BEYOND REPRESENTATION DRIFT: DECOMPOSED RELATIVE GEOMETRY PRESERVATION FOR CONTINUAL VISION-LANGUAGE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual adaptation of contrastive vision-language models leads to catastrophic forgetting, accompanied by substantial drift in their visual and textual representations. Existing methods mitigate forgetting by constraining changes in model parameters, feature representations, or cross-modal affinities, yet they do not explicitly separate changes that preserve the joint image–text geometry from those that alter it. We find that this distinction is substantial in practice: across a seven-task continual sequence, nearly half of the accumulated representation drift is attributable to a shared orthogonal transformation that leaves image–text retrieval unchanged. Motivated by this observation, we propose Decomposed Relative Geometry Preservation (Decomp-RGP), which measures drift after factoring out shared orthogonal transformation through closed-form Procrustes alignment and exactly decomposes the remaining residual into visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch. We further introduce a budget-based adaptive weighting scheme to control the contributions of these components in fine-tuning process. Across seven image–text retrieval tasks, Decomp-RGP improves Mean R@1 by 5.32% over the strongest baseline, while maintaining stronger zero-shot performance across multiple CLIP backbones.

1 INTRODUCTION

Real-world image–text data continually arrive from new domains and tasks, requiring vision-language models to continually adapt to changing data distributions (Zhu et al., 2023; Yu et al., 2024). This requires sufficient plasticity to acquire new knowledge while maintaining the stability of previously learned representations and cross-modal relationships (Wang et al., 2024). However, sequential fine-tuning can progressively disrupt previously learned representations, leading to catastrophic forgetting (French, 1999; Rebuffi et al., 2017; Zheng et al., 2023; Wu et al., 2025). The key challenge is therefore to preserve prior knowledge without restricting the representational flexibility required for continual adaptation.

Existing methods approach this challenge by preserving prior knowledge in different ways. Rehearsal-based methods mitigate this by replaying past data (Hayes et al., 2019; Yoon et al., 2022), but storing and repeatedly processing historical samples introduces additional memory and computational overhead (Khan et al., 2023; Smith et al., 2023). Rehearsal-free methods address forgetting without retaining past examples (Volpi et al., 2021; Liu & Chang, 2025). Recent approaches preserve prior knowledge through constraints on model parameters (Kirkpatrick et al., 2017), feature representations or projection spaces (Liu et al., 2025), or cross-modal relational structures (Ni et al., 2023; Cui et al., 2024; Huang et al., 2025). Although these methods differ in what they preserve, they largely treat representation change itself as something to be constrained, leaving a more fundamental question open: *Which representation changes should be constrained during continual adaptation?*

Shared Orthogonal Invariance. Representation displacement does not necessarily imply a change in the joint image–text geometry. When visual and textual embeddings undergo the same orthogonal transformation, all pairwise similarities and retrieval rankings remain unchanged, whereas divergent modality-specific transformations do not preserve this invariance. Figs. 1(a)-(b) empirically validate

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

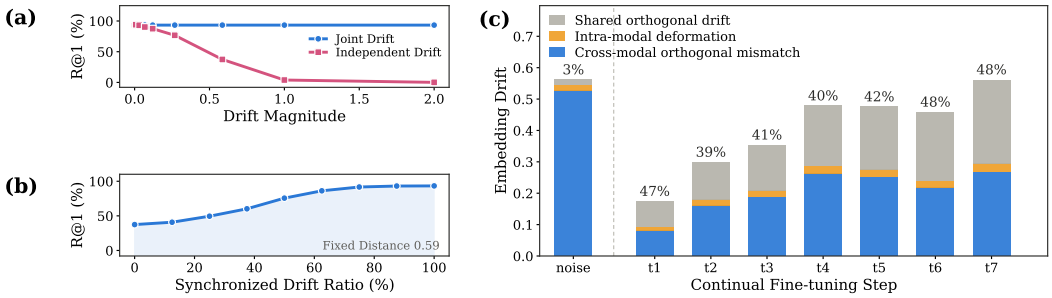

Figure 1: **Representation drift contains a large geometry-preserving component ignored by existing continual learning objectives.** (a). Shared orthogonal drift preserves retrieval, whereas matched independent drift degrades it. (b). At fixed drift magnitude, retrieval improves as the visual and textual transformations become synchronized. (c). Drift during continual fine-tuning decomposed into shared orthogonal drift, cross-modal alignment mismatch, and intra-modal deformation; percentages denote the shared-rotation fraction, with matched Gaussian noise as a control.

this distinction on CLIP, showing that shared orthogonal drift preserves retrieval regardless of its magnitude, while independent drift of matched magnitude degrades retrieval performance.

Prevalence of shared orthogonal drift. A substantial fraction of the representation drift observed during continual fine-tuning is attributable to a shared orthogonal transformation. We quantify this fraction by fitting a single orthogonal transformation between the pretrained and current representations and measuring how much of the total displacement it explains. Across the seven continual tasks in Fig. 1(c), this proportion consistently ranges from 39% to 48%. To test whether this shared component reflects structured representation drift, we apply the same decomposition to independent Gaussian perturbations matched to the total displacement at the final continual step, where about 3% of the drift can be explained by a shared orthogonal transformation. This contrast indicates that the the substantial shared component observed during continual fine-tuning is a systematic property of the learned drift rather than a consequence of random displacement. Thus, geometry-preserving drift is not a theoretical corner case, but a substantial component of real continual adaptation. Existing representation-preserving objectives do not explicitly factor out this geometry-preserving shared component before constraining the remaining drift.

This motivates a different principle: *representation changes explained by a shared geometry-preserving transformation should be factored out before the remaining drift is constrained*. We formalize this principle with Relative Geometry Preservation (RGP), which measures representation drift after removing the best shared orthogonal transformation through closed-form Procrustes alignment. The remaining drift contains distinct intra-modal and cross-modal geometric changes. Decomposed Relative Geometry Preservation (Decomp-RGP) exactly decomposes the residual drift into visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch. This decomposition enables separate control of the three residual components. Since their magnitudes vary across tasks and training, we introduce budget-based adaptive weighting to maintain prescribed contributions under a shared regularization budget.

