymbol{W}}_i)$ as our metric for task-wise capability retention.

In practice, the exact low-rank feature subspace $\boldsymbol{U}_i$ is inaccessible and we approximate it using the subspace spanned by the left singular vectors of $\Delta\boldsymbol{W}_i$. As shown in Figure 3c, the projected DirSim computed in this manner exhibits a clear monotonic relationship with the normalized accuracy improvement of merged models across different datasets and methods. Points with higher projected DirSim correspond to better task knowledge retention. This confirms that the proposed projected DirSim can measure how well task-specific knowledge is preserved during model merging.

3. The Proposed DC-Merge Approach

The goal of our method is to preserve the complete directional geometry of task vectors during model merging. To this end, we propose two modules, i.e., *energy smoothing* and *cover space merging*.

Energy Smoothing for Balanced Knowledge Representations. As discussed in Section 2.2, the imbalanced energy distribution of knowledge components biases the merging process, potentially ignoring the direction of weaker but semantically rich knowledge components. To mitigate these issues, we balance the energy distribution of each task vector via energy smoothing before merging. For each task vector $\boldsymbol{\Delta}_i$, we consider its knowledge decomposition $\boldsymbol{\Delta}_i = \sum_{j=1}^{r} \sigma_i^j \boldsymbol{u}_i^j \boldsymbol{v}_i^{j\top}$, where $\boldsymbol{\sigma}_i = (\sigma_i^1, \ldots, \sigma_i^r)$ contains the singular values sorted in descending order. Instead of directly using the original energy distribution $\boldsymbol{\sigma}_i$, we replace it with a smoothed version $\overline{\boldsymbol{\sigma}}_i$ to redistribute the energy more evenly across the top-r components, alleviating dominance on a few knowledge components.

For example, we consider the simple yet effective form of smoothing by replacing all top-r singular values with their mean: $\overline{\boldsymbol{\sigma}}_i = \left(\frac{1}{r}\sum_{j=1}^{r}\sigma_i^j\right)\mathbf{1}_r$, which equalizes the contribution of all retained knowledge components. For completeness, additional smoothing strategies are discussed in Appendix E.4. We then perform merging on these energy-balanced task vectors rather than the original ones.

Algorithm 1 DC-Merge

Input: Task vectors $\{\boldsymbol{\Delta}_i\}_{i=1}^{T}$ with $\boldsymbol{\Delta}_i \in \mathbb{R}^{m\times n}$
Output: Merged multi-task vector $\widetilde{\boldsymbol{\Delta}}$
▷ **Step 1:** Construct cover space for all task vectors
for $i = 1 \to T$ **do**
  Compute r-rank SVD: $\boldsymbol{\Delta}_i \approx \boldsymbol{U}_i^{(r)}\boldsymbol{\Sigma}_i^{(r)}\boldsymbol{V}_i^{(r)\top}$
  Smoothing $\boldsymbol{\Sigma}_i^{(r)}$ by $\overline{\boldsymbol{\Sigma}}_i^{(r)} = \mathrm{diag}([\overline{\sigma}_i^1; \overline{\sigma}_i^2; \ldots; \overline{\sigma}_i^r])$
  Reconstruct $\overline{\boldsymbol{\Delta}}_i = \boldsymbol{U}_i^{(r)}\overline{\boldsymbol{\Sigma}}_i^{(r)}\boldsymbol{V}_i^{(r)\top}$
end for
Obtain concatenated basis $\boldsymbol{U}, \boldsymbol{V}$ by Eq. (9)
Whitening $\boldsymbol{U}$ and $\boldsymbol{V}$ respectively to obtain $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$
▷ **Step 2:** Project $\overline{\boldsymbol{\Delta}}_i$ onto cover space and merge
Project each $\overline{\boldsymbol{\Delta}}_i$ and obtain $\boldsymbol{M}_i$ by Eq. (10)
$\widetilde{\boldsymbol{M}} \leftarrow \mathrm{Merging}(\{\boldsymbol{M}_i\}_{i=1}^{T})$ via TA or TIES
▷ **Step 3:** Project $\widetilde{\boldsymbol{M}}$ back to parameter space
Construct mask $\boldsymbol{\mathcal{M}} \leftarrow \mathrm{block\text{-}diag}(\mathbf{1}_{r\times r}, \cdots, \mathbf{1}_{r\times r})$
Obtain merged task vector $\widetilde{\boldsymbol{\Delta}}$ by Eq. (11)

