This is the peer reviewd version of the followng article:

Transporting Task Vectors across Different Architectures without Training / Rinaldi, F., Panariello, A., Salici, G., Porrello, A., Calderara, S.. - (2026). (International Conference on Machine Learning Seoul, South Korea July 6th - 11th, 2026).

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

11/10/2026 13:43

(Article begins on next page)

Transporting Task Vectors across Different Architectures without Training

Filippo Rinaldi [1] **Aniello Panariello** [1] **Giacomo Salici** [1] **Angelo Porrello** [1] **Simone Calderara** [1]

Abstract

Adapting large pre-trained models to downstream tasks often produces task-specific parameter updates that are expensive to relearn for every model variant. While recent work has shown that such updates can be transferred between models with identical architectures, transferring them across models of different widths remains unexplored. In this work, we introduce **THESEUS**, a training-free method for transporting task updates across heterogeneous-width models. Rather than matching parameters, we characterize a task update by the functional effect it induces on intermediate representations. We formalize task-vector transport as a functional matching problem on observed activations and show that, after aligning representation spaces via orthogonal Procrustes analysis, it admits a stable closed-form solution that preserves the geometry of the update. We evaluate THESEUS on vision and language models across different widths, showing consistent improvements over baselines without additional training or backpropagation. Our results show that task updates can be meaningfully transferred across architectures when task identity is defined functionally rather than parametrically. Code is available at https://github.com/apanariello4/merge-and-rebase.

1. Introduction

Large pre-trained models have become a cornerstone of modern machine learning, demonstrating remarkable performance across a wide range of tasks in computer vision (Dosovitskiy et al., 2021; He et al., 2022) and natural language processing (Brown et al., 2020; Chowdhery et al., 2023). A key factor contributing to their success is the ability to adapt these pre-trained models to downstream tasks through fine-tuning or other lightweight techniques (Houlsby et al., 2019; Liu et al., 2022). At the same time, practitioners often maintain families of models that share architectural similarities but differ in size, training data, or other characteristics, raising a natural question: *can a task-specific update learned by one model be transferred directly to a different one without retraining?*

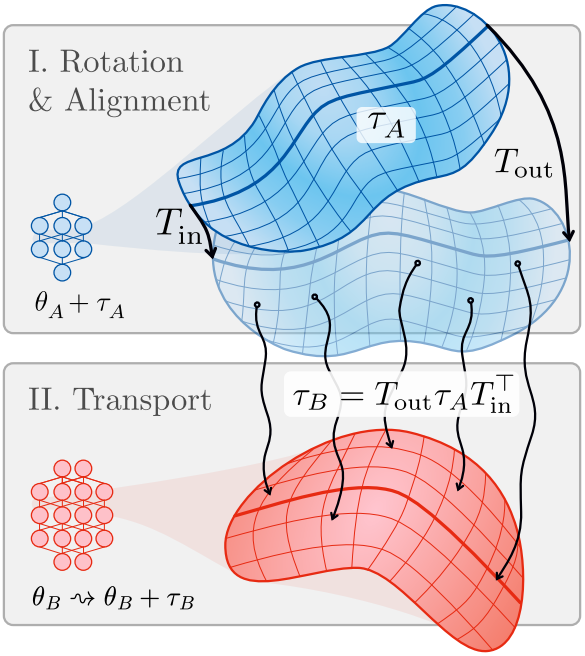

Figure 1. **Task-vector transport via orthogonal alignment.** Procrustes rotations identify shared structure between representation spaces, enabling training-free transfer of task updates across models with mismatched dimensions.

Existing work on model transfer and rebasin has primarily focused on transferring knowledge between models of the same architecture and number of parameters (Ainsworth et al., 2023; Rinaldi et al., 2025). When models differ in size, however, the challenge becomes more pronounced due to the mismatch in parameter dimensions and representations. Previous approaches to this problem have relied on techniques such as knowledge distillation (Hinton et al., 2015) or fine-tuning smaller models initialized from larger ones (Turc et al., 2019; Chen et al., 2022) or vice versa (Chen et al., 2015). However, these methods usually require additional training and do not directly transfer the task-specific update itself.

[1]AImageLab, University of Modena and Reggio Emilia, Modena, Italy. Correspondence to: Filippo Rinaldi <filippo.rinaldi@unimore.it>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

In this work, we introduce a novel *task-vector transport* framework for transferring task-specific parameter updates between models of different sizes (Fig. 1). Rather than operating in parameter space, our approach focuses on how task updates modify intermediate representations, enabling transfer across models with mismatched widths or identical architectures.

Despite the apparent simplicity of transferring an update between two layers, this problem is fundamentally ill-posed when models differ in width. A task update learned in one model acts on a representation space whose geometry, dimensionality, and basis are generally incompatible with those of another model. Thus, directly resizing or projecting parameters does not preserve the functional effect of the update on the model's activations. Any meaningful transfer must therefore account for how task-specific updates interact with the internal representations induced by each model.

We address this challenge by formulating task-vector transport as a functional matching problem on observed activations. We characterize a task update by the bilinear form it induces between input and output activations of a layer, and seek a corresponding update in the target model that reproduces this effect. Since this problem is underdetermined across mismatched representation spaces, we resolve the ambiguity by aligning the representation spaces of the two models using orthogonal Procrustes analysis. This approach identifies shared subspaces that preserve norms and angles, yielding a closed-form, training-free transport rule.

At a conceptual level, this perspective resembles the classical Ship of Theseus (Hobbes, 1656). As model architectures evolve in width, depth, or pre-training data, their parameters may share no direct correspondence, yet we may still ask whether a task-specific update remains the same when transferred across models. We argue that task identity should be defined by functional behavior rather than parameter values. THESEUS embodies this view by transporting task updates in a way that preserves their effect on representations. In this work, we focus primarily on transport across heterogeneous Transformer variants differing in width, depth, and pre-training distributions. In summary, our contributions are threefold:

- We formulate task-vector transport across heterogeneous models as a functional matching problem based on the effect of task updates on layer activations.
- We introduce a Procrustes-based alignment that endows the transport problem with a principled geometric structure and a closed-form transport rule.
- We show that THESEUS enables effective transfer of task-specific updates across models with different widths and pre-training distributions, outperforming vision and language baselines.

2. Related Work

Task updates and parameter-space transfer. Recent works have explored treating task-specific fine-tuning updates as first-class objects that can be manipulated, transferred, or composed in parameter space (Ilharco et al., 2023; Marczak et al., 2025; Yadav et al., 2023; Panariello et al., 2025a; Porrello et al., 2026). Rebasin methods exploit symmetries in neural networks to align weights across independently trained models with identical architectures, enabling interpolation or update transfer (Ainsworth et al., 2023; Singh & Jaggi, 2020; Rinaldi et al., 2025). Related work studies task arithmetic and model editing, where parameter differences encode task information that can be combined (Ilharco et al., 2023; Wortsman et al., 2022a). Recent advances have introduced more sophisticated alignment strategies: Nasery et al. (2025) use permutations and least squares to optimize neuron alignment, while Stoica et al. (2024) enable the merging of models from distinct tasks by "zipping" similar features across layers. For transformer-based architectures, Imfeld et al. (2024) leverage optimal transport to reconcile the permutation symmetries inherent in attention blocks. However, these approaches rely on architectural equivalence and matching parameter dimensionality. When models differ in width or internal representation size, direct parameter alignment, padding, or projection no longer preserves the functional effect of the update, making the transfer problem ill-posed. Our work departs from parameter-centric alignment and instead defines task identity through its induced functional effect on representations, enabling transfer across models with mismatched widths.

Knowledge transfer and representation alignment. A large body of work addresses transfer across heterogeneous models using training-based techniques such as knowledge distillation, teacher–student learning, or architecture-aware transformations (Hinton et al., 2015; Turc et al., 2019; Chen et al., 2015; Kangaslahti et al., 2026). While effective, these methods require additional optimization and relearn the task in the target model rather than transporting an existing update. Separately, representation alignment methods such as canonical correlation analysis and orthogonal Procrustes have been widely used to compare and translate neural representations across models, layers, and training runs (Panariello et al., 2025b; Raghu et al., 2017; Gargiulo et al., 2025; Williams et al., 2021). Related work has also explored transferring latent-space interventions and representation-space transformations across models (Moschella et al., 2023; Maiorca et al., 2023). In contrast, THESEUS uses aligned representations to derive a closed-form transport rule for task-specific parameter updates through a functional operator-matching objective, enabling transport across models with mismatched dimensionalities. We provide a detailed empirical ablation comparing THESEUS against feature translation methods in Sec. G.

3. Method

Setting. Let θ_A and θ_B denote the parameters of the base models A and B, which share a common Transformer-style structure but may differ in width, depth, or pre-training. We treat A as the source model and B as the target model. Let ℓ denote a linear or attention submodule (*e.g.*, an MLP projection or an attention head projection). After fine-tuning model A on a downstream task, we define the updated weights of layer ℓ as $\theta_A^\ell + \tau_A^\ell \in \mathbb{R}^{d_{\text{out},A} \times d_{\text{in},A}}$. Our goal is to construct a corresponding update $\tau_B^\ell \in \mathbb{R}^{d_{\text{out},B} \times d_{\text{in},B}}$ for model B such that the function implemented by layer ℓ after applying the update $\theta_B^\ell + \tau_B^\ell$ approximates the function implemented by the updated layer in model A.

Given a set of N inputs we denote the input and output activations of layer ℓ in models A and B as:

$$\begin{aligned} H_{\text{in},A}^{(\ell)} &\in \mathbb{R}^{N \times L^A \times d_{\text{in},A}}, \quad H_{\text{out},A}^{(\ell)} \in \mathbb{R}^{N \times L^A \times d_{\text{out},A}}, \\ H_{\text{in},B}^{(\ell)} &\in \mathbb{R}^{N \times L^B \times d_{\text{in},B}}, \quad H_{\text{out},B}^{(\ell)} \in \mathbb{R}^{N \times L^B \times d_{\text{out},B}}, \end{aligned} \tag{1}$$

where L^A and L^B are the sequence lengths of models A and B, and $d_{\text{in},A}, d_{\text{out},A}, d_{\text{in},B}, d_{\text{out},B}$ are the input and output dimensions of the layer ℓ in models A and B respectively. In the following, we drop the superscript ℓ to ease the notation.