Our main contributions are as follows.

- We show that substantial continual representation drift corresponds to a shared orthogonal transformation that preserves image–text retrieval, indicating that indiscriminate constraints penalize geometry-preserving changes.
- We propose Decomposed Relative Geometry Preservation (Decomp-RGP), which factors out shared orthogonal drift and decomposes the residual into visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch for selective regularization.
- We introduce budget-based adaptive weighting to control the contributions of the decomposed drift components during continual adaptation. Across seven image–text retrieval tasks, Decomp-RGP achieves a 5.32% gain in Mean R@1 over the strongest baseline while better preserving zero-shot capability across multiple CLIP backbones.

2 Related Work

2.1 Continual Learning for Vision-Language Models

Continual learning seeks to acquire new tasks while preserving prior knowledge (Kirkpatrick et al., 2017). Existing approaches mainly rely on replay, parameter-efficient adaptation, or regularization. Replay methods revisit historical samples but require additional memory and computation (Rebuffi et al., 2017; Chaudhry et al., 2019; Yoon et al., 2021; Chen et al., 2026a; Bako & Kalita, 2026). Parameter-efficient methods preserve prior knowledge by restricting adaptation to lightweight modules, but may limit plasticity (Liang & Li, 2024; Liu et al., 2025; Chen et al., 2026b; Ye et al., 2026). Regularization instead constrains changes in model parameters, feature representations, or cross-modal structures while retaining full-parameter adaptation (Wen et al., 2024), particularly relevant to vision-language models, where continual adaptation must preserve both modality-specific information and cross-modal alignment.

Recent continual vision-language methods preserve prior knowledge through different forms of representation and cross-modal regularization. Mod-X preserves prior cross-modal alignment by distilling the off-diagonal information of contrastive matrices (Ni et al., 2023). while DKR filters and rectifies incorrect image–text affinities during knowledge distillation (Cui et al., 2024). LPI models cross-modal and cross-task interactions through low-rank prompt learning (Yan et al., 2024), while MG-CLIP preserves and compensates for the modality gap during continual adaptation (Huang et al., 2025). C-CLIP consolidates historical cross-modal knowledge through contrastive learning in a projection space (Liu et al., 2025). Despite preserving different aspects of prior knowledge and cross-modal structure, it remains unclear which components of representation drift preserve the joint image-text geometry and which disrupt it during continual adaptation.

2.2 Representation Geometry and Orthogonal Alignment

Orthogonal transformations preserve pairwise inner products and Euclidean distances, leaving the underlying representation geometry unchanged (Kornblith et al., 2019). Orthogonality has also been used in continual learning, where OGD (Farajtabar et al., 2020) and GPM (Saha et al., 2021) restrict new-task gradient updates to directions orthogonal to past-task gradient subspaces, reducing interference with prior knowledge. ORTCL uses orthogonal transformations for continual model editing by aligning new features with the existing feature space while preserving metric structure (Lin et al., 2026). However, these works use orthogonality as a learning constraint or alignment mechanism, whereas the role of naturally occurring orthogonal drift during continual vision-language adaptation remains unexplored.

3 Approach

Continual adaptation inevitably induces representation changes in vision-language models, but the effect of such changes depends on their geometric nature. Representation displacement does not necessarily imply a loss of learned cross-modal structure: some changes preserve the joint image-text geometry, whereas others alter cross-modal relationships that determine retrieval behavior. Existing preservation objectives maintain prior representations or cross-modal relations (Kirkpatrick et al., 2017; Ni et al., 2023; Cui et al., 2024; Liu et al., 2025), but do not explicitly identify whether drift arises from geometry-preserving motion or geometry-altering deformation. We therefore formulate continual drift as a geometric decomposition problem, where geometry-preserving motion is factored out before selectively constraining residual drift that modifies cross-modal structure.

We instantiate our formulation in CLIP-based vision-language models, where each image-text pair is mapped into a shared embedding space. Given a batch of N image-text pairs, the visual and textual encoders produce two sets of normalized embeddings, denoted as $V = \{v_1, v_2, \ldots, v_N\}$ and $T = \{t_1, t_2, \ldots, t_N\}$, where $v_i, t_i \in \mathbb{R}^d$. We concatenate them as a joint representation matrix $X = [V; T] \in \mathbb{R}^{2N \times d}$, where $V, T \in \mathbb{R}^{N \times d}$ represent visual and textual embeddings, respectively.

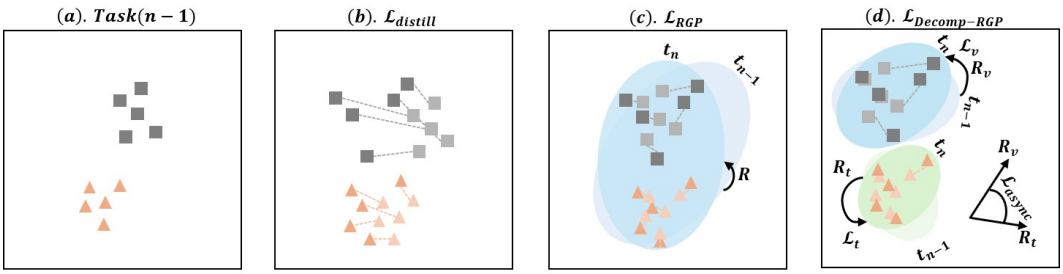

Figure 2: **Geometric view of representation drift and its regularization.** **(a)** The visual and textual embedding space of the previous model contains squares and triangles denoting visual and textual embeddings, respectively. **(b)** Point-wise L_2 feature distillation penalizes the displacement of each embedding from its previous position. **(c)** RGP finds a shared orthogonal transformation R that best aligns the previous visual and textual representations with the current ones, leaving only the residual drift to be penalized. **(d)** Decomp-RGP separates this residual into visual deformation ($\mathcal{L}_v$), textual deformation ($\mathcal{L}_t$), and cross-modal orthogonal alignment mismatch ($\mathcal{L}_{\text{async}}$), allowing the three components to be regularized independently.