Projection and Merging in the Cover Space. Directly merging task vectors in the original parameter space may distort directional geometry due to misaligned subspaces $\{(\boldsymbol{U}_i, \boldsymbol{V}_i)\}$. To preserve directional consistency across all tasks, we seek a pair of shared orthonormal basis $(\widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}})$ that define a *cover space* capturing the directional geometry of all task vectors, and perform merging within the cover space. The objective can be formulated as:

$$\min_{\widetilde{\boldsymbol{U}},\widetilde{\boldsymbol{V}}} \sum_{i=1}^{T}\sum_{j=1}^{r} \min_{\boldsymbol{\sigma}_i^j \in \mathbb{R}^k} \left\| \boldsymbol{u}_i^j \boldsymbol{v}_i^{j\top} - \widetilde{\boldsymbol{U}}\, \mathrm{diag}(\boldsymbol{\sigma}_i^j)\, \widetilde{\boldsymbol{V}}^\top \right\|_F^2, \quad \text{s.t. } \widetilde{\boldsymbol{U}}^\top \widetilde{\boldsymbol{U}} = \widetilde{\boldsymbol{V}}^\top \widetilde{\boldsymbol{V}} = \boldsymbol{I}. \tag{7}$$

where $k = rT$. The inner minimization over $\boldsymbol{\sigma}_i^j$ determines the optimal coefficients along the cover basis. Using Proposition 2(a), we obtain the surrogate objective:

$$\max_{\widetilde{\boldsymbol{U}},\widetilde{\boldsymbol{V}}} \sum_{i=1}^{T}\sum_{j=1}^{r} \left\| \boldsymbol{\sigma}(\widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}}, \boldsymbol{u}_i^j \boldsymbol{v}_i^{j\top}) \right\|_2^2, \quad \text{s.t. } \widetilde{\boldsymbol{U}}^\top \widetilde{\boldsymbol{U}} = \widetilde{\boldsymbol{V}}^\top \widetilde{\boldsymbol{V}} = \boldsymbol{I}, \tag{8}$$

where $\boldsymbol{\sigma}(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\Delta}) = \mathrm{diag}\left(\boldsymbol{U}^\top \boldsymbol{\Delta} \boldsymbol{V}\right) \in \mathbb{R}^k$ denotes projection onto the shared dyadic directions.

As directly optimizing Eq. (8) incurs non-trivial computational overhead, we adopt whitening [54] here as it serves as a near-optimal solution to Eq. (8) while being computationally efficient. In Appendix D, we provide an iterative approach for constructing cover basis and theoretically show its relation to the whitening transformation. Specifically, we construct the cover basis $(\widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}})$ by whitening the column-wise concatenated per-task knowledge basis:

$$\boldsymbol{U} = [\boldsymbol{U}_1^{(r)}, \ldots, \boldsymbol{U}_T^{(r)}], \qquad \boldsymbol{V} = [\boldsymbol{V}_1^{(r)}, \ldots, \boldsymbol{V}_T^{(r)}]. \tag{9}$$

Thus, $\widetilde{\boldsymbol{U}}^\top \widetilde{\boldsymbol{U}} = \widetilde{\boldsymbol{V}}^\top \widetilde{\boldsymbol{V}} = \boldsymbol{I}$ defines an orthogonal basis that contains the union of all tasks' directional geometry. Each smoothed task vector is then projected onto cover space by:

$$\boldsymbol{M}_i = \widetilde{\boldsymbol{U}}^\top \overline{\boldsymbol{\Delta}}_i \widetilde{\boldsymbol{V}}. \tag{10}$$