Token Preprocessing. To align the input activations of the two models, we first ensure that they have the same sequence length. If $L^A < L^B$, we interpolate the sequence length of model A to match L^B using bilinear interpolation. We then flatten the first two dimensions of the input activations; let $M = N \times L^B$ be the total number of tokens after interpolation. The input activations can then be represented as $H_{\text{in},A} \in \mathbb{R}^{M \times d_{\text{in},A}}$ and $H_{\text{in},B} \in \mathbb{R}^{M \times d_{\text{in},B}}$, and similarly for the output activations. We evaluate different sequence alignment strategies in Sec. E.1.

Objective. We recast task-vector transport as the problem of preserving the *functional effect* of a task update on observed activations. Consider a linear layer whose output activations are given by $H_{\text{out}} = H_{\text{in}}\theta^\top$, where θ denotes the layer weight matrix of the base model. Applying a task update τ_A to model A induces a change in the output activations

$$\Delta H_{\text{out},A} = H_{\text{in},A}\tau_A^\top. \tag{2}$$

This expression characterizes how the update modifies the layer outputs for the given inputs. Rather than matching $\Delta H_{\text{out},A}$ pointwise, we characterize the update by how it *couples* input and output activations across the dataset. Specifically, we define the matrix

$$G_A \;=\; \Delta H_{\text{out},A} H_{\text{out},A}^\top \in \mathbb{R}^{M \times M}, \tag{3}$$

which aggregates the interaction between input activations, the task update, and the pre-update output representations.

Algorithm 1 Training-free Task-Vector Transport

Require: Source weights θ_A, θ_A^{ft}, target weights θ_B, calibration dataset $\mathcal{D}$, scaling α, sequence lengths L^A, L^B.

for each layer $l \in \{1, \dots, L\}$ **do**
  Compute source task vector: $\tau_A^l \leftarrow \theta_A^{ft,l} - \theta_A^l$
  Collect $\{H_{in,h}, H_{out,h}\}_{h \in \{A,B\}}$ from $\mathcal{D}$
  if $L^A \neq L^B$ **then** interpolate $\min\{L^A, L^B\}$
  Compute cross-covariances:
    $C_{in} \leftarrow H_{in,A}^\top H_{in,B}, \quad C_{out} \leftarrow H_{out,A}^\top H_{out,B}$
  Compute SVD: ▷ Eq. (6)
    $C_{in} = U_{in}\Sigma_{in}V_{in}^\top, C_{out} = U_{out}\Sigma_{out}V_{out}^\top$
  Solve orthogonal Procrustes problems: ▷ Eq. (7)
    $T_{in} \leftarrow U_{in}V_{in}^\top, \quad T_{out} \leftarrow U_{out}V_{out}^\top$
  Compute transported update: ▷ Eq. (10)
    $\tau_B^l \leftarrow T_{out}\tau_A^l T_{in}^\top$
end for
Return $\theta_B + \alpha\tau_B$

For any pair of tokens (i, j), the entry $(G_A)_{ij}$ measures how the update-induced change at token i aligns with the original output representation at token j. G_A defines a bilinear form over token representations that captures the global functional signature of the task update on the observed data.

Thus, we seek an update τ_B for model B whose induced bilinear form matches that of τ_A:

$$\min_{\tau_B} \left\| H_{\text{in},A}\tau_A^\top H_{\text{out},A}^\top - H_{\text{in},B}\tau_B^\top H_{\text{out},B}^\top \right\|_F^2. \tag{4}$$

Matching these bilinear forms ensures that τ_B reproduces the same input-output interaction structure as τ_A, even when the two models operate in different representation spaces.

While Eq. (4) provides a principled formulation of task-vector transport, it admits many solutions when the representation spaces of the two models are mismatched or only partially observed. In particular, unconstrained least-squares alignment becomes underdetermined whenever the sampled activations are rank-deficient, allowing solutions that arbitrarily amplify poorly conditioned directions. Rather than resolving this ambiguity through direct matrix inversion, we impose orthogonality constraints on the alignment maps. This removes scaling ambiguities, regularizes the transport problem, and preserves the norm and geometry of the transported update. Sec. B provides the full derivation showing how the constrained problem reduces to a trace maximization solved in closed form by orthogonal Procrustes.

Geometric Alignment via Procrustes. We therefore align the representation spaces of the two models through orthonormal embeddings. Concretely, we seek orthogonal alignment maps that identify the maximally aligned shared subspaces while preventing arbitrary rescaling of poorly conditioned directions. This alignment is obtained by solving

two orthogonal Procrustes problems:

$$\min_{T_{\text{in}}} \|H_{\text{in},A}T_{\text{in}} - H_{\text{in},B}\|_F^2 \quad \text{s.t.} \quad T_{\text{in}}T_{\text{in}}^\top = I_{d_{\text{in},A}},$$
$$\min_{T_{\text{out}}} \|H_{\text{out},A}T_{\text{out}} - H_{\text{out},B}\|_F^2 \quad \text{s.t.} \quad T_{\text{out}}T_{\text{out}}^\top = I_{d_{\text{out},A}}. \tag{5}$$

Here $T_{\text{in}} \in \mathbb{R}^{d_{\text{in},A} \times d_{\text{in},B}}$ and $T_{\text{out}} \in \mathbb{R}^{d_{\text{out},A} \times d_{\text{out},B}}$ are rectangular matrices with orthonormal rows. They define orthogonal alignments between the dominant shared subspaces of the two models, while accommodating mismatched widths without requiring a bijective feature correspondence. Both admit closed-form solutions via Singular Value Decomposition (SVD) of the cross-covariances $C_{\text{in}} = H_{\text{in},A}^\top H_{\text{in},B}$, and $C_{\text{out}} = H_{\text{out},A}^\top H_{\text{out},B}$:

$$C_{\text{in}} = U_{\text{in}}\Sigma_{\text{in}}V_{\text{in}}^\top, \quad C_{\text{out}} = U_{\text{out}}\Sigma_{\text{out}}V_{\text{out}}^\top, \tag{6}$$

yielding

$$T_{\text{in}} = U_{\text{in}}V_{\text{in}}^\top, \quad T_{\text{out}} = U_{\text{out}}V_{\text{out}}^\top. \tag{7}$$

The resulting embeddings minimize the discrepancy between aligned activations in a least-squares sense. Consequently, the activations of model B can be approximated through aligned representations of model A:

$$H_{\text{in},B} \approx H_{\text{in},A}T_{\text{in}}, \qquad H_{\text{out},B} \approx H_{\text{out},A}T_{\text{out}}. \tag{8}$$

When the target dimensionality exceeds the source dimensionality, the transport acts as a norm-preserving embedding into a higher-dimensional aligned subspace. Conversely, when the target dimensionality is smaller, it reduces to a projection onto the maximally aligned shared subspace. Thus, the approximation in Eq. (8) reflects the fact that transport is restricted to the shared aligned components of the two representation spaces. Substituting these aligned representations into Eq. (4), we obtain the following *surrogate* objective:

$$\min_{\tau_B} \left\| H_{\text{in},A}\tau_A^\top H_{\text{out},A}^\top - (H_{\text{in},A}T_{\text{in}})\tau_B^\top (H_{\text{out},A}T_{\text{out}})^\top \right\|_F^2$$

which can be rewritten as

$$\min_{\tau_B} \left\| H_{\text{in},A}\, E\, H_{\text{out},A}^\top \right\|_F^2, \quad E := \tau_A^\top - T_{\text{in}}\tau_B^\top T_{\text{out}}^\top. \tag{9}$$

This formulation restricts transport to the aligned shared subspaces identified by Procrustes. Under this aligned-subspace approximation, the transport objective is minimized when the residual term vanishes, *i.e.*, $E = 0$, yielding the closed-form transport rule

$$\tau_B = T_{\text{out}}\tau_A T_{\text{in}}^\top. \tag{10}$$

Thus, the transported update τ_B is obtained by orthogonally transporting the source update τ_A into the target representation space while preserving its dominant functional effect under the aligned-subspace approximation. A full derivation is provided in Sec. A. We summarize the complete pipeline in Algorithm 1.

3.1. Properties and Scope

THESEUS Preserves the Norm. A key property of the transport rule in Eq. (10) is that it preserves the Frobenius norm of the task update:

$$\|\tau_B\|_F = \|T_{\text{out}}\tau_A T_{\text{in}}^\top\|_F = \|\tau_A\|_F, \tag{11}$$

since T_{in} and T_{out} have orthonormal rows. This norm preservation ensures that the magnitude of the update remains consistent across models, preventing unintended scaling effects and preserving the relative strength of the task update across architectures. Beyond preserving the update magnitude, orthogonality also stabilizes the transport problem. Indeed, in unconstrained least-squares formulations, rank-deficient activation covariances admit infinitely many equivalent solutions that can arbitrarily amplify poorly conditioned directions. By restricting the transport maps to orthogonal transformations, we remove these scaling ambiguities while preserving the norm and relative geometry of the update within the aligned subspace.

On the Generalization of THESEUS. While we present the method assuming model A has lower-dimensional representations than model B, the formulation is symmetric. In practice, task-vector transport can be applied in either direction by swapping the roles of the two models. Moreover, when the two models have representations of equal dimensionality, the Procrustes mappings reduce to square orthogonal transformations, and the same transport rule applies without modification. We empirically study both settings in Sec. 4.

Extension to Different Depths. Our formulation assumes that the two models have the same number of layers; however, it can be extended to different depths by combining THESEUS with a separate layer-matching or interpolation strategy. We provide preliminary results on this extension in Sec. 4.4, leaving a detailed exploration to future work.

4. Experiments

We evaluate the efficacy of our transport method across diverse vision and language benchmarks. Our experimental evaluation is organized around three questions. First, we assess our training-free functional matching approach when transporting updates between models of **different widths**. Second, we evaluate transfer across different pre-training distributions for models with **identical architectures**, enabling direct comparison with prior rebasin methods. Finally, we move beyond the core assumptions of our framework and evaluate a challenging regime in which both depth and width differ (*e.g.*, ViT-B $\rightarrow$ ViT-L), showing that THESEUS enables transport across **different widths and depths**.

Table 1. **Width Scaling (Narrow → Wide).** Task-vector transport from a ViT-B/16 model pre-trained on `LAION-2B` (A) to the wider ViT-B/16+ model pre-trained on `LAION-400M` (B). Results are reported for different numbers of alignment batches $\mathcal{B}$ used to estimate the Procrustes maps. ΔAcc denotes the accuracy improvement over the zero-shot baseline of model B. THESEUS represents $\theta_B + \tau_B$.