3.1 Geometry-Preserving Drift in Joint Embedding Space

In contrastive vision-language models, knowledge is encoded by the relative geometry between visual and textual embeddings rather than their absolute coordinates, so representation displacement does not necessarily imply a change in the learned cross-modal structure. During continual adaptation, we denote the joint embeddings from the previous and current continual states as $X_o, X_n \in \mathbb{R}^{2N\times d}$, we define geometry-preserving drift as a shared orthogonal transformation $X_n = X_o R$, where $R^\top R = I$. Under this transformation, the joint embedding geometry remains unchanged:

$$X_n X_n^\top = X_o R R^\top X_o^\top = X_o X_o^\top. \tag{1}$$

Thus, all pairwise similarities used by CLIP retrieval are preserved, even though individual embeddings may move substantially in the representation space. This indicates that displacement magnitude alone is insufficient to measure harmful drift. Motivated by this observation, we factor out geometry-preserving motion and focus on the residual drift that modifies cross-modal structure. A complete derivation is given in Appendix A.2.

3.2 Relative Geometry Preservation

Although the joint similarity geometry is unchanged under a shared orthogonal transformation, the embeddings themselves may still move substantially in the ambient space. This distinction is not captured by point-wise representation matching. The standard L_2 feature distillation directly constrains the displacement between the previous and current representations. As illustrated in Fig. 2(b), the standard L_2 feature distillation constrains each visual and textual embedding to remain close to its previous position. Let $v_o^i, v_n^i \in \mathbb{R}^d$ and $t_o^i, t_n^i \in \mathbb{R}^d$ denote the ℓ_2-normalized visual and textual embeddings of the i-th image–text pair, where the subscript o denotes the frozen previous-task model and n denotes the current model being optimized. Stacking these embeddings row-wise over the N pairs in a mini-batch gives the representation matrices $V_o, V_n \in \mathbb{R}^{N\times d}$ and $T_o, T_n \in \mathbb{R}^{N\times d}$. We define $M_v = V_o^\top V_n \in \mathbb{R}^{d\times d}$, $M_t = T_o^\top T_n \in \mathbb{R}^{d\times d}$, and $M = M_v + M_t \in \mathbb{R}^{d\times d}$. Then,

$$\mathcal{L}_{\text{distill}} = \frac{1}{2N}\sum_{i=1}^{N}\left(\left\|v_n^i - v_o^i\right\|_2^2 + \left\|t_n^i - t_o^i\right\|_2^2\right) = 2 - \frac{\operatorname{tr}(M)}{N}, \tag{2}$$

where N denotes the number of image-text pairs. Since each term measures the displacement of an individual embedding, any nonzero displacement is penalized even when the relative geometry is preserved. A detailed derivation of Eq. 2 is provided in Appendix A.3.

Recent studies have gone beyond direct representation matching by preserving similarities or relational structures between previous and current representations (Ni et al., 2023; Cui et al., 2024; Liu et al., 2025). However, they do not explicitly distinguish the geometry-preserving transformation from changes that alter the joint embedding geometry. When the visual and textual embeddings undergo the same orthogonal transformation, their cross-modal similarities and relative geometry remain unchanged, and so do the retrieval rankings. Unlike Eq. 2, which penalizes representation displacement directly, we introduce the Relative Geometry Preservation (RGP) loss to factor out such shared orthogonal motion and measure only the residual drift. As shown in Fig. 1(c), a substantial fraction of the drift under unconstrained continual fine-tuning can be explained by this geometry-preserving component. Accordingly, we define RGP as the minimum residual displacement after aligning the previous representations to the current ones by a shared orthogonal transformation as

$$\mathcal{L}_{\mathrm{RGP}} = \min_{R \in O(d)} \frac{1}{2N} \sum_{i=1}^{N} \left(\left\| v_n^i - v_o^i R \right\|_2^2 + \left\| t_n^i - t_o^i R \right\|_2^2 \right) = 2 - \frac{\sum_{k=1}^{d} \sigma_k(M)}{N}, \tag{3}$$

where $\sigma_k(M)$ denotes the k-th singular value of M. The RGP loss searches for a single orthogonal transformation R that jointly aligns the previous visual and textual embeddings with their current representations. Representation changes explained by this shared transformation incur no penalty, while the remaining misalignment contributes to the loss. Therefore, $\mathcal{L}_{\mathrm{RGP}}$ is vanished when the visual and textual representations can be jointly related by a shared orthogonal transformation, and becomes positive when residual geometric changes remain after this alignment. By factoring out the geometry-preserving shared motion, RGP avoids penalizing harmless drift and allows greater plasticity during continual adaptation. The detailed derivation is provided in Appendix A.4.

3.3 Decomposed Relative Geometry Preservation

The residual drift after shared alignment captures how the two modalities deviate from a common geometric evolution. However, these deviations may originate from different sources, including modality-specific deformation and cross-modal alignment mismatch. We therefore further decompose the RGP objective into three components: visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch. To disentangle them, we allow modality-specific alignments R_v and R_t. Their residuals define the visual and textual deformation terms L_v and L_t, while the additional residual incurred by enforcing a shared alignment defines the cross-modal orthogonal alignment mismatch L_{async}. This ambiguity arises because a single orthogonal matrix R is used to align both modalities. We therefore decouple the shared alignment into modality-specific orthogonal alignments $R_v, R_t \in O(d)$:

$$\left\| v_n^i - v_o^i R_v \right\|_2^2, \qquad \left\| t_n^i - t_o^i R_t \right\|_2^2. \tag{4}$$

When $R_v = R_t$, the visual and textual embeddings undergo the same orthogonal motion, and the cross-modal similarity structure is preserved. Allowing modality-specific alignments isolates the residual deformation within each modality, while comparing their summed residual with the shared-alignment residual quantifies the cross-modal orthogonal alignment mismatch. Following the same Procrustes formulation as RGP, the visual and textual deformation terms are

$$\mathcal{L}_v = 1 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_v), \qquad \mathcal{L}_t = 1 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_t). \tag{5}$$

These two terms measure the residual deformation after the best modality-specific orthogonal motion is removed from the visual and textual embeddings, respectively. The complete derivation of $\mathcal{L}_v$ and $\mathcal{L}_t$ is in Appendix A.5.