This projection ensures that all task vectors are expressed under shared cover basis, which facilitates directionally consistent task vectors aggregation via existing element-wise merging methods, such as TA [29] and TIES-Merging [68], to obtain $\widetilde{\boldsymbol{M}}$. Finally, the merged multi-task vector is reconstructed by projecting $\widetilde{\boldsymbol{M}}$ back to the original parameter space:

$$\widetilde{\boldsymbol{\Delta}} = \widetilde{\boldsymbol{U}} \left(\widetilde{\boldsymbol{M}} \odot \boldsymbol{\mathcal{M}}\right) \widetilde{\boldsymbol{V}}^\top, \tag{11}$$

where $\boldsymbol{\mathcal{M}}$ serves as a structural mask. We leave further discussion of the structural mask in Appendix D and summarize the key steps of our approach in Algorithm 1.

4. Experiments

In this section, we evaluate the performance of DC-Merge against existing baselines through extensive experiments using vision models and vision-language models (VLMs) in both FFT and LoRA settings, demonstrating the versatility of DC-Merge. We further perform comprehensive ablation studies to analyze the effectiveness of each key component in DC-Merge.

4.1. Results for Vision Tasks

In this subsection, we investigate the merging of vision models. For fully fine-tuned vision models, following prior works [20, 46], we use 8-task, 14-task, and 20-task benchmarks for evaluation, respectively, and employ three CLIP [52] variants: `ViT-B-32`, `ViT-B-16`, and `ViT-L-14` as visual encoders [15]. For LoRA fine-tuned vision models, we extend previous evaluations by assessing both our method and existing baselines on a larger number of tasks, specifically 8, 12 and 16. Consistent with prior work [6, 20, 46, 58], we report the average absolute and normalized accuracy of merged models.

In the full parameter fine-tuning setting, we compare our method against Weight Averaging [64], Task Arithmetic [29], TIES-Merging [68], Consensus TA [62], TSV-M [20] and Iso-CTS [46]. For LoRA fine-tuned models, Task Arithmetic, KnOTS-TIES [58], WUDI-Merging [6], TSV-M, and Iso-CTS serve as baselines. We provide details on benchmarks and experimental setups in the Appendix E.

Table 1 shows the results of merging vision models fine-tuned by LoRA. Across three different backbones, the performance of our method consistently surpasses the current state-of-the-art methods. Moreover, the performance gains remain substantial with the growth of tasks. We also conduct experiments on the checkpoints provided by KnOTS [58], where our method still achieves superior performance compared to existing state-of-the-art methods. The corresponding results are reported in Appendix F.1. Table 2 presents the results under the full fine-tuning setting. The results demonstrate that our approach not only exhibits strong capability when merging LoRA fine-tuned models but also achieves state-of-the-art performance under the FFT scenario. Notably, the superiority of our method becomes more significant as the number of tasks increases.

4.2. Results for Vision-Language Tasks

In the multi-modal model merging setting, we compare our method with Task Arithmetic, TIES-Merging, DARE [69], PCB-Merging [16], and RobustMerge [70] on eight multi-modal datasets using `LLaVA-v1.5-7B` [38] as backbone. Following the experimental setup of RobustMerge, we further evaluate the merged model on four additional datasets to assess its generalization ability to unseen tasks. We adopt