Model	$\mathcal{B}$	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*	–	50.92	39.23	49.63	64.53	55.48	84.53	57.06	68.67	58.76 (+0.00)
θ_B *fine-tune*	–	98.96	91.08	98.63	92.59	77.81	87.65	99.63	76.76	90.39 (+31.63)
θ_A *fine-tune*	–	98.69	97.45	98.64	95.65	82.24	91.53	99.61	79.86	92.96 (+34.20)
$\theta_B + \tau_A^{\text{padded}}$	–	48.82	38.17	48.45	65.14	54.94	84.52	58.52	69.46	58.50 (−0.26)
$\theta_B + \tau_B^{\text{random}}$	–	50.63	38.96	49.65	65.65	55.58	84.56	57.39	69.60	59.00 (+0.24)
$\theta_B + \tau_{\text{pinv}}$	1	14.29	14.54	31.66	54.73	48.94	69.61	40.49	64.79	42.38 (−16.38)
$\theta_B + \tau_{\text{pinv-Tikh}}$	1	47.83	38.24	50.23	63.55	56.12	82.29	54.68	68.49	57.68 (−1.08)
$\theta_B + \tau_A^{\text{random}\to B}$	1	51.53	39.49	49.79	65.44	55.53	84.31	57.33	69.61	59.13 (+0.37)
THESEUS	1	60.41	56.80	56.62	66.98	57.23	84.62	64.49	69.60	**64.59 (+5.83)**
$\theta_B + \tau_{\text{pinv}}$	2	44.34	41.92	48.19	64.22	56.43	77.59	20.44	68.88	52.25 (−6.51)
$\theta_B + \tau_{\text{pinv-Tikh}}$	2	53.04	46.23	50.11	64.52	57.18	81.83	55.13	68.76	59.60 (+0.84)
$\theta_B + \tau_A^{\text{random}\to B}$	2	50.96	39.41	49.75	65.60	55.48	84.41	56.62	69.51	58.97 (+0.21)
THESEUS	2	64.03	59.30	59.55	68.12	57.86	84.89	68.97	70.12	**66.60 (+7.84)**
$\theta_B + \tau_{\text{pinv}}$	5	52.25	44.82	50.12	64.24	56.92	82.51	27.72	69.85	56.80 (−1.96)
$\theta_B + \tau_{\text{pinv-Tikh}}$	5	54.51	46.16	50.34	65.53	57.93	82.12	56.97	68.76	60.29 (+1.53)
$\theta_B + \tau_A^{\text{random}\to B}$	5	50.79	39.24	49.69	65.54	55.37	84.26	56.92	69.55	58.92 (+0.16)
THESEUS	5	63.59	59.61	60.46	68.40	58.83	84.85	67.01	70.35	**66.64 (+7.88)**
THESEUS	10	65.91	61.59	61.23	68.73	59.52	84.93	74.46	70.42	**68.35 (+9.59)**
THESEUS	20	67.01	62.52	61.01	69.27	60.48	85.04	76.66	70.65	**69.08 (+10.32)**
THESEUS	50	67.13	64.29	59.46	70.01	61.28	85.19	77.23	71.28	**69.48 (+10.72)**

4.1. Experimental Setting

Task-specific updates are obtained by fine-tuning a source model on a downstream task and are transported to a target model without additional optimization. THESEUS is applied to all Transformer linear submodules, including attention projections and MLP layers, while positional and patch embeddings are excluded. Alignment operations are performed using frozen models and require only forward passes. Full implementation details are provided in Sec. C.

Activation Collection. Procrustes maps (T_{in} and T_{out}) are estimated independently for each layer using activations collected from calibration batches. For vision models, all spatial tokens are retained after sequence interpolation, while for language models we use the full token sequence. A naive computation would require storing the full activation tensors $H_{\text{in},A}, H_{\text{in},B}, H_{\text{out},A}, H_{\text{out},B}$, each of shape $\mathbb{R}^{M\times d}$, in memory. Under this approach, the memory footprint would grow as the total token count M increases with every additional calibration batch. Instead, we compute activation statistics incrementally by aggregating the cross-covariance matrices $C_{\text{in}} = H_{\text{in},A}^\top H_{\text{in},B} \in \mathbb{R}^{d_{\text{in},A}\times d_{\text{in},B}}$ and $C_{\text{out}} = H_{\text{out},A}^\top H_{\text{out},B} \in \mathbb{R}^{d_{\text{out},A}\times d_{\text{out},B}}$ across batches, keeping the memory footprint bounded by the models' internal feature dimensions.

This implementation decouples the memory requirements from the number of calibration tokens M, allowing the alignment process to scale to large calibration sets. The resulting covariance estimates are then used to compute the orthogonal Procrustes alignments via SVD. Further details and complexity analysis are provided in Sec. D.

Few-shot protocol. We evaluate our method using a $\mathcal{B}$-shot protocol, where $\mathcal{B} \in \{1, 2, 5, 10, 20\}$ denotes the number of sampled batches, each with a fixed batch size of 32, used to characterize the representation spaces of models A and B. For each value of $\mathcal{B}$, we sample training batches uniformly at random to construct the support set required for Procrustes alignment (Eq. (5)). Additionally, to maintain consistency with existing literature, we also report results under a standard $\mathcal{K}$-shot evaluation protocol with $\mathcal{K} \in \{1, 2, 5, 10, 20\}$ examples per class.

Update Scaling. Following standard practice in the task-vector and model editing literature (Ilharco et al., 2023; Gargiulo et al., 2025; Marczak et al., 2025; Yadav et al., 2023; Rinaldi et al., 2025; Wortsman et al., 2022b), we integrate the transported update into the base model through a scalar interpolation α, chosen via a validation set, applied as $\theta + \alpha\tau$. All reported results correspond to the value of α that achieves the best performance in a linear search.

4.2. Transfer across Mismatched Widths

Vision. We evaluate task-vector transport across models of differing widths in a vision setting. Specifically, we transport task updates τ_A learned by a CLIP ViT-B/16 model pre-trained on `LAION-2B` to the wider ViT-B/16+ variant from OpenCLIP (Cherti et al., 2023), pre-trained on `LAION-400M`. We also evaluate reverse transport from the wider model to a smaller ViT-B/16 target ($B \to A$). In this reverse setting, the target model is a ViT-B/16 pre-trained on the stronger `DataComp-XL` dataset. We evaluate on the 8-Vision benchmark (Ilharco et al., 2023); more details are provided in Sec. F.1.

Baselines. For width upscaling transfer, we compare our method against several baselines in Tab. 1. We report the zero-shot performance of the target model B as a lower bound and fully fine-tuned models A and B as upper bounds. As naive baselines, we include: (i) zero-padding the source update τ_A to match the dimensionality of model B, and (ii) adding a randomly initialized update to model B. Both ignore representation alignment and test whether dimensional compatibility alone preserves task behavior. We also evaluate a pseudo-inverse transport baseline (τ_{pinv}), which solves Eq. (4) using the Moore–Penrose pseudo-inverse of the activation matrices, implemented through the truncated SVD approximation in `torch.linalg.pinv`.

We further consider a Tikhonov-regularized variant ($\tau_{\text{pinv-Tikh}}$). While regularization improves numerical stability, both methods lack explicit representation alignment. Additional ablations are reported in Sec. G. Finally, we include a randomized alignment baseline ($\tau_A^{\text{random}\to B}$), where a random update is transported through the learned Procrustes maps.

Results. As shown in Tab. 1, THESEUS consistently improves over the zero-shot baseline for forward transfer ($A \to B$) across all $\mathcal{B}$-shot regimes, without requiring gradient-based updates on the target model. While a gap to supervised fine-tuning remains, the method recovers a substantial fraction of the achievable gains, particularly in low-shot settings. Additional results for the $\mathcal{K}$-shot protocol and reverse transport ($B \to A$) are reported in Sec. E.2, where we observe similar trends.

In the reverse setting, the target model A is pre-trained on `DataComp-XL`, yielding a stronger zero-shot baseline. Nevertheless, THESEUS consistently improves performance even in this high-accuracy regime, showing that transported updates can inject task-specific knowledge into already strong base models. In both transfer directions, the proposed alignment is substantially more stable than direct matrix-inversion approaches, whose pseudo-inverse variants deteriorate in low-shot regimes due to rank-deficient activation matrices.

Table 2. Linear-probing performance with and without the transported task vector. Source: `T5-3B` (A), target: `T5-Large` (B). THESEUS represents $\theta_B + \tau_B$.

Encoder	$\mathcal{B}$	MNLI	QNLI	RTE	SCITAIL	SNLI	AVG (ΔAcc)
θ_A *fine-tune*	All	89.88	95.78	87.73	92.45	90.45	91.26 (-)
θ_B *fine-tune*	All	86.34	92.14	81.23	91.12	88.75	87.92 (-)
θ_A	20	66.84	80.85	71.84	69.40	45.06	66.80 (+8.39)
θ_B	20	64.87	66.61	64.62	49.85	43.42	57.87 (+0.00)
THESEUS	20	78.67	86.09	74.00	84.74	76.04	**79.91** (+22.04)
θ_A	50	74.91	82.43	74.01	68.67	72.61	74.53 (−2.25)
θ_B	50	78.58	81.79	76.17	75.15	72.20	76.78 (+0.00)
THESEUS	50	81.81	89.10	80.90	88.30	80.30	**84.08** (+7.30)
θ_A	100	80.14	90.59	79.42	82.06	80.78	82.60 (+0.69)
θ_B	100	81.04	84.55	81.59	82.36	80.02	81.91 (+0.00)
THESEUS	100	83.60	90.70	83.80	89.00	82.10	**85.84** (+3.93)

Language. To assess whether task-vector transport generalizes beyond vision models, we evaluate THESEUS in a language setting using an encoder–decoder Transformer (Raffel et al., 2020). We consider `T5-3B` as the source model A and `T5-Large` as the target model B. Both models are treated as text encoders (θ) followed by task-specific classification heads (ω). Our protocol is designed to isolate the effect of the transported *encoder* representations. We first fine-tune model A on each downstream task, computing a task-specific encoder update τ_A for θ_A. We then transport this encoder update to model B using THESEUS , yielding the updated encoder $\theta_B + \tau_B$.

To evaluate the quality of the learned representations, we adopt a linear probing protocol (Alain & Bengio, 2017). In all experiments, encoder parameters are kept strictly frozen, and a newly initialized linear classification head is trained from scratch. We consider three fixed encoders: the source encoder θ_A, the target encoder θ_B, and the target encoder augmented with the transported task update, $\theta_B + \tau_B$ (THESEUS). All heads are trained under identical optimization settings, ensuring that performance differences reflect solely the quality of the underlying representations rather than any additional encoder fine-tuning.