However, RGP fits a single shared orthogonal transformation to both modalities. The additional residual relative to separate modality-specific alignments quantifies the disagreement between visual and textual orthogonal alignments. We define this quantity as the cross-modal orthogonal alignment mismatch:

$$\mathcal{L}_{\mathrm{async}} \triangleq \mathcal{L}_{\mathrm{RGP}} - (\mathcal{L}_{\mathrm{v}} + \mathcal{L}_{\mathrm{t}}) = \frac{1}{N} \left[\sum_{k=1}^{d} \sigma_k(M_v) + \sum_{k=1}^{d} \sigma_k(M_t) - \sum_{k=1}^{d} \sigma_k(M_v + M_t) \right] \geq 0. \tag{6}$$

The resulting decomposition separates the RGP residual into visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch. The complete derivation is provided in Appendix A.6. Fig. 1(b) shows retrieval depends on intra-modal synchronization, not displacement magnitude, which is exactly what $\mathcal{L}_{\text{async}}$ measures, while $\mathcal{L}_v$ and $\mathcal{L}_t$ capture only benign within-modality deformation once realigned. We therefore expect $\mathcal{L}_{\text{async}}$ to dominate the regularization budget, confirmed empirically in Section 4.4 ($\rho_a = 75\%$).

3.4 Budget-Based Adaptive Weighting

The three regularization terms exhibit different scales that vary across tasks in the training stages, making fixed weighting unreliable. Consequently, fixed regularization weights do not maintain a consistent balance between adaptation and representation preservation. Instead, we control the contribution of each component through a target budget.

We assign the target budget ratios for visual deformation, textual deformation, and cross-modal orthogonal alignment mismatch, ρ_v, ρ_t, and ρ_a, respectively, and define the total regularization budget by

$$\rho = \rho_v + \rho_t + \rho_a, \qquad 0 \le \rho < 1. \tag{7}$$

Therefore, the regularization terms are assigned a target fraction ρ of the total objective, while the remaining fraction $1-\rho$ is reserved for the CLIP contrastive loss. Importantly, ρ_v, ρ_t and ρ_a specify the target loss contributions rather than the fixed regularization weights.

Since individual mini-batch losses can fluctuate, we compute the adaptive weights from exponentially smoothed loss magnitudes. Let μ_{CLIP}, μ_v, μ_t and μ_a denote the exponential moving averages of $\mathcal{L}_{\text{CLIP}}$, $\mathcal{L}_v$, $\mathcal{L}_t$ and $\mathcal{L}_{\text{async}}$, respectively. At mini-batch s,

$$\mu_j^{(s)} = \beta \mu_j^{(s-1)} + (1-\beta)\mathcal{L}_j^{(s)}, \qquad j \in \{\text{CLIP}, v, t, a\}, \tag{8}$$

where $\mathcal{L}_a \equiv \mathcal{L}_{\text{async}}$ and $\beta \in [0, 1)$ controls the smoothing strength. For the first mini-batch of each task, the EMA statistics are initialized from the corresponding losses, with $\mu_j^{(1)} = \mathcal{L}_j^{(1)}$, and are reinitialized at the beginning of each new task. Let $S^{(s)}$ denote the target scale of the total objective at mini-batch s. Since the CLIP loss is assigned the fraction $1-\rho$, we set

$$S^{(s)} = \frac{\mu_{\text{CLIP}}^{(s)}}{1-\rho}. \tag{9}$$

For each drift component $x \in \{v, t, a\}$, its weighted contribution is required to match the assigned budget ρ_x. This gives

$$\lambda_x^{(s)} \mu_x^{(s)} = \rho_x S^{(s)} \quad \Longrightarrow \quad \lambda_x^{(s)} = \frac{\rho_x}{1-\rho} \frac{\mu_{\text{CLIP}}^{(s)}}{\mu_x^{(s)}}, \qquad x \in \{v, t, a\}. \tag{10}$$

At the beginning of a new task, the current model and its frozen reference are identical, making the three drift terms zero or close to zero. Directly applying Eq. 10 would therefore produce unstable weights due to the small denominator. We introduce an activation threshold $\epsilon = 10^{-3}$. A regularization component remains inactive until its EMA first exceeds ϵ. Once activated, it remains active until the end of the current task. Accordingly,

$$\lambda_x^{(s)} = \begin{cases} 0, & \text{before } \mu_x^{(s)} \text{ first exceeds } \epsilon, \\ \dfrac{\rho_x S^{(s)}}{\max\left(\mu_x^{(s)}, \epsilon\right)}, & \text{after activation,} \end{cases} \qquad x \in \{v, t, a\}. \tag{11}$$

Finally, the resulting training objective at mini-batch s is

$$\mathcal{L}_{\text{total}}^{(s)} = \mathcal{L}_{\text{CLIP}}^{(s)} + \underbrace{\lambda_v^{(s)}\mathcal{L}_v^{(s)} + \lambda_t^{(s)}\mathcal{L}_t^{(s)} + \lambda_a^{(s)}\mathcal{L}_{\text{async}}^{(s)}}_{\mathcal{L}_{\text{Decomp-RGP}}^{(s)}}. \tag{12}$$

Both the EMA statistics and the activation states are reset at the beginning of each task. EMA values are used only to determine adaptive weights, whereas gradients are propagated through the current mini-batch losses in $\mathcal{L}_{\text{total}}$, where $\mathcal{L}_{\text{CLIP}}$ denotes the standard CLIP InfoNCE loss (Radford et al., 2021). The complete algorithm is provided in Appendix A.7.

Table 1: Continual image–text retrieval (Mean R@1). All methods are trained on seven domains in sequence without access to previous data. Results report per-task R@1 after the final training stage.