Method	ViT-B-32			ViT-B-16			ViT-L-14		
	8 tasks	12 tasks	16 tasks	8 tasks	12 tasks	16 tasks	8 tasks	12 tasks	16 tasks
Individual	87.82	88.90	87.50	89.71	90.76	89.11	92.36	93.57	92.11
Task Arithmetic	52.80$_{(61.73)}$	60.76$_{(69.12)}$	60.04$_{(68.94)}$	57.70$_{(65.30)}$	64.26$_{(71.12)}$	62.40$_{(70.16)}$	68.29$_{(74.38)}$	73.69$_{(78.84)}$	69.98$_{(75.58)}$
KnOTS-TIES	55.93$_{(65.03)}$	63.03$_{(71.43)}$	61.78$_{(70.78)}$	60.80$_{(68.59)}$	66.35$_{(73.38)}$	64.31$_{(72.29)}$	73.61$_{(79.92)}$	75.64$_{(80.90)}$	72.19$_{(77.98)}$
WUDI-Merging	55.25$_{(64.38)}$	62.20$_{(70.64)}$	61.24$_{(70.27)}$	58.95$_{(66.63)}$	65.29$_{(72.29)}$	64.59$_{(72.51)}$	69.78$_{(75.91)}$	74.25$_{(79.45)}$	71.79$_{(77.59)}$
TSV-M	58.91$_{(68.25)}$	65.30$_{(73.86)}$	63.51$_{(72.72)}$	62.97$_{(70.87)}$	68.92$_{(76.06)}$	67.21$_{(75.35)}$	76.52$_{(83.00)}$	79.67$_{(85.13)}$	74.37$_{(80.31)}$
Iso-CTS	63.01$_{(72.71)}$	66.28$_{(75.01)}$	64.61$_{(74.02)}$	69.06$_{(77.38)}$	71.52$_{(78.87)}$	69.88$_{(78.22)}$	81.64$_{(88.31)}$	81.35$_{(86.87)}$	77.50$_{(83.65)}$
DC-Merge	**64.17$_{(73.90)}$**	**68.40$_{(77.22)}$**	**66.27$_{(75.80)}$**	**70.53$_{(78.86)}$**	**73.12$_{(80.56)}$**	**70.57$_{(78.91)}$**	**82.61$_{(89.42)}$**	**83.62$_{(89.31)}$**	**79.53$_{(85.71)}$**

Table 1. Average absolute accuracy results on vision model merging benchmarks in LoRA setting; subscript (in parentheses) is the average normalized accuracy. The best results are in **bold** and the second-best are underlined.

Method	ViT-B-32			ViT-B-16			ViT-L-14		
	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks
Individual	92.83	90.88	91.37	94.64	92.76	93.17	95.81	94.29	94.73
Weight Averaging	66.34$_{(72.13)}$	64.34$_{(71.12)}$	61.04$_{(67.53)}$	72.22$_{(76.60)}$	69.46$_{(74.82)}$	65.31$_{(70.36)}$	79.56$_{(83.15)}$	76.73$_{(81.10)}$	71.60$_{(75.60)}$
Task Arithmetic	70.79$_{(76.55)}$	65.32$_{(72.09)}$	60.52$_{(66.79)}$	75.41$_{(79.58)}$	70.52$_{(75.89)}$	65.78$_{(70.76)}$	84.93$_{(88.65)}$	79.41$_{(83.95)}$	74.01$_{(78.07)}$
TIES-Merging	75.09$_{(81.08)}$	68.02$_{(74.83)}$	63.38$_{(69.90)}$	79.74$_{(84.34)}$	73.22$_{(78.73)}$	68.18$_{(73.26)}$	86.88$_{(90.69)}$	79.46$_{(84.05)}$	75.71$_{(79.80)}$
Consensus TA	75.03$_{(80.84)}$	70.39$_{(77.36)}$	65.43$_{(71.98)}$	79.39$_{(83.86)}$	74.39$_{(79.92)}$	69.76$_{(74.93)}$	86.34$_{(90.08)}$	82.22$_{(86.94)}$	79.00$_{(83.22)}$
TSV-M	85.86$_{(92.31)}$	80.06$_{(87.88)}$	77.07$_{(84.29)}$	89.01$_{(93.94)}$	84.58$_{(91.01)}$	80.57$_{(86.45)}$	92.98$_{(96.98)}$	89.17$_{(94.43)}$	87.72$_{(92.50)}$
Iso-CTS	86.20$_{(91.78)}$	81.71$_{(89.70)}$	78.05$_{(85.48)}$	**90.91$_{(95.95)}$**	86.40$_{(92.81)}$	82.38$_{(88.36)}$	**94.69$_{(98.81)}$**	90.98$_{(96.28)}$	90.05$_{(94.88)}$
DC-Merge	**87.05$_{(93.55)}$**	**82.52$_{(90.62)}$**	**80.58$_{(88.18)}$**	90.78$_{(95.83)}$	**87.06$_{(93.70)}$**	**84.57$_{(90.76)}$**	94.31$_{(98.38)}$	**91.01$_{(96.43)}$**	**90.51$_{(95.43)}$**