We evaluate this setup on five closed-vocabulary natural language inference tasks from the GLUE benchmark (Wang et al., 2018). We vary the number of batches $\mathcal{B}$ used to estimate the Procrustes maps and to train the classifier, using the same data budget for all methods. A detailed description of datasets and preprocessing is provided in Sec. F.2. This evaluation protocol allows us to directly measure how much task-relevant structure is transferred by THESEUS into the representation space of the target encoder, without interference from further encoder fine-tuning. By constraining adaptation to a linear classifier, we test whether the transported update reorganizes the target representation space in a task-aware manner.

Table 3. **Identical Architecture Transfer.** We report task-vector transport accuracies between ViT-B/16 models with identical architectures but different pre-training distributions (A: `DataComp-XL` $\rightarrow$ B: `LAION-2B`). Results are reported for varying support set sizes $\mathcal{K}$ used for alignment or optimization. ΔAcc denotes the average accuracy improvement over the zero-shot baseline of model B. The **No Grad.** column indicates whether the method requires gradients. THESEUS represents $\theta_B + \tau_B$.

Model	$\mathcal{K}$	No Grad.	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*	–	✓	49.41	50.58	48.29	67.98	55.96	84.50	57.10	68.70	60.31 (+0.00)
θ_B *fine-tune*	–	✗	98.70	97.45	98.65	95.66	83.19	88.10	99.60	77.20	92.32 (+32.01)
$\theta_B + \tau_A$	–	✓	49.58	50.84	49.31	67.87	56.27	84.60	57.40	68.90	60.60 (+0.28)
TransFusion	–	✓	50.12	67.99	50.24	53.26	56.70	84.96	59.26	69.12	61.46 (+1.14)
GradFix	1	✗	61.94	71.07	60.88	70.05	58.32	87.95	86.39	72.67	71.16 (+10.84)
THESEUS	1	✓	58.64	71.38	60.30	71.95	59.79	88.89	87.17	73.08	**71.40 (+11.09)**
+ GradFix	1	✗	62.09	73.04	64.13	70.66	58.62	87.92	88.15	72.93	**72.19 (+11.88)**
GradFix	2	✗	65.07	70.19	64.33	71.42	58.51	87.15	88.38	72.92	72.25 (+11.93)
THESEUS	2	✓	65.49	75.66	62.67	72.38	60.48	88.85	91.56	73.02	**73.76 (+13.45)**
+ GradFix	2	✗	67.73	75.53	66.51	72.49	60.21	87.18	90.67	73.14	**74.18 (+13.87)**
GradFix	5	✗	66.05	73.59	66.61	71.57	60.02	86.82	89.06	72.97	73.34 (+13.02)
THESEUS	5	✓	66.12	72.09	61.82	73.33	61.48	88.80	93.03	73.19	**73.73 (+13.42)**
+ GradFix	5	✗	69.48	74.72	68.48	72.55	61.22	87.01	90.16	73.16	**74.60 (+14.28)**
GradFix	10	✗	66.59	74.82	66.02	72.05	60.18	86.64	89.67	73.18	73.64 (+13.33)
THESEUS	10	✓	65.21	73.95	63.46	74.09	62.29	88.83	92.48	73.21	**74.19 (+13.88)**
+ GradFix	10	✗	67.63	76.92	68.11	72.62	61.44	86.29	90.68	73.33	**74.63 (+14.31)**
GradFix	20	✗	67.05	74.11	66.42	72.29	60.92	85.99	89.63	73.06	73.68 (+13.37)
THESEUS	20	✓	68.42	74.82	65.73	74.16	63.77	88.92	94.62	73.33	**75.47 (+15.16)**
+ GradFix	20	✗	66.76	75.08	68.40	73.54	61.48	85.85	90.60	73.19	**74.36 (+14.05)**

As shown in Tab. 2, transporting the encoder update from `T5-3B` to `T5-Large` consistently improves linear probing performance across all evaluated tasks and data regimes. While a gap relative to fully fine-tuned models remains, THESEUS recovers a substantial fraction of the task-specific signal using only forward passes and closed-form alignment. Performance improves monotonically as the number of alignment batches $\mathcal{B}$ increases, indicating that richer activation statistics lead to more accurate alignment of the linguistic representation spaces.

Notably, the gains are obtained while keeping the target encoder completely frozen, suggesting that the transported task vector reorganizes the target representation space in a task-aware manner rather than merely transferring superficial classifier-level statistics. These results demonstrate that functional task-vector transport extends beyond vision models and applies effectively to encoder–decoder language architectures despite their additional structural complexity.

To evaluate THESEUS on a non-classification task, we transport a summarization task update from `Flan-T5` to `T5-v1.1`. Using the ROUGE-L (Lin, 2004) metric, we observe a 36.5% relative improvement over the zero-shot target model, demonstrating that our activation-alignment mechanism extends beyond classification to a generative language task. Additional details are reported in Sec. E.3.

4.3. Transfer on Identical Architectures

We further evaluate THESEUS in a controlled setting where models share an identical ViT-B/16 architecture but differ only in their pre-training distributions, mirroring the protocol from Rinaldi et al. (2026). Specifically, we transport task updates from a source model A pre-trained on `DataComp-XL` to a target model B pre-trained on `LAION-2B`. This configuration enables a direct comparison with TransFusion (Rinaldi et al., 2025) and GradFix. Following prior work, we adopt a standard $\mathcal{K}$-shot protocol, where $\mathcal{K}$ denotes the number of examples per class used for alignment or optimization.

Results in Tab. 3 show that our method consistently outperforms TransFusion across all shot regimes and remains highly competitive with GradFix under the same conditions. While GradFix relies on repeated backward passes and iterative optimization, THESEUS achieves comparable or superior performance using only forward activations and a closed-form transport rule. Moreover, performance scales favorably with increasing supervision, matching or exceeding GradFix in higher-shot regimes. These results show that functional matching via orthogonal alignment provides an effective alternative to gradient-based task-vector transport even when models share the same architecture, demonstrating that the proposed formulation is not limited to heterogeneous-width settings.

Table 4. **Depth and Width Scaling.** Task-vector transport from a ViT-B/16 model to a ViT-L/14 model, both pre-trained on `DataComp-XL`. This setting involves simultaneous depth and width mismatches and uses a naive layer-interpolation heuristic to align Transformer depth before transport. Results are reported for $\mathcal{B} = 10$ alignment batches. ΔAcc denotes the average accuracy improvement over the zero-shot baseline of model B. THESEUS represents $\theta_B + \tau_B$.

Model	$\mathcal{B}$	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_A *interpolated*	–	16.04	18.37	6.44	4.78	3.99	0.72	16.58	1.07	8.50 (−65.13)
θ_B *zero-shot*	–	68.40	67.44	58.96	71.27	66.81	93.01	86.84	76.34	73.63 (+0.00)
θ_A *fine-tune*	–	98.66	97.70	98.77	95.40	83.29	92.43	99.64	79.91	93.23 (+19.60)
θ_B *fine-tune*	–	99.00	98.22	98.42	96.73	86.88	94.80	99.76	82.94	94.59 (+20.96)
THESEUS	10	76.78	78.34	73.59	75.02	72.71	93.10	95.18	77.16	**80.24** (+6.61)

4.4. Transfer across Depth and Width

We finally assess structural generalization by transporting task updates between models that differ in both depth and width. Specifically, we transfer updates from a ViT-B/16 source model A to a ViT-L/14 target model B, both pre-trained on `DataComp-XL`. This setting is particularly challenging since it requires reconciling mismatches in both representation dimensionality and transformer depth.

To extend THESEUS beyond equal-depth settings, we adopt a simple interpolation-based heuristic to induce a coarse layer correspondence. Starting from a ViT-B model with 12 transformer blocks, we construct a depth-expanded model by inserting one additional block after each original block, yielding a 24-layer architecture compatible with ViT-L. Each inserted block B'_i is obtained by linearly interpolating the parameters of adjacent blocks, $\theta_{B'_i} = 0.5\,\theta_{B_i} + 0.5\,\theta_{B_{i+1}}$, while the final inserted block is a copy of the last transformer block. The interpolation is applied to all transformer parameters, including LayerNorms, attention projections, MLP layers, and LayerScale coefficients. The same interpolation procedure is applied independently to both the base model A and its fine-tuned counterpart A^{ft}, and the enlarged task update is computed from the resulting interpolated models. We then apply THESEUS to handle the remaining width mismatch between the two architectures.

We emphasize that this depth alignment is heuristic and intentionally naive. Indeed, interpolation is highly destructive: the source model drops to 8.50% zero-shot accuracy and 17.88% fine-tuned accuracy after interpolation alone. Consequently, interpolation does not preserve the source predictor and serves only to induce a depth correspondence.

Nevertheless, as shown in Tab. 4, THESEUS still enables effective task-vector transport on top of this degraded initialization, consistently improving over the target ViT-L zero-shot baseline. These results suggest that, despite severe parameter-space distortion, sufficient layerwise structure remains for activation-space alignment to recover transferable task directions. A more principled treatment of depth alignment remains an important direction for future work.

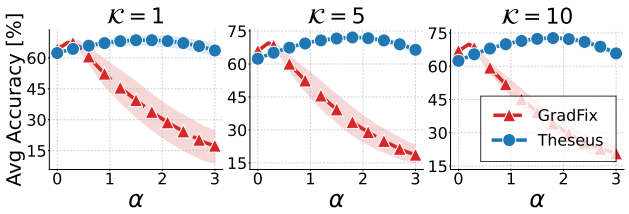

Figure 2. **Robustness to update scaling (α).** THESEUS exhibits high stability across different fixed α values, whereas GradFix experiences severe fluctuations in accuracy.

4.5. Sensitivity to Update Scaling

Standard task-vector transport and merging methods (Marczak et al., 2025; Ilharco et al., 2023; Gargiulo et al., 2025; Yadav et al., 2023; Rinaldi et al., 2026) typically rely on a labeled validation set to select the optimal scalar interpolation coefficient α for integrating the transported update, applied as $\theta + \alpha\tau$. While this avoids gradient-based optimization on the target model, the reliance on target-task validation data partially relaxes the strict notion of training-free transfer. To evaluate the robustness of THESEUS in a genuinely label-free regime, we analyze its performance across a sweep of α values, with particular interest in the unscaled $\alpha = 1.0$ setting, which eliminates the need for any downstream validation search.