Dataset	CL	Flickr30K	COCO	Kream	VG	Wikido	Lexica	SUN397	Average
Mean R@1									
CLIP	✗	72.20	33.73	49.13	39.90	43.71	39.92	25.39	43.42
CLIP	✓	73.08	39.08	46.12	49.56	39.18	48.33	**38.36**	47.67
EWC	✓	82.11	42.45	59.72	52.97	48.61	47.96	32.37	52.31
Mod-X	✓	78.97	42.27	54.36	52.16	49.18	48.45	31.69	51.01
LPI	✓	80.96	39.72	58.58	58.47	50.16	58.57	30.87	53.88
DKR	✓	81.99	44.25	58.58	57.39	50.76	52.73	33.59	54.19
C-CLIP	✓	79.73	42.74	48.46	54.77	45.11	49.65	33.28	50.53
MG-CLIP	✓	81.58	44.43	55.50	57.72	53.10	59.95	34.75	55.29
Ours	✓	**83.89**	**46.63**	**70.64**	**65.28**	**55.08**	**66.05**	36.72	**60.61**

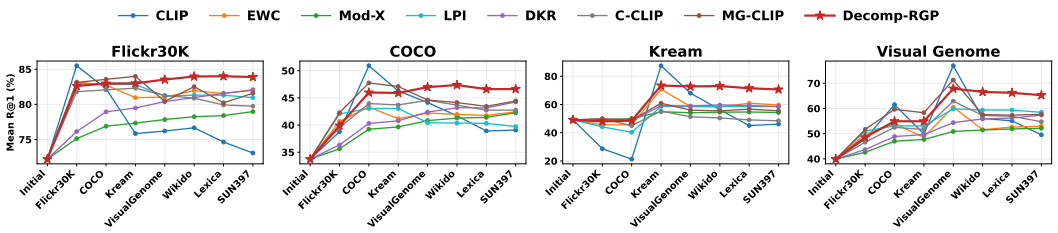

Figure 3: Mean R@1 on Flickr30K, COCO, Kream, and Visual Genome tracked across the seven-task training sequence.

4 Analysis of Continual Adaptation

4.1 Experimental Setup

All experiments are conducted on a single NVIDIA H200 GPU with 141 GB of memory. Input images are resized to 224×224 pixels, and each task is trained for 10 epochs with an effective batch size of 1024. We use AdamW with a learning rate of 1×10^{-5} and weight decay of 0.2, using the same optimization configuration across all tasks without task-specific tuning. Full training details are provided in Appendix A.1.

For continual retrieval, we construct a seven-task sequence consisting of Flickr30K (Plummer et al., 2015), COCO (Chen et al., 2015), Kream (anonhs), Visual Genome (Krishna et al., 2017), Wikido (Kalyan et al., 2024), Lexica (Shen et al., 2024), and SUN397 (Xiao et al., 2010). To evaluate whether continual adaptation preserves transferable knowledge beyond the training sequence, we further perform zero-shot classification on ImageNet-1K (Russakovsky et al., 2015), Stanford Cars (Krause et al., 2013), Stanford Flowers (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), DTD (Cimpoi et al., 2014), and RSICB256 (Li et al., 2020). We compare Decomp-RGP with EWC (Kirkpatrick et al., 2017), Mod-X (Ni et al., 2023), LPI (Yan et al., 2024), DKR (Cui et al., 2024), C-CLIP (Liu et al., 2025), and MG-CLIP (Huang et al., 2025). For fair comparison, all methods use the same CLIP ViT-B/16 backbone and follow the same continual training and evaluation protocol.

4.2 Analysis of Continual Vision-Language Retrieval

Fig. 3 tracks per-task Mean R@1 across the seven-task sequence. Without any preservation mechanism, CLIP's Mean R@1 on Flickr30K drops from above 85% to 73.08%, illustrating severe catastrophic forgetting under distributional shift. Decomp-RGP consistently achieves the highest plasticity across all four datasets shown, with especially large margins on Kream and Visual Genome where competing methods plateau early. As summarized in Table 1, Decomp-RGP outperforms the strongest baseline (MG-CLIP) by 5.32% in average Mean R@1 and ranks first on every dataset ex-

Table 2: Component ablation on the seven-task retrieval sequence. Values in parentheses indicate improvement over the pretrained CLIP baseline (first row).

Method	Weighted Budget	ACC (%)↑	BWT (%)↑
Pretrained CLIP (no training)	✗	43.42	—
Fine-tune, w/o regularization	✗	47.67 (+4.25)	−20.40
+ Feature distillation ($\lambda = 40$)	✗	51.03 (+7.61)	+0.35
+ Feature distillation ($\Sigma\rho = 40\%$)	✓	52.12 (+8.70)	**+0.46**
+ RGP: factor out shared orthogonal component ($\Sigma\rho = 40\%$)	✓	53.82 (+10.40)	+0.19
+ Decomp-RGP: decompose into $\mathcal{L}_v/\mathcal{L}_t/\mathcal{L}_{\text{async}}$ ($\lambda = 40$)	✗	60.17 (+16.75)	−1.38
+ Decomp-RGP: adaptive budget $\rho_v : \rho_t : \rho_a = 5:5:30$ ($\Sigma\rho = 40\%$)	✓	60.21 (+16.79)	−0.58
+ Decomp-RGP: asymmetric $\rho_v : \rho_t : \rho_a = 8:2:30$ ($\Sigma\rho = 40\%$)	✓	**60.61 ±0.11 (+17.19)**	−0.75

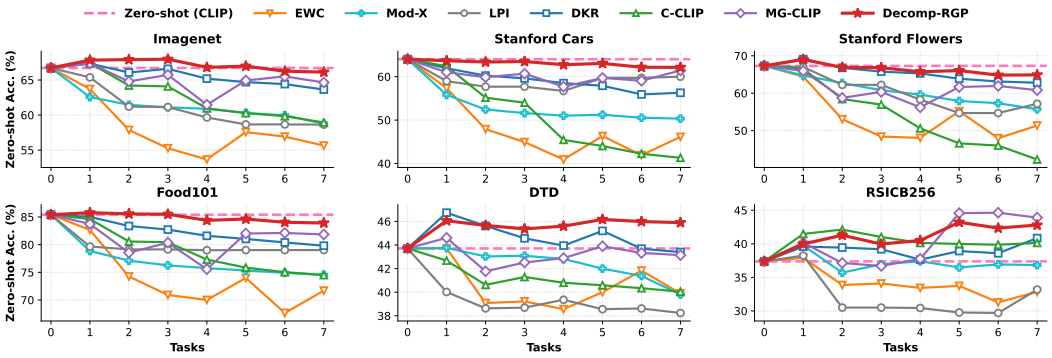

Figure 4: Zero-shot classification performance across seven continual learning tasks on six datasets.