Table 2. Average absolute accuracy results on vision model merging benchmarks in FFT setting; subscript (in parentheses) is the average normalized accuracy. The best results are in **bold** and the second-best are underlined.

the checkpoints released by RobustMerge and provide detailed experimental configurations in Appendix E.

As shown in Table 3, our method notably outperforms existing state-of-the-art methods on both seen and unseen tasks, demonstrating that its applicability is not limited to vision models but can also scale to large multi-modal models. We also evaluate the generalization capability to unseen tasks of our method on vision models and the detailed results are presented in Appendix F.3.

Method	Seen Tasks	Unseen Tasks
Zeroshot	43.37	25.22
Individual	69.23	–
Multi Task	63.62	36.06
Task Arithmetic	53.93	33.31
DARE-Merging	53.84	33.15
TIES-Merging	53.09	33.14
PCB-Merging	53.70	33.53
RobustMerge	57.33	37.99
DC-Merge	**59.63**	**39.84**

Table 3. Performance on MM-MergeBench [70], containing eight seen tasks (LoRA fine-tuned) and four unseen tasks. The best results are in **bold** and the second-best are underlined. We report average absolute accuracy. See Appendix F.7 for detailed results.

4.3. Ablations and Analysis

Unless otherwise specified, all experiments in this subsection are conducted in LoRA setting.

The Effectiveness of Energy Smoothing. We investigate the impact of our energy smoothing strategy on the performance of our method, with the results summarized in Table 4. The results align well with our observations: applying energy smoothing to each task vector effectively ensures that all the knowledge components can be adequately expressed, leading to a significant improvement in overall performance. Notably, preserving a moderate degree of skewness in the energy distribution (i.e., linear smoothing) can yield better results than averaging. We provide additional comparisons of smoothing strategy on ViT-B-16 and ViT-L-14 in Appendix E.4.

Method	8 tasks	12 tasks	16 tasks
No smoothing	69.27	74.60	74.47
Averaging	73.09 (+3.82)	76.42 (+1.82)	75.51 (+1.04)
Linear smoothing	73.90 (+4.63)	77.22 (+2.62)	75.80 (+1.33)

Table 4. Performance comparison of different smoothing strategies. We report average normalized accuracy using ViT-B-32.

Impact of Performing a Post-hoc Pruning. In Algorithm 1, we perform a post-hoc pruning by applying a mask $\mathcal{M}$ to mitigate directional inconsistency of different tasks before projecting the merged parameter matrix $\widetilde{M}$ back to the original parameter space. Table 5 presents the effect of such structural pruning on overall performance in both LoRA and FFT settings. The performance degradation becomes more pronounced with the increase of tasks. Moreover, since the number of fine-tuned parameters in the FFT setting is substantially larger than that in LoRA, incorporating masks leads to significant performance gains of up to 10.55% in average normalized accuracy, highlighting the crucial role

that structural pruning plays in preventing cross-task directional inconsistency. We investigate the impact of mask size on the performance in Appendix D.