As depicted in Fig. 2, THESEUS remains robust to scaling adjustments, maintaining a stable plateau of high accuracy across a wide range of α values and support set sizes ($\mathcal{K}$). In contrast, gradient-based baselines such as GradFix exhibit extreme sensitivity to this scaling coefficient, peaking sharply at low α values before decaying precipitously. THESEUS scores $71.21/73.32/73.33$ for $\alpha = 1/1.5/2$, while GradFix drops from 54.54 to $44.64/36.12$.

This stability highlights a theoretical advantage of our formulation. By explicitly preserving the Frobenius norm and the geometric structure of the original update during transport (Eq. (11)), the unscaled vector ($\alpha = 1.0$) inherently retains its calibrated functional effect. Consequently, THESEUS is uniquely suited for strictly label-free deployments where gradient-based alternatives suffer severe degradation.

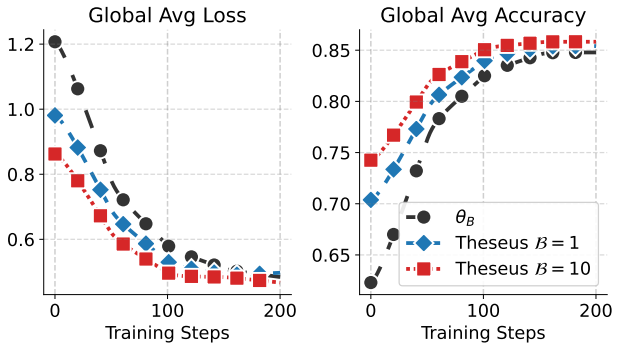

Figure 3. **Early convergence during downstream fine-tuning.** Average validation loss and accuracy on 8-Vision for standard fine-tuning from the baseline θ_B and from initialization with THESEUS .

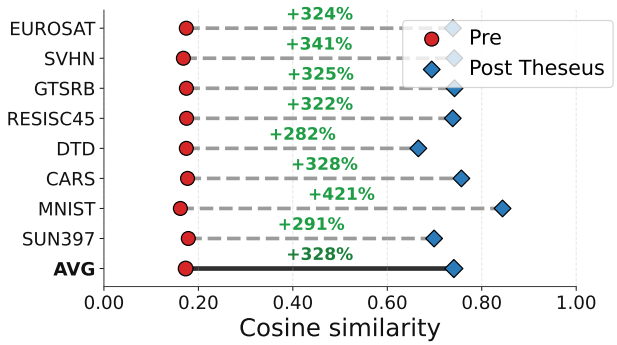

Figure 4. **Activation cosine similarity.** Cosine similarity between intermediate activations from the source and target models before (Pre) and after Procrustes alignment (Post THESEUS).

4.6. THESEUS as Warm-Start for Fine-Tuning

Finally, we investigate whether THESEUS can improve downstream fine-tuning efficiency by providing a strong initialization for subsequent optimization. Specifically, we use the transported task vector to initialize the target model θ_B prior to standard supervised fine-tuning. Fig. 3 reports the average loss and accuracy over eight vision classification tasks during the first training steps, capturing the early convergence before all models reach similar minima. Models with THESEUS start with a better initialization, exhibiting lower initial loss and higher accuracy than the standard pre-trained baseline. This warm-start effect is particularly evident in the convergence dynamics: the 10-batch alignment variant achieves after only 50 steps what the baseline θ_B reaches after 150 steps.

Overall, these results indicate that THESEUS not only enables effective zero-shot adaptation but also serves as a practical initialization strategy that reduces the computational cost of supervised fine-tuning.

4.7. Activation Space Alignment Analysis

To evaluate the effectiveness of the alignment, we measure the cosine similarity between intermediate activations from model A (H_A) and model B (H_B) before and after applying the transformation T. To ensure an out-of-sample evaluation, the alignment maps are estimated using calibration batches, while cosine similarity is measured on a disjoint validation set.

As shown in Fig. 4, the pre-trained models and their mismatched widths initially produce poorly aligned representation spaces, resulting in low baseline similarity ("Pre"). Applying the Procrustes transformations to project the source activations into the target representation space ("Post THESEUS ") improves the correspondence between activations. On 8-Vision, THESEUS yields an average relative increase in cosine similarity of $+328\%$.

These results provide strong empirical support for our central hypothesis: although the raw parameter spaces of heterogeneous models are not directly compatible, their representations remain organized around a shared latent functional subspace. By identifying and aligning this subspace, THESEUS establishes the geometric basis necessary for accurate functional transport of task-specific updates.

5. Conclusions

In this work, we introduced THESEUS, a training-free framework for transporting task-specific updates across heterogeneous pre-trained models. By formulating task-vector transfer as a functional matching problem and conditioning it through orthogonal Procrustes alignment, THESEUS yields a simple closed-form transport rule that preserves the geometric structure of task updates while naturally accommodating mismatches in representation dimensionality.

Across a wide range of experiments, we showed that THESEUS enables transfer of task knowledge across models of different widths, pre-training distributions, and architectural depths. The method consistently outperforms naive baselines and direct pseudo-inverse approaches while remaining competitive with gradient-based alternatives when the architectures match, despite requiring only forward passes.

We also showed that transported updates provide a strong initialization for downstream fine-tuning, accelerating convergence and improving adaptation efficiency. More broadly, our results support a functional view of task identity, in which task-specific knowledge is characterized by its effect on representations. This suggests that task updates can remain transferable even across evolving architectures, widths, and pre-training distributions, provided that their underlying representation geometry is properly aligned. We believe this perspective opens promising directions for modular model adaptation, continual reuse of task updates, and scalable transfer across increasingly heterogeneous models.

Acknowledgments

We acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support. The research activities of Angelo Porrello have been supported by the Department of Engineering "Enzo Ferrari" through the program FAR2025DIP (CUP E93C25000370005).

Impact Statement

This work introduces THESEUS, a training-free approach for transferring task-specific updates across pre-trained models by matching their functional effect on internal representations. By enabling reuse of task adaptations without additional optimization, our method has the potential to reduce computational costs, energy consumption, and environmental impact associated with repeatedly fine-tuning large models. This can facilitate more efficient deployment of machine learning systems, especially in resource-constrained settings or when adapting models across evolving architectures.

The proposed framework may also support more modular and sustainable model development practices, allowing practitioners to share and reuse task knowledge across model variants without access to original training data. However, as with any method that enables model adaptation, care should be taken to ensure that transferred task updates do not propagate unintended biases or harmful behaviors learned during fine-tuning. Our method preserves functional behavior by design, and thus inherits both the strengths and limitations of the source task updates.

We do not foresee direct misuse scenarios unique to THESEUS beyond those already present in standard fine-tuning and model reuse pipelines. Nonetheless, responsible deployment should include appropriate evaluation and monitoring when transferring task updates across domains or applications. We hope this work encourages further research into efficient, transparent, and accountable mechanisms for model adaptation.

References

Ainsworth, S. K., Hayase, J., and Srinivasa, S. S. Git rebasin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations*, 2017.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. A large annotated corpus for learning natural language inference. *Empirical Methods in Natural Language Processing*, 2015.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 2020.

Chen, B., Bakhshi, A., Batista, G., Ng, B., and Chin, T.-J. Update compression for deep neural networks on the edge. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2022.

Chen, T., Goodfellow, I., and Shlens, J. Net2net: Accelerating learning via knowledge transfer. In *International Conference on Learning Representations*, 2015.

Cheng, G., Han, J., and Lu, X. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 2017.

Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., and Jitsev, J. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2023.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 2023.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing textures in the wild. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Gargiulo, A. A., Crisostomi, D., Bucarelli, M. S., Scardapane, S., Silvestri, F., and Rodolà, E. Task singular vectors: Reducing task interference in model merging. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2025.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2022.

Helber, P., Bischke, B., Dengel, A., and Borth, D. 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.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, 2015.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *Neural Information Processing Systems Workshops*, 2015.

Hobbes, T. *Elements of Philosophy, the First Section, Concerning Body*. London: R & W Leybourn., 1656.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, 2019.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Schmidt, L., Hajishirzi, H., and Farhadi, A. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Imfeld, M., Graldi, J., Giordano, M., Hofmann, T., Anagnostidis, S., and Singh, S. P. Transformer Fusion with Optimal Transport. In *International Conference on Learning Representations*, 2024.

Kangaslahti, S., Nayak, N. V., Geuter, J., Fumero, M., Locatello, F., and Alvarez-Melis, D. Boomerang distillation enables zero-shot model size interpolation. In *International Conference on Learning Representations*, 2026.

Khot, T., Sabharwal, A., and Clark, P. Scitail: A textual entailment dataset from science question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. 3D Object representations for fine-grained categorization. In *IEEE International Conference on Computer Vision Workshops*, 2013.

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

Lin, C.-Y. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*. Association for Computational Linguistics, 2004.

Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., and Raffel, C. A. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*, 2022.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Maiorca, V., Moschella, L., Norelli, A., Fumero, M., Locatello, F., and Rodolà, E. Latent space translation via semantic alignment. In *Advances in Neural Information Processing Systems*, 2023.

Marczak, D., Magistri, S., Cygert, S., Twardowski, B., Bagdanov, A. D., and van de Weijer, J. No task left behind: Isotropic model merging with common and task-specific subspaces. In *International Conference on Machine Learning*, 2025.

Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., and Rodolà, E. Relative representations enable zero-shot latent space communication. In *International Conference on Learning Representations*, 2023.

Nallapati, R., Zhou, B., Dos Santos, C., Gulçehre, Ç., and Xiang, B. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL conference on computational natural language learning*, 2016.

Nasery, A., Hayase, J., Koh, P. W., and Oh, S. Pleas-merging models with permutations and least squares. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2025.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *Neural Information Processing Systems Workshops*, 2011.

Panariello, A., Frascaroli, E., Buzzega, P., Bonicelli, L., Porrello, A., and Calderara, S. Modular embedding recomposition for incremental learning. In *British Machine Vision Conference*, 2025a.

Panariello, A., Marczak, D., Magistri, S., Porrello, A., Twardowski, B., Bagdanov, A. D., Calderara, S., and van de Weijer, J. Accurate and efficient low-rank model merging in core space. In *Advances in Neural Information Processing Systems*, 2025b.

Porrello, A., Buzzega, P., Dangel, F., Sommariva, T., Salami, R., Bonicelli, L., and Calderara, S. Dataless weight disentanglement in task arithmetic via kronecker-factored approximate curvature. *International Conference on Learning Representations*, 2026.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 2020.

Raghu, M., Gilmer, J., Yosinski, J., and Sohl-Dickstein, J. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, 2017.