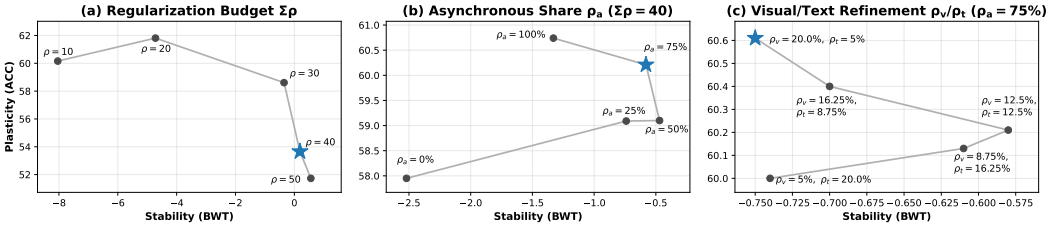

Figure 5: Stability–plasticity trade-off under different regularization budgets and allocations.

cept SUN397, with gains of 15.14% over MG-CLIP on Kream and 10.51% over C-CLIP on Visual Genome. The exception on SUN397 is expected because it is the final task in the sequence, where unconstrained fine-tuning retains full plasticity at the cost of forgetting all previous tasks. Overall, Decomp-RGP preserves stability on learned tasks without sacrificing plasticity on new ones, outperforming all baselines in both dimensions.

4.3 Analysis of Zero-Shot Generalization

Fig. 4 compares the zero-shot classification performance of Decomp-RGP with existing continual vision-language methods. Decomp-RGP consistently exhibits stronger resistance to catastrophic forgetting, retaining most of the original zero-shot capability of pretrained CLIP across the evaluated datasets. Notably, on DTD, its zero-shot accuracy after learning all seven tasks exceeds that of the strongest competing method by approximately 2%, demonstrating stronger generalization to an unseen classification task. On the remaining datasets, Decomp-RGP incurs only minor forgetting throughout the continual sequence. These results suggest that preserving relative cross-modal geometry helps retain transferable pre-trained structure while accommodating new knowledge, thereby better preserving zero-shot generalization.

Table 3: Evaluation of Decomp-RGP across CLIP backbones ($\rho_v : \rho_t : \rho_a = 8:2:30$, $\Sigma\rho = 40\%$).

	Retrieval		Zero-shot Classification					
Backbone	I2T R@1	T2I R@1	ImageNet	Cars	Flowers	Food101	DTD	RSICB256
ViT-B/32	56.85	54.58	61.39	55.69	58.71	77.95	40.98	38.99
ViT-B/16	61.28	59.94	66.13	62.14	64.90	83.91	45.90	42.79
ViT-L/14	67.68	66.13	72.04	71.29	67.86	87.28	54.96	43.08
ViT-L/14@336px	68.20	67.63	72.17	71.49	66.44	87.47	54.84	44.70

4.4 Ablation on the Stability–Plasticity Trade-off

The pretrained CLIP model exhibits different initial loss scales across datasets, making fixed regularization weights unreliable during early adaptation. We therefore use budget-based adaptive weighting to maintain a controlled regularization share throughout training. As shown in Fig. 5(a), we select a total regularization budget of 40%, where BWT is close to zero while plasticity remains high. Under this budget, Fig. 5(b) shows that assigning 75% to the asynchronous term gives the best stability–plasticity trade-off. Fig. 5(c) further shows that stronger visual preservation improves retrieval accuracy with only minor changes in BWT. We therefore allocate 20%, 5%, and 75% of the regularization budget to the visual, textual, and asynchronous terms, respectively, corresponding to final loss shares of 8%, 2%, and 30%.

4.5 Ablation of Preservation Strategies

Fixed vs. Budget-Based Weighting. Table 2 compares different preservation strategies. We first apply our budget-based adaptive weighting to standard ℓ_2 feature distillation. When the regularization coefficient is fixed at 40, feature distillation yields lower BWT than its 40% budget-based counterpart, suggesting that a single fixed coefficient can provide insufficient preservation on some tasks. At the same time, its relatively low ACC indicates that the same coefficient can over-constrain other tasks and suppress plasticity. This shows the difficulty of maintaining a consistent stability–plasticity balance with a fixed regularization strength.

Undecomposed vs. Decomposed RGP. Factoring out orthogonal embedding transformation and penalizing only drift unexplained by shared orthogonal alignment, RGP alone improves ACC by approximately 1.7% over budget-based feature distillation. Decomposing RGP further improves plasticity via separate control of visual, textual, and cross-modal alignment mismatch, while adaptive weighting alleviates the BWT degradation seen under fixed weighting. The final Decomp-RGP trades a slightly lower BWT than feature distillation for a much larger plasticity gain, raising ACC from 52.12% to 60.61% with only ± 0.11 variation across three random seeds, 41, 42, and 43.

We evaluate Decomp-RGP across CLIP backbones in Table 3. The method remains effective from ViT-B/32 to ViT-L/14@336px, with larger backbones yielding stronger continual retrieval and zero-shot classification. This shows that Decomp-RGP generalizes across backbone scales and input resolutions. As a robustness check, Appendix A.8 verifies that mini-batch Procrustes estimates reliably approximate the task-level shared rotation rather than reflecting batch-sampling noise: task-level orthogonal drift explains 69.15% of the representation drift on average, with 93.86% average agreement between batch-wise and task-level estimates (Table 5).

5 Conclusion

We proposed Decomp-RGP for continual vision–language retrieval. It factors out the geometry-preserving shared orthogonal component and selectively regularizes the residual intra-modal deformation and cross-modal orthogonal alignment mismatch. Budget-based adaptive weighting balances these terms across tasks and training stages. This preserves representational flexibility while constraining geometry-altering changes. Across a seven-task continual sequence, Decomp-RGP consistently improves retrieval over existing methods. It achieves higher plasticity with minimal forgetting while retaining strong zero-shot performance across CLIP backbones.