Method	Tasks	w/o mask	w/ mask
FFT	8 tasks	87.98	93.55 (+5.57)
	14 tasks	82.39	90.50 (+8.11)
	20 tasks	77.63	88.18 (+10.55)
LoRA	8 tasks	73.61	73.90 (+0.29)
	12 tasks	75.94	77.22 (+1.28)
	16 tasks	74.42	75.80 (+1.38)

Table 5. Comparison of performance with and w/o applying masks. We report average normalized accuracy using `ViT-B-32`.

Impact of Merging in the Shared Cover space. To maintain the directional geometry of each task vector, we project the smoothed task vectors onto a shared subspace prior to model merging. As shown in Table 6, compared to merging in the original parameter space, CSM significantly boosts the performance of both TA [29] and TIES [68]. Moreover, after applying energy smoothing to the task vectors, the performance is further enhanced, indicating that the two main components of our proposed method are complementary. An illustrative example provided in Appendix D further demonstrates the importance of shared cover basis in preserving the directional structure of task vectors.

Method	8 tasks	12 tasks	16 tasks
Vanilla TA	61.73	69.12	68.94
TA + ES	69.12 (+7.39)	74.35 (+5.23)	73.01 (+4.07)
TA + CSM	68.13 (+6.40)	73.92 (+4.80)	72.64 (+3.70)
TA + CSM + ES	73.82 (+12.09)	77.16 (+8.04)	75.73 (+6.79)
Vanilla TIES	62.09	69.30	70.06
TIES + ES	69.94 (+7.85)	74.97 (+5.67)	74.74 (+4.68)
TIES + CSM	69.27 (+7.18)	74.60 (+5.30)	74.47 (+4.41)
TIES + CSM + ES	73.90 (+11.81)	77.22 (+7.92)	75.80 (+5.74)

Table 6. Performance of individually applying energy smoothing (ES) and cover space merging (CSM) as well as combining them to TA or TIES compared with vanilla settings. We report the average normalized accuracy using `ViT-B-32`.

5. Related Work

Model merging has emerged as a promising approach to integrate expert models fine-tuned on different downstream tasks into a single multi-task model. Task Arithmetic (TA) [29] first introduces the concept of a *task vector*, defined as the difference between an expert and its pre-trained model, and combines them through scaled averaging to construct a merged model. Subsequent studies propose meticulously crafted parameter-wise strategies to mitigate interference during merging. TIES [68] reduces sign conflicts by adopting the majority sign across all models. Consensus Merging [62] applies binary masks to exclude parameters important to fewer than two tasks. Recent studies WUDI-Merging [6] and FDA [55] optimizes the merged task vector to keep the output of merged model align with each fine-tuned model given the same input of corresponding task.

These merging methods are data-free, producing merged task vectors that can be directly integrated into the pre-trained model. A number of recent approaches, however, focus on creating model with multi-task capabilities by modifying the inference stage. Twin-Merging [42] composes task-specific components at test time, requiring two forward passes. EMR-Merging [27] employs additional per-task masks and rescalers for inference. In this paper, we restrict our study to merging methods which are data-free and leave the inference stage unaffected.

SVD-based Model Merging. Recent data-free model merging methods have incorporated SVD to improve performance [58, 63]. State-of-the-art approaches include TSV-M [20], which enforces orthogonality between task-specific subspaces to reduce task interference, and Iso-CTS [46], which standardizes singular values after combining a common subspace constructed by TA [29] and task-specific subspaces. More recently, ESM [36] projects parameter updates into an activation-aware essential subspace and applies polarized scaling to amplify critical weights. In contrast to these methods, our approach prioritizes the directional consistency of each original task vector with the merged vector. We achieve this by balancing the energy distribution of knowledge components and performing the merge process within a shared orthogonal subspace induced by a pair of cover basis.

6. Conclusion and Limitation

Conclusion. In this work, we are the first to identify that preserving the directional consistency of task vectors after merging is crucial for retaining the capabilities of individual tasks. Building upon this insight, we propose DC-Merge, which maintains the directional consistency between the merged multi-task vector and each original task vector by *energy smoothing* and *cover space merging*. Our method achieves state-of-the-art performance in both FFT and LoRA settings.