Rinaldi, F., Capitani, G., Bonicelli, L., Crisostomi, D., Bolelli, F., Ficarra, E., Rodolà, E., Calderara, S., and Porrello, A. Update your transformer to the latest release: Re-basin of task vectors. *International Conference on Machine Learning*, 2025.

Rinaldi, F., Panariello, A., Salici, G., Liu, F., Ciccone, M., Porrello, A., and Calderara, S. Gradient-sign masking for task vector transport across pre-trained models. In *International Conference on Learning Representations*, 2026.

Singh, S. P. and Jaggi, M. Model fusion via optimal transport. In *Advances in Neural Information Processing Systems*, 2020.

Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. The german traffic sign recognition benchmark: a multi-class classification competition. In *IJCNN*, 2011.

Stoica, G., Bolya, D., Bjorner, J. B., Ramesh, P., Hearn, T., and Hoffman, J. Zipit! merging models from different tasks without training. In *International Conference on Learning Representations*, 2024.

Turc, I., Chang, M.-W., Lee, K., and Toutanova, K. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2018.

Williams, A., Nangia, N., and Bowman, S. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018.

Williams, A. H., Kunz, E., Kornblith, S., and Linderman, S. Generalized shape metrics on neural representations. In *Advances in Neural Information Processing Systems*, 2021.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, 2022a.

Wortsman, M., Ilharco, G., Kim, J. W., Li, M., Kornblith, S., Roelofs, R., Lopes, R. G., Hajishirzi, H., Farhadi, A., Namkoong, H., and Schmidt, L. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2022b.

Xiao, J., Ehinger, K. A., Hays, J., Torralba, A., and Oliva, A. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 2016.

Yadav, P., Tam, D., Choshen, L., Raffel, C., and Bansal, M. TIES-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, 2023.

A. Closed-form Transport Rule

Starting from the surrogate objective introduced in the main text,

$$\min_{\tau_B} \left\| H_{\text{in},A} \left(\tau_A^\top - T_{\text{in}} \tau_B^\top T_{\text{out}}^\top \right) H_{\text{out},A}^\top \right\|_F^2, \tag{12}$$

define the residual operator

$$E := \tau_A^\top - T_{\text{in}} \tau_B^\top T_{\text{out}}^\top. \tag{13}$$

The objective can then be rewritten as

$$\min_{\tau_B} \ \| H_{\text{in},A} E H_{\text{out},A}^\top \|_F^2. \tag{14}$$

Under the aligned-subspace approximation induced by Procrustes alignment, the transport objective is minimized when the residual term vanishes, *i.e.*,

$$E = 0. \tag{15}$$

Substituting Eq. (13) gives

$$T_{\text{in}} \tau_B^\top T_{\text{out}}^\top = \tau_A^\top. \tag{16}$$

Multiplying by $T_{\text{in}}^\top$ on the left and T_{out} on the right, and using the row-orthonormality of the Procrustes maps, yields

$$\tau_B^\top = T_{\text{in}}^\top \tau_A^\top T_{\text{out}}, \tag{17}$$

which gives the transport rule

$$\tau_B = T_{\text{out}} \tau_A T_{\text{in}}^\top. \tag{18}$$

This derivation shows that the transported update corresponds to rotating the source task operator through the aligned representation subspaces identified by orthogonal Procrustes analysis.

B. Orthogonality, Pseudo-Inverse Transport, and Alignment Objectives

B.1. Underdetermined Nature of Least-Squares Alignment

A natural approach to solving the transport objective Eq. (4) is to derive a least-squares solution directly from the observed activations. Consider the alignment problem

$$\min_T \|H_A T - H_B\|_F^2, \tag{19}$$

where $H_A \in \mathbb{R}^{N \times d_A}$ and $H_B \in \mathbb{R}^{N \times d_B}$ denote sampled activations from the source and target models.

When H_A is rank-deficient, the problem admits infinitely many equivalent minimizers. Indeed, if $T^\star$ is a solution, then for any perturbation Δ satisfying

$$H_A \Delta = 0, \tag{20}$$

we also have

$$H_A(T^\star + \Delta) = H_A T^\star. \tag{21}$$

Thus, unconstrained least-squares alignment leaves null-space directions unconstrained. These directions can arbitrarily amplify poorly conditioned modes without affecting the reconstruction objective, leading to unstable transport operators and severe norm distortion.

This ambiguity becomes particularly problematic when transporting task updates across models with mismatched widths, since the activation covariances are often low-rank or only partially observed.

B.2. Orthogonal Procrustes as a Regularized Alignment Objective

To regularize the alignment problem, THESEUS restricts the transport maps to have orthonormal rows:

$$TT^\top = I. \tag{22}$$

Expanding the Frobenius objective in Eq. (19),

$$\begin{aligned} \|H_A T - H_B\|_F^2 &= \operatorname{tr}\big((H_A T - H_B)^\top (H_A T - H_B)\big) \\ &= \operatorname{tr}(T^\top H_A^\top H_A T) - 2\operatorname{tr}(T^\top H_A^\top H_B) \\ &\quad + \operatorname{tr}(H_B^\top H_B). \end{aligned}$$

Under the orthogonality constraint,

$$\operatorname{tr}(T^\top H_A^\top H_A T) = \operatorname{tr}(H_A T T^\top H_A^\top) \tag{23}$$

$$= \operatorname{tr}(H_A H_A^\top) = \|H_A\|_F^2, \tag{24}$$

which is constant with respect to T. Therefore, the constrained problem reduces to

$$\max_{T:\, TT^\top = I} \operatorname{tr}(T^\top C), \qquad C = H_A^\top H_B. \tag{25}$$

Let

$$C = U \Sigma V^\top \tag{26}$$

be the singular value decomposition of the cross-covariance matrix. The optimal orthogonal alignment is then given by

$$T^\star = U V^\top. \tag{27}$$

Therefore, orthogonal Procrustes is not introduced as a heuristic assumption that the two representation spaces are exactly isometric. Rather, it emerges naturally as the closed-form optimizer of a regularized alignment problem that removes scaling ambiguities while preserving geometric structure.

B.3. Relation to Pseudo-Inverse Transport

A direct least-squares solution of the transport objective Eq. (4) yields

$$\tau_B^\top = \left(H_{\text{in},B}^\top H_{\text{in},B}\right)^\dagger H_{\text{in},B}^\top G_A H_{\text{out},B} \left(H_{\text{out},B}^\top H_{\text{out},B}\right)^\dagger, \tag{28}$$

where $(\cdot)^\dagger$ denotes the Moore–Penrose pseudo-inverse.

However, this formulation is often numerically unstable in practice. When the two models differ in width, the activation Gram matrices $H_{\text{in},B}^\top H_{\text{in},B}$ and $H_{\text{out},B}^\top H_{\text{out},B}$ are frequently ill-conditioned or rank-deficient, especially when estimated from a finite number of samples.

Our Procrustes-based conditioning step can be interpreted as an orthogonal change of basis that aligns the dominant shared subspaces of the two representation spaces. In the aligned coordinates, the transport becomes well-conditioned and reduces to the closed-form transport rule derived in Sec. A.

Empirically, this distinction is critical: unconstrained least-squares transport suffers from severe norm amplification and unstable performance, whereas orthogonal alignment yields stable transport with near-perfect norm preservation.

C. Experimental setting

Models. In the vision experiments, we employ CLIP models (Radford et al., 2021) with architectures and weights obtained from OpenCLIP (Cherti et al., 2023). We consider three distinct architectural configurations: (i) **Width scaling**: transferring from a standard ViT-B/16 pre-trained on `LAION-2B` to a wider ViT-B/16+ variant pre-trained on `LAION-400M`; (ii) **Same architecture**: transferring between identical ViT-B/16 models pre-trained on different distributions, specifically `DataComp-XL` and `LAION-2B`; (iii) **Width and depth scaling**: transferring across both depth and width from a ViT-B/16 to a ViT-L/14, both pre-trained on `DataComp-XL`. Unless otherwise specified, model A denotes the smaller or source model and model B the larger or target model. For the language experiments, we utilize T5 encoder–decoder models (Raffel et al., 2020) of varying scales, specifically transferring from `T5-3B` to `T5-Large`.

Task Updates. The task-specific updates τ_A are derived by fine-tuning the base model A on the target downstream task via standard supervised objectives. Following the optimization protocol established by Ilharco et al. (2023), we fine-tune for 2000 iterations with a batch size of 128. We employ the AdamW optimizer (Loshchilov & Hutter, 2019) with a peak learning rate of 1×10^{-5}, a weight decay of 0.1, and a cosine annealing schedule preceded by 200 warm-up steps. In accordance with Cherti et al. (2023), the text encoder backbone remains frozen throughout the training process. We then transport τ_A to model B using the proposed method, without performing any additional optimization or gradient-based updates on model B.

Evaluation. For vision tasks, we evaluate classification accuracy on standard benchmarks following the CLIP evaluation protocol. For language tasks, we report accuracy on each dataset. All results are averaged across tasks unless otherwise specified.

D. Computational Complexity Analysis

We compare the computational and memory complexity of THESEUS against gradient-based transport methods such as GradFix as the target width d_B increases. Unlike optimization-based approaches, THESEUS performs a one-shot closed-form transport using only forward activations and covariance accumulation, avoiding repeated backward passes and optimizer-state storage.

Memory Complexity. Gradient-based approaches must retain activations for backpropagation together with gradients, optimizer states, and intermediate tensors, leading to memory usage that grows rapidly with model width. In contrast, THESEUS accumulates cross-covariance statistics in a streaming fashion and does not store full activation tensors across batches.

Specifically, during the forward pass of each calibration mini-batch, the algorithm extracts the source and target activation tensors for each layer l (denoted as H_A^l and H_B^l). Instead of retaining these large tensors in memory, the method immediately computes their uncentered cross-product ($(H_A^l)^\top H_B^l$) alongside the column-wise sums of the activations ($\sum H_A^l$ and $\sum H_B^l$). These lower-dimensional batch-level statistics are added to running accumulators, and the original, memory-intensive activation tensors are detached and discarded.

Once all $\mathcal{B}$ calibration batches have been processed, the final centered cross-covariance matrix is computed algebraically on demand. The global means μ_A and μ_B are derived by dividing the accumulated sums by the total token count M. The exact centered covariance is then constructed via the identity $C^l = (H_A^l)^\top H_B^l - M(\mu_A \otimes \mu_B)$. Because the raw tokens are never concatenated or stored across batches, the maximum memory footprint overhead is strictly bounded by the $\mathcal{O}(d_A \times d_B)$ size of the covariance accumulators, making the memory complexity entirely independent of the total sequence length or the number of calibration batches used.