AI USE STATEMENT

In this work, we used generative AI tools for polish the writing of this manuscript. We have not used generative AI tools for generating experimental data or dataset preparation, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for debugging segments of code. We have reviewed all AI-assisted work. We take responsibility for the final content of this work.

ETHICS STATEMENT

This work does not involve human subjects and does not raise concerns related to potentially harmful insights, discrimination, bias, fairness, legal compliance, or research integrity. All datasets used in this study are publicly available, and we do not collect or process private user data. Our proposed method is not intended for safety-critical or other high-risk applications, and is designed solely for continual model adaptation and iterative model updating.

REPRODUCIBILITY STATEMENT

All derivations and experimental configurations are provided in the appendix. We release the experimental logs and source code in the supplementary material. All datasets used in this work are publicly available, and we will provide the official download links together with the complete metadata files.

REFERENCES

anonhs. Kream clothes dataset. Hugging Face Datasets. URL `https://huggingface.co/datasets/anonhs/kream-clothes-dataset`.

John Bako and Jugal Kalita. Addressing catastrophic forgetting in class-incremental learning—a survey. *Artificial Intelligence Review*, 2026.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019.

Jianting Chen, Dianzhi Yu, and Irwin King. Smart replay: Adaptive scheduling of memory rehearsal for computational resource-aware incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39945–39961, 2026a.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Yitian Chen, Shigeng Zhang, Xuan Liu, Mingming Lu, Kai Chen, Hongye Zhu, and Xinning Chen. Parameter-efficient continual learning for enhancing plasticity without forgetting under limited model capacity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10789–10798, 2026b.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

Zhenyu Cui, Yuxin Peng, Xun Wang, Manyu Zhu, and Jiahuan Zhou. Continual vision-language retrieval via dynamic knowledge rectification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11704–11712, 2024.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International conference on artificial intelligence and statistics*, pp. 3762–3773. PMLR, 2020.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Tyler L Hayes, Nathan D Cahill, and Christopher Kanan. Memory efficient experience replay for streaming learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 9769–9776. IEEE, 2019.

Linlan Huang, Xusheng Cao, Haori Lu, Yifan Meng, Fei Yang, and Xialei Liu. Mind the gap: Preserving and compensating for the modality gap in clip-based continual learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3777–3786. IEEE, 2025.

T P Kalyan, Piyush S Pasi, Sahil N Dharod, Azeem A Motiwala, Preethi Jyothi, Aditi Chaudhary, and Krishna Srinivasan. Wikido: A new benchmark evaluating cross-modal retrieval for vision-language models. *Advances in Neural Information Processing Systems*, 37:140812–140827, 2024.

Muhammad Gul Zain Ali Khan, Muhammad Ferjad Naeem, Luc Van Gool, Didier Stricker, Federico Tombari, and Muhammad Zeshan Afzal. Introducing language guidance in prompt-based continual learning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11429–11439. IEEE, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMlR, 2019.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

Haifeng Li, Xin Dou, Chao Tao, Zhixiang Wu, Jie Chen, Jian Peng, Min Deng, and Ling Zhao. Rsi-cb: A large-scale remote sensing image classification benchmark using crowdsourced data. *Sensors*, 20(6):1594, 2020.

Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23638–23647. IEEE, 2024.

Li Lin, Xinrui Zhang, Qi Zhang, Shuai Wang, and Kaiwen Xia. Ortcl: Towards continual learning of time series foundation models on streaming data via orthogonal rotation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23541–23549, 2026.

Wenzhuo Liu, Fei Zhu, Longhui Wei, and Qi Tian. C-clip: Multimodal continual learning for vision-language model. In *International Conference on Learning Representations*, volume 2025, pp. 46461–46477, 2025.

Xuan Liu and Xiaobin Chang. Lora subtraction for drift-resistant space in exemplar-free continual learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15308–15318. IEEE, 2025.

Zixuan Ni, Longhui Wei, Siliang Tang, Yueting Zhuang, and Qi Tian. Continual vision-language representation learning with off-diagonal information. In *International Conference on Machine Learning*, pp. 26129–26149. PMLR, 2023.

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*, pp. 722–729. IEEE, 2008.

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 *Proceedings of the IEEE international conference on computer vision*, pp. 2641–2649, 2015.

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 *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 5533–5542. IEEE, 2017.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=3AOj0RCNC2`.

Xinyue Shen, Yiting Qu, Michael Backes, and Yang Zhang. Prompt Stealing Attacks Against Text-to-Image Generation Models. In *USENIX Security Symposium (USENIX Security)*. USENIX, 2024.

James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11909–11919. IEEE, 2023.

Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4441–4451. IEEE, 2021.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024.

Haitao Wen, Lili Pan, Yu Dai, Heqian Qiu, Lanxiao Wang, Qingbo Wu, and Hongliang Li. Class incremental learning with multi-teacher distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28443–28452. IEEE, 2024.

Bin Wu, Wuxuan Shi, Jinqiao Wang, and Mang Ye. Synthetic data is an elegant gift for continual vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2813–2823. IEEE, 2025.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3485–3492, 2010. doi: 10.1109/CVPR.2010.5539970.

Weicai Yan, Ye Wang, Wang Lin, Zirun Guo, Zhou Zhao, and Tao Jin. Low-rank prompt interaction for continual vision-language retrieval. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 8257–8266, 2024.

Fei Ye, YongCheng Zhong, Qihe Liu, Adrian G Bors, JingLing Sun, JinYu Guo, and ShiJie Zhou. Learning adaptive and expandable mixture model for continual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27773–27781, 2026.

Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. *arXiv preprint arXiv:2106.01085*, 2021.

Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/pdf?id=f9D-5WNG4Nv.

Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23219–23230. IEEE, 2024.

Zangwei Zheng, Mingyuan Ma, Kai Wang, Ziheng Qin, Xiangyu Yue, and Yang You. Preventing zero-shot transfer degradation in continual learning of vision-language models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19068–19079. IEEE, 2023.

Hongguang Zhu, Yunchao Wei, Xiaodan Liang, Chunjie Zhang, and Yao Zhao. Ctp: Towards vision-language continual pretraining via compatible momentum contrast and topology preservation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22200–22210. IEEE, 2023.