Limitation. There still exists a noticeable performance gap between merging LoRA fine-tuned models and full parameter fine-tuned models. This phenomenon may arise from the number of knowledge components in each task vector. A larger number of knowledge components provides redundancy that is robust to direction shift, whereas a smaller set makes the task vector more fragile to directional inconsistency. The knowledge components of each LoRA task vector are scarce, even fewer than the LoRA rank due to its long-tailed energy distribution. A potential remedy lies in promoting a balanced energy distribution of the knowledge components during fine-tuning.

Acknowledgements

This work was supported by the National Science Foundation of China (62576092, 62225602), the Basic Research Program of Jiangsu (BK20253021), and the Big Data Computing Center of Southeast University. We would like to thank anonymous reviewers for their constructive suggestions.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv: 1607.06450*, 2016. 18

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – Mining Discriminative Components with Random Forests. In *ECCV*, 2014. 16

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *ICCV*, 2021. 1

[5] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 2017. 16

[6] Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *ICML*, 2025. 1, 6, 8, 22

[7] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 16

[8] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep Learning for Classical Japanese Literature. *arXiv preprint arXiv: 1607.06450*, 2018. 16

[9] Adam Coates, Andrew Ng, and Honglak Lee. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2011. 16

[10] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: Extending MNIST to handwritten letters. In *IJCNN*, 2017. 16

[11] Nico Daheim, Thomas Möllenhoff, Edoardo M. Ponti, Iryna Gurevych, and Mohammad Emtiyaz Khan. Model merging by uncertainty-based gradient matching. In *ICLR*, 2024. 1

[12] Mohammad-Javad Davari and Eugene Belilovsky. Model breadcrumbs: Scaling multi-task model merging with sparse masks. In *ECCV*, 2024. 1

[13] Grégoire Delétang, Anian Ruoss, Paul-Ambroise Duquenne, Elliot Catt, Tim Genewein, Christopher Mattern, Jordi Grau-Moya, Li Kevin Wenliang, Matthew Aitchison, Laurent Orseau, Marcus Hutter, and Joel Veness. Language modeling is compression. *arXiv preprint arXiv:2309.10668*, 2024. 5

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 17

[15] 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. In *ICLR*, 2021. 2, 6

[16] Guodong Du, Junlin Lee, Jing Li, Runhua Jiang, Yifei Guo, Shuyang Yu, Hanting Liu, Sim K Goh, Ho-Kin Tang, Daojing He, et al. Parameter competition balancing for model merging. In *NeurIPS*, 2024. 6

[17] Ziqing Fan, Ruipeng Zhang, Jiangchao Yao, Bo Han, Ya Zhang, and Yanfeng Wang. Federated learning with bilateral curation for partially class-disjoint data. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 32006–32019, 2023. 1

[18] Ziqing Fan, Shengchao Hu, Jiangchao Yao, Gang Niu, Ya Zhang, Masashi Sugiyama, and Yanfeng Wang. Locally estimated global perturbations are better than local perturbations for federated sharpness-aware minimization. In *Proceedings of the 41st International Conference on Machine Learning*, pages 12858–12881, 2024. 1

[19] Giorgio Franceschelli, Claudia Cevenini, and Mirco Musolesi. Training foundation models as data compression: On information, model weights and copyright law. *arXiv preprint arXiv:2407.13493*, 2025. 5

[20] Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. In *CVPR*, 2025. 1, 6, 8, 18, 22

[21] Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Zhang Chuang, and Yoshua Bengio. Challenges in Representation Learning: A Report on Three Machine Learning Contests. *Neural Networks*, 2013. 16, 17

[22] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017. 17

[23] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *CVPR*, 2018. 17

[24] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019. 16

[25] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu,

Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021. 17

[26] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 2

[27] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. Emr-merging: Tuning-free high-performance model merging. In *NeurIPS*, 2024. 8