Runtime Complexity. Using a fixed alignment set of $M = 512$ tokens, THESEUS incurs a one-time alignment cost consisting of activation collection, covariance accumulation,

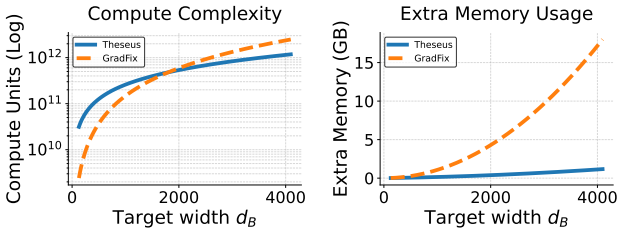

Figure 5. **Computational and memory scaling.** Runtime (left) and additional memory usage (right) as a function of the target model width d_B (with source width $d_A = 768$). While gradient-based fine-tuning methods such as GradFix exhibit steep growth due to backpropagation and activation storage, THESEUS scales substantially better and maintains a nearly flat memory profile thanks to its streaming covariance formulation.

and a single SVD per layer. Gradient-based approaches may be slightly faster at very small widths, but their cost scales more aggressively due to repeated backward passes and iterative optimization.

As shown in Fig. 5, THESEUS exhibits substantially more favorable scaling behavior as target width increases, making it particularly attractive for large-scale target architectures.

E. Additional Experiments

E.1. Sequence Length Alignment Strategies.

To resolve sequence length mismatches between the source θ_A and target θ_B, we evaluate three distinct alignment strategies. These strategies are necessary because different ViT configurations may encode images into different numbers of patches, resulting in distinct token sequence lengths L^A and L^B.

The *mean* strategy serves as a baseline by averaging the feature representations across the sequence dimension, effectively removing spatial granularity. The *interpolation* strategy treats the token sequences as 1D signals and uses linear interpolation to resample the source sequence to the target length. Finally, *interpolation2d* leverages the spatial inductive bias of the Vision Transformer by reshaping the 1D token sequence into a 2D grid that mirrors the original image patch layout. A bilinear interpolation is then applied to this grid to match the target resolution before flattening the result back into a sequence.

As shown in Tab. 5, the *interpolation2d* strategy consistently yields the highest accuracy across all batch sizes, significantly outperforming both the $1D$ linear and mean-based approaches. This emphasizes the importance of preserving the spatial topology of the features: by maintaining the geometric correspondence between patches, the method more effectively transfers the task-specific knowledge distilled by the source model. Notably, these sequence-level adjustments are exclusive to the vision domain. In contrast, for the T5 experiments (Tabs. 2 and 9), no sequence interpolation is necessary. Because T5 natively supports variable-length text sequences, input lengths can be adjusted prior to activation computation to directly match source and target representations, allowing the transport process to be performed without additional preprocessing.

E.2. Extended Results on Width Scaling.

We provide a comprehensive evaluation of the $(A \to B)$ transport direction under varying $\mathcal{K}$-shot settings in Tab. 6, illustrating how performance scales with the number of labeled samples per class. Complementary results for the reverse transport direction $(B \to A)$ are detailed in Tabs. 7 and 8, using both batch-based $\mathcal{B}$-shot and class-balanced $\mathcal{K}$-shot protocols. Collectively, these findings confirm that THESEUS is robust across diverse sampling strategies and consistently yields improvements, even when the target model already exhibits a strong zero-shot baseline.

E.3. Language.

Following the experimental setting of Rinaldi et al. (2025), we consider the model $\theta = \{\phi, \omega\}$ composed of a pre-trained feature extractor ϕ and a classification head ω. We transport the task vector τ_A^{ϕ} from model A (`T5-3B`) to the target feature extractor ϕ_B (`T5-Large`). We then refrain from training a new classifier for the target model and instead reuse the source fine-tuned head ω_{ft}^A. This allows us to evaluate whether the transported representation space aligns with the source model's classification logic without additional head optimization. As baselines, we report the performance of the target backbone θ_B equipped with a randomly initialized classification head, alongside a fully fine-tuned version of θ_B where both the encoder and head are optimized for the target task. As shown in Table 9, the gains are modest but consistent across different values of $\mathcal{B}$, and THESEUS outperforms the baseline θ_B in every setting.

Summarization Task. As shown in Tab. 10, THESEUS can also be applied to the summarization task. `Flan-T5` was fine-tuned on the CNN/DailyMail dataset (Sec. F.2) using the same hyperparameters as in the other experiments. The resulting task vector was then transported to `T5-v1.1`. Performance was evaluated using the ROUGE-L metric (Lin, 2004), which measures the longest common subsequence overlap between the generated and reference summaries.

Table 5. **Ablation on Interpolation Modes.** Performance across datasets for different values of $\mathcal{B}$ and interpolation strategies. Results are reported as accuracy (%).

Model	$\mathcal{B}$	Interpolation	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*	–	–	56.69	62.66	55.46	68.51	58.19	88.81	76.47	72.42	67.40 (+0.00)
θ_B *fine-tune*	–	–	98.66	97.70	98.77	95.40	83.29	92.43	99.64	79.91	93.23 (+25.83)
THESEUS	1	mean	59.60	63.21	55.47	68.45	58.33	88.89	79.00	72.53	68.19 (+0.79)
		interpolate	57.96	62.91	56.88	68.75	58.85	88.92	79.33	72.49	68.23 (+0.83)
		interpolate2d	66.31	76.55	63.57	70.33	60.00	88.86	88.14	72.67	**73.30 (+5.90)**
THESEUS	2	mean	58.31	62.96	56.72	68.73	58.39	88.83	79.10	72.47	68.44 (+1.04)
		interpolate	62.94	63.33	57.13	69.03	58.97	88.98	78.66	72.64	68.96 (+1.56)
		interpolate2d	70.00	73.86	68.38	72.59	59.95	88.85	88.64	72.75	**74.38 (+6.98)**
THESEUS	5	mean	62.33	63.20	58.16	68.89	59.41	88.88	80.94	72.88	69.34 (+1.94)
		interpolate	62.31	64.25	58.79	69.71	59.81	89.08	82.27	72.77	69.88 (+2.48)
		interpolate2d	71.14	79.66	69.78	72.27	62.02	88.95	93.88	73.01	**76.34 (+8.94)**

Table 6. **Width Scaling (Narrow → Wide)** Cross-dataset performance (A: `laion2b` → B: `laion400m`). Results are reported for varying support set sizes $\mathcal{K}$

Model	$\mathcal{K}$	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*	–	50.92	39.23	49.63	64.53	55.48	84.53	57.06	68.67	58.76 (+0.00)
θ_B *fine-tune*	–	98.96	91.08	98.63	92.59	77.81	87.65	99.63	76.76	90.39 (+31.63)
θ_A *fine-tune*	–	98.69	97.45	98.64	95.65	82.24	91.53	99.61	79.86	92.96 (+34.20)
THESEUS	1	60.79	55.73	57.39	67.48	57.34	85.02	61.67	70.76	**64.52 (+5.76)**
THESEUS	2	61.46	56.07	59.03	68.01	58.56	85.06	68.19	70.81	**65.90 (+7.14)**
THESEUS	5	64.58	55.20	59.05	69.18	59.25	85.17	77.57	70.93	**67.62 (+8.86)**
THESEUS	10	65.51	56.86	59.72	69.48	60.48	85.16	76.26	71.28	**68.09 (+9.33)**
THESEUS	20	66.83	60.33	58.25	70.03	60.53	85.51	78.38	72.12	**69.00 (+10.24)**

F. Dataset Details

F.1. Visual Datasets.

We evaluate our method on a diverse suite of vision classification benchmarks, covering a wide range of domains from natural images to satellite imagery:

- **EuroSAT:** A remote sensing dataset based on Sentinel-2 satellite imagery, comprising 27 000 geo-referenced samples across 10 distinct classes (Helber et al., 2019).
- **SVHN:** A digit recognition benchmark containing 73 257 real-world images of house numbers across 10 classes extracted from Google Street View (Netzer et al., 2011).
- **GTSRB:** The German Traffic Sign Recognition Benchmark, a standard for autonomous driving tasks featuring 51 839 images across 43 traffic sign categories (Stallkamp et al., 2011).
- **RESISC45:** A high-resolution remote sensing dataset containing 31 500 images from Google Earth, categorized into 45 scene classes to evaluate fine-grained aerial classification (Cheng et al., 2017).
- **DTD:** The Describable Textures Dataset, consisting of 5640 texture-centric images organized into 47 perceptual categories (Cimpoi et al., 2014).
- **Stanford Cars:** A fine-grained classification dataset comprising 16 185 images across 196 categories of cars, distinguished by make, model, and year (Krause et al., 2013).
- **MNIST:** The Modified National Institute of Standards and Technology database, a foundational benchmark for handwritten digit recognition containing 70 000 grayscale images across 10 classes (Lecun et al., 1998).
- **SUN397:** A large-scale scene understanding benchmark consisting of 108 754 images covering 397 categories, ranging from indoor environments to diverse outdoor landscapes (Xiao et al., 2016).

F.2. Textual Datasets

- **SNLI**: Stanford Natural Language Inference dataset, containing 570 000 sentence pairs labeled for entailment, contradiction, or neutral (Bowman et al., 2015).
- **MNLI**: The Multi-Genre Natural Language Inference dataset contains 433 000 sentence pairs annotated with

Table 7. **Width Scaling (Wide → Narrow)** Cross-dataset performance (A: `laion400m` → B: `datacomp_xl`). Results are reported for varying support set sizes $\mathcal{K}$

Model	$\mathcal{K}$	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*		56.69	62.66	55.46	68.51	58.19	88.81	76.47	72.42	67.40 (+0.00)
θ_B *fine-tune*		98.66	97.70	98.77	95.40	83.29	92.43	99.64	79.91	93.23 (+25.83)
THESEUS	1	62.95	73.56	66.99	71.03	60.16	88.79	83.31	73.18	**72.50** (+5.10)
	2	68.01	74.19	67.19	70.89	60.53	88.91	88.15	73.26	**73.89** (+6.49)
	5	68.35	75.92	71.20	72.68	62.13	88.92	92.25	73.22	**75.58** (+8.18)
	10	73.16	76.93	67.63	73.44	62.93	88.89	91.98	73.21	**76.02** (+8.62)
	20	70.84	82.82	70.63	73.89	63.62	89.03	94.33	73.38	**77.32** (+9.92)

Table 8. **Width Scaling (Wide → Narrow)** Cross-dataset performance (A: `laion400m` → B: `datacomp_xl`). Results for varying number of alignment batches B.