[28] Gabriel Ilharco, Mitchell Wortsman, Samir Yitzhak Gadre, Shuran Song, Hannaneh Hajishirzi, Simon Kornblith, Ali Farhadi, and Ludwig Schmidt. Patching open-vocabulary models by interpolating weights. In *NeurIPS*, 2022. 1

[29] Gabriel Ilharco, Marco Túlio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023. 1, 2, 3, 6, 8, 14, 20

[30] Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *ICLR*, 2023. 1

[31] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 17

[32] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D Object representations for fine-grained categorization. In *ICCV Workshops*, 2013. 16

[33] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *University of Toronto*, 2009. 16, 17

[34] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 16

[35] Mario Lezcano-Casado and David Martínez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. In *ICML*, 2019. 14

[36] Longhua Li, Lei Qi, Qi Tian, and Xin Geng. Model merging in the essential subspace. *arXiv preprint arXiv:2602.20208*, 2026. 8

[37] Weishi Li, Yong Peng, Miao Zhang, Liang Ding, Han Hu, and Li Shen. Deep model fusion: A survey. *arXiv preprint arXiv: 2309.15698*, 2023. 1

[38] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 6

[39] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021. 17

[40] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022. 17

[41] Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*, 2022. 17

[42] Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. Twin-merging: Dynamic integration of modular expertise in model merging. In *NeurIPS*, 2024. 8

[43] Mao-Lin Luo, Zi-Hao Zhou, Yi-Lin Zhang, Yuanyu Wan, Min-Ling Zhang, and Tong Wei. KeeploRA: Continual learning with residual gradient adaptation. *arXiv preprint arXiv:2601.19659*, 2026. 1

[44] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 17

[45] Daniel Marczak, Bartlomiej Twardowski, Tomasz Trzcinski, and Sebastian Cygert. MagMax: Leveraging Model Merging for Seamless Continual Learning. In *ECCV*, 2024. 1

[46] Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost van de Weijer. No Task Left Behind: Isotropic Model Merging with Common and Task-Specific Subspaces. In *ICML*, 2025. 4, 5, 6, 8, 18, 21, 22

[47] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *ICDAR*, 2019. 17

[48] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NeurIPS Workshops*, 2011. 16

[49] Maria-Elena Nilsback and Andrew Zisserman. Automated Flower Classification over a Large Number of Classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 2008. 16

[50] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012. 16

[51] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 17

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 6

[53] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *ECCV*, 2022. 17

[54] Peter H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 1966. 6, 14

[55] Kexuan Shi, Yandong Wen, and Weiyang Liu. Model merging with functional dual anchors. *arXiv preprint arXiv:2510.21223*, 2025. 8

[56] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, 2013. 16, 17

[57] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *IJCNN*, 2011. 16

[58] George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the Knots. In *ICLR*, 2025. 6, 8, 17, 20

[59] Vinita Vasudevan and M. Ramakrishna. A hierarchical singular value decomposition algorithm for low rank matrices. *arXiv preprint arXiv: 1710.02812*, 2017. 21

[60] Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation Equivariant CNNs for Digital Pathology. In *MICCAI*, 2018. 16

[61] Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. Screen2words: Automatic mobile ui summarization with multimodal learning. In *UIST*, 2021. 17

[62] Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *ICML*, 2024. 6, 8

[63] Yongxian Wei, Anke Tang, Li Shen, Chun Yuan, and Xiaochun Cao. Modeling multi-task model merging as adaptive projective gradient descent. *arXiv preprint arXiv:2501.01230*, 2025. 8

[64] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, 2022. 6

[65] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *CVPR*, 2022. 1

[66] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv: 1708.07747*, 2017. 16, 17

[67] Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *IJCV*, 2016. 16

[68] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In *NeurIPS*, 2023. 1, 6, 8, 18

[69] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *ICML*, 2024. 6

[70] Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Robustmerge: Parameter-efficient model merging for mllms with direction robustness. In *NeurIPS*, 2025. 6, 7, 17, 22

[71] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023. 1