Model	$\mathcal{B}$	EUROSAT	SVHN	GTSRB	RESISC45	DTD	CARS	MNIST	SUN397	AVG (ΔAcc)
θ_B *zero-shot*		56.69	62.66	55.46	68.51	58.19	88.81	76.47	72.42	67.40 (+0.00)
θ_B *fine-tune*		98.66	97.70	98.77	95.40	83.29	92.43	99.64	79.91	93.23 (+25.83)
THESEUS	1	66.31	76.55	63.57	70.33	60.00	88.86	88.14	72.67	**73.30** (+5.90)
THESEUS	2	70.00	73.86	68.38	72.59	59.95	88.85	88.64	72.75	**74.38** (+6.98)
THESEUS	5	71.14	79.66	69.78	72.27	62.02	88.95	93.88	73.01	**76.34** (+8.94)
THESEUS	10	71.75	79.73	69.58	72.33	62.25	89.00	94.12	73.16	**76.49** (+9.09)
THESEUS	20	72.36	80.33	68.58	74.01	62.61	88.97	94.30	73.24	**76.80** (+9.40)

textual entailment information across various genres (Williams et al., 2018).

- **RTE**: Recognizing Textual Entailment dataset, with 2490 examples for training, 277 for validation, and 3000 for testing, divided into two classes (Wang et al., 2018).
- **QNLI**: The Question Natural Language Inference dataset contains 104 743 training examples across two classes (Wang et al., 2018).
- **SCITAIL**: A science entailment dataset built from science question answering, with 23 596 training examples of two classes (Khot et al., 2018).
- **CNN/DailyMail**: An English-language dataset containing more than 300k news articles written by journalists at CNN and the Daily Mail, paired with summaries (Hermann et al., 2015; Nallapati et al., 2016).

G. Functional Transport vs. Feature Alignment

As observed in Sec. 4.2 (Tab. 1), unconstrained pseudo-inverse baselines (τ_{pinv}, $\tau_{\text{pinv-Tikh}}$) consistently underperform THESEUS , particularly in low-shot regimes where the activation statistics become severely rank-deficient. In this section, we analyze the source of this instability and isolate the contributions of both our coupled functional objective (Eq. (4)) and the orthogonal constraints from standard latent-space feature alignment approaches (Moschella et al., 2023; Maiorca et al., 2023).

To do so, we ablate six alternative transport constructions across two fundamental axes: whether they optimize a coupled operator objective or decoupled feature maps, and whether they are unconstrained or orthogonally regularized (Tab. 11).

1. Coupled Unconstrained Transport

These baselines share THESEUS 's functional matching objective, attempting to find a target update τ_B that directly minimizes the bilinear interaction mismatch:

- **Pinv / Pinv-Tikh.** Direct least-squares solutions to the coupled objective. As derived in Sec. B.3, the unconstrained Moore–Penrose pseudo-inverse yields:

$$\tau_B^\top = (H_{\text{in},B}^\top H_{\text{in},B})^\dagger H_{\text{in},B}^\top (H_{\text{in},A} \tau_A^\top H_{\text{out},A}^\top) \times H_{\text{out},B} (H_{\text{out},B}^\top H_{\text{out},B})^\dagger. \tag{29}$$

  The Tikhonov variant adds ridge regularization (λI) to the Gram matrices prior to inversion to improve numerical stability.

2. Decoupled Feature Alignment

These baselines abandon the coupled operator objective. Instead, they estimate independent linear maps R_{in} and

Table 9. **Width Scaling (Wide → Narrow).** Cross-dataset performance for transferred task-specific encoder representations (A: `T5-3B` → B: `T5-Large`). Results are reported for varying numbers of alignment batches $\mathcal{B}$.

Model	$\mathcal{B}$	MNLI	QNLI	RTE	SCITAIL	SNLI	AVG (ΔAcc)
θ_B	–	32.23	48.05	48.38	50.77	32.44	42.37 (+0.00)
θ_B *fine-tune*	–	89.80	94.27	83.59	95.86	91.83	91.09 (+48.72)
THESEUS	20	57.01	50.56	49.46	50.77	57.24	**53.01** (+10.64)
THESEUS	50	54.52	56.31	50.12	53.45	60.42	**54.96** (+12.59)
THESEUS	100	55.66	59.49	50.98	63.11	64.89	**58.83** (+16.46)

Table 10. ROUGE-L scores for summarization transfer. Source: `Flan-T5` (A), target: `T5-v1.1` (B). THESEUS represents $\theta_B + \tau_B$. Results are reported for varying numbers of alignment samples $\mathcal{D}$.

Configuration	α	$\mathcal{D}=200$	$\mathcal{D}=500$	$\mathcal{D}=800$	Average
θ_B	–		0.1053		
θ_B *fine-tune*	–		0.2821		
THESEUS	0.5	0.1173	0.1201	0.1265	0.1213
THESEUS	1.0	0.1278	0.1337	0.1437	0.1351
THESEUS	2.0	0.1237	0.1189	0.1423	0.1283
THESEUS	4.0	0.0876	0.0512	0.0879	0.0756

R_{out} to translate the input and output feature spaces, and form the update via $\tau_B = R_{\text{out}} \tau_A R_{\text{in}}^\top$.

- **Linear.** Standard unconstrained least-squares alignment for each space independently:

$$R_{\text{in}}^{\text{lin}} = (H_{\text{in},A}^\top H_{\text{in},A})^\dagger (H_{\text{in},A}^\top H_{\text{in},B}). \tag{30}$$

- **Ortho.** Orthogonal Procrustes alignment computed directly from the cross-covariance $H_{\text{in},A}^\top H_{\text{in},B} = U\Sigma V^\top$, yielding the orthogonal map $R_{\text{in}}^{\text{ortho}} = UV^\top$.
- **Linear-Ortho / Pinv-Ortho.** Orthogonal projections of the unconstrained *Linear* and *Pinv* feature maps onto the nearest orthogonal matrix through SVD. This mirrors the standard pipeline in prior latent-space translation methods, where an unconstrained alignment is first estimated and subsequently orthogonalized.

3. Coupled Constrained Transport (THESEUS)

Our proposed formulation combines the coupled functional objective with strict orthogonal constraints. This reduces to aligning the input and output cross-covariances ($C_{\text{in}} = H_{\text{in},A}^\top H_{\text{in},B}$, $C_{\text{out}} = H_{\text{out},A}^\top H_{\text{out},B}$), producing the closed-form Procrustes embeddings $T_{\text{in}} = U_{\text{in}} V_{\text{in}}^\top$ and $T_{\text{out}} = U_{\text{out}} V_{\text{out}}^\top$, and the transported update $\tau_B = T_{\text{out}} \tau_A T_{\text{in}}^\top$.

The results in Tab. 11 reveal the distinct failure modes of the baselines.

The necessity of geometric constraints. Unconstrained solutions to the coupled objective (*Pinv*) exhibit severe norm amplification, particularly in low-shot settings where the activation covariances become poorly conditioned. Null-space directions are arbitrarily amplified, yielding massive norm distortion ($\|\tau_B\|_F \gg \|\tau_A\|_F$). While Tikhonov regularization (*Pinv-Tikh*) partially alleviates this by damping high-gain directions, it introduces a sensitive hyperparameter and still distorts the update geometry.

The necessity of the coupled objective. Enforcing orthogonality (*Ortho*, *Linear-Ortho*, *Pinv-Ortho*) resolves the instability. Because orthogonal maps preserve norms and angles, they eliminate scaling ambiguities and maintain $\|\tau_B\|_F \approx \|\tau_A\|_F$. However, despite this stabilization, all orthogonal feature-alignment baselines remain consistently below THESEUS . This gap reveals that independent alignment of input and output representations is insufficient to guarantee preservation of the functional effect of the task update. There is no mathematical guarantee that composing independently optimal feature maps preserves the bilinear interaction structure encoded by $H_{\text{in}} \tau^\top H_{\text{out}}^\top$. Furthermore, for two-step variants like *Pinv-Ortho* and *Linear-Ortho*, computing an unconstrained mapping first and then projecting it onto the orthogonal group means orthogonalizing an already corrupted matrix: the initial inversion step amplifies noise in rank-deficient directions, locking in a suboptimal rotation.

In contrast, THESEUS succeeds because it enforces geometric stability while deriving the transport rule directly from the coupled functional objective, effectively bypassing ill-conditioned intermediate inversions entirely.

Table 11. **Alignment Method Ablation.** Accuracy and norm-preservation error ($\|\tau_B\|_F - \|\tau_A\|_F$) for different transport operators in the ViT-B/16 $\rightarrow$ ViT-B/16+ setting.

$\mathcal{B}$	Method	Accuracy (%)	Norm Error
2	τ_{pinv}	52.25	6.30×10^{4}
	$\tau_{\text{pinv-Tikh}}$	59.60	2.30×10^{-4}
	τ_{ortho}	63.31	2.40×10^{1}
	τ_{linear}	4.52	1.15×10^{4}
	$\tau_{\text{linear-ortho}}$	60.93	-4.02×10^{-9}
	$\tau_{\text{pinv-ortho}}$	60.47	3.18×10^{-11}
	THESEUS	**66.60**	$\mathbf{7.60 \times 10^{-10}}$
5	τ_{pinv}	56.80	4.70×10^{4}
	$\tau_{\text{pinv-Tikh}}$	60.29	1.10×10^{-4}
	τ_{ortho}	64.23	1.90×10^{1}
	τ_{linear}	4.53	7.83×10^{3}
	$\tau_{\text{linear-ortho}}$	61.64	7.78×10^{-9}
	$\tau_{\text{pinv-ortho}}$	60.51	6.82×10^{-11}
	THESEUS	**66.64**	$\mathbf{8.40 \times 10^{-10}}$
10	τ_{pinv}	58.39	3.80×10^{4}
	$\tau_{\text{pinv-Tikh}}$	60.75	3.90×10^{-7}
	τ_{ortho}	65.44	1.70×10^{1}
	τ_{linear}	4.69	6.27×10^{3}
	$\tau_{\text{linear-ortho}}$	62.77	3.21×10^{-9}
	$\tau_{\text{pinv-ortho}}$	66.75	-1.90×10^{-10}
	THESEUS	**68.35**	$\mathbf{7.90 \times 10^{-10}}$