A APPENDIX

A.1 IMPLEMENTATION DETAILS

All benchmark methods strictly follow the common experimental settings as shown in Table 4, while method-specific hyperparameters are kept consistent with their original implementations. A learning rate of 1×10^{-5} is used for all methods, except C-CLIP, for which we follow the original setting and use a text-encoder learning rate $10\times$ larger than that of the image encoder. For all methods, we avoid dataset-specific hyperparameter tuning. This ensures a fair comparison under a unified protocol and better reflects practical continual-learning scenarios.

Table 4: Implementation details used in the main experiments.

Common Settings	
CLIP backbone	ViT-B/16
Input resolution	224×224
Training epochs	10 per task
Batch size	512
Gradient accumulation	2
Effective batch size	1024
Optimizer	AdamW
Learning rate	1×10^{-5}
Visual / text learning-rate ratio	$1:1$
Weight decay	0.2
AdamW coefficients	$\beta_1 = 0.9,\ \beta_2 = 0.98,\ \epsilon = 1 \times 10^{-6}$
Learning-rate schedule	Linear warm-up followed by cosine decay
Warm-up	1 epoch
Gradient clipping	1.0
Mixed precision	FP16
Random seed	42
Number of workers	8
Decomp-RGP Specific Settings	
Total regularization budget $\Sigma\rho$	40%
Visual / textual / asynchronous allocation	$20\%/5\%/75\%$ of $\Sigma\rho$
ρ_v, ρ_t, ρ_a	$8\%/2\%/30\%$
EMA coefficient	0.95
Drift threshold / denominator floor ϵ	10^{-3}

A.2 ORTHOGONAL INVARIANCE OF THE JOINT EMBEDDING GEOMETRY

Let $X_o, X_n \in \mathbb{R}^{2N\times d}$ denote the previous and current joint visual-textual embeddings, with Gram matrices $S_o = X_o X_o^\top$ and $S_n = X_n X_n^\top$. If $X_n = X_o R$ for some $R \in O(d)$, then

$$S_n = X_n X_n^\top = X_o R R^\top X_o^\top = X_o X_o^\top = S_o. \tag{13}$$

Since $X = [V; T]$, equality of the joint Gram matrices preserves both within-modality and cross-modal similarities.

For the converse, assume $X_n X_n^\top = X_o X_o^\top$. Let $X_o = U\Sigma Q_o^\top$ be a thin SVD with $r = \mathrm{rank}(X_o)$. The columns of U and Q_o are orthonormal, and Σ contains the r nonzero singular values. Hence,

$$X_o X_o^\top = U\Sigma^2 U^\top. \tag{14}$$

The Gram equality therefore implies that X_n has the same nonzero singular values and the same left singular subspace as X_o. Define

$$Q_n = X_n^\top U\Sigma^{-1}. \tag{15}$$

Using $X_n X_n^\top = U\Sigma^2 U^\top$, it follows that $Q_n^\top Q_n = I_r$ and $X_n = U\Sigma Q_n^\top$. Thus,

$$X_o = U\Sigma Q_o^\top, \qquad X_n = U\Sigma Q_n^\top, \tag{16}$$

so the two embeddings differ only in their right orthonormal factors. Extend Q_o and Q_n to full orthogonal bases $\widetilde{Q}_o, \widetilde{Q}_n \in O(d)$, and set

$$R = \widetilde{Q}_o \widetilde{Q}_n^\top. \tag{17}$$

By construction, $R^\top R = I$, and the shared factors above give

$$X_o R = U\Sigma Q_n^\top = X_n. \tag{18}$$

The above result shows that two joint embedding configurations have the same Gram matrix exactly when their difference can be represented by a shared orthogonal transformation. Equivalently,

$$X_n X_n^\top = X_o X_o^\top \quad \Longleftrightarrow \quad \exists\, R \in O(d) \text{ such that } X_n = X_o R. \tag{19}$$

A.3 Distillation Derivation

Starting from Eq. 2, the standard L_2 feature distillation objective is

$$\mathcal{L}_{\text{distill}} = \frac{1}{2N} \sum_{i=1}^{N} \left(\left\| v_n^i - v_o^i \right\|_F^2 + \left\| t_n^i - t_o^i \right\|_F^2 \right). \tag{20}$$

Using $\|A\|_F^2 = \operatorname{tr}(A^\top A)$, Eq. 20 can be expanded as

$$\begin{aligned} \mathcal{L}_{\text{distill}} &= \frac{1}{2N} \left[\operatorname{tr}\big((V_n - V_o)^\top (V_n - V_o)\big) + \operatorname{tr}\big((T_n - T_o)^\top (T_n - T_o)\big) \right] \\ &= \frac{1}{2N} \left[\operatorname{tr}(V_n^\top V_n) + \operatorname{tr}(V_o^\top V_o) - 2\operatorname{tr}(V_o^\top V_n) \right. \\ &\qquad \left. + \operatorname{tr}(T_n^\top T_n) + \operatorname{tr}(T_o^\top T_o) - 2\operatorname{tr}(T_o^\top T_n) \right]. \end{aligned} \tag{21}$$

Since all visual and textual embeddings are ℓ_2-normalized, $\operatorname{tr}(V_n^\top V_n) = \operatorname{tr}(V_o^\top V_o) = \operatorname{tr}(T_n^\top T_n) = \operatorname{tr}(T_o^\top T_o) = N$. Therefore,

$$\mathcal{L}_{\text{distill}} = 2 - \frac{1}{N} \left[\operatorname{tr}(V_o^\top V_n) + \operatorname{tr}(T_o^\top T_n) \right]. \tag{22}$$

Let $M_v = V_o^\top V_n$ and $M_t = T_o^\top T_n$, and define $M = M_v + M_t$. The distillation objective can then be written as

$$\mathcal{L}_{\text{distill}} = 2 - \frac{\operatorname{tr}(M)}{N}. \tag{23}$$

Assume that the previous and current representations are related by a shared orthogonal transformation, $V_n = V_o R$ and $T_n = T_o R$, where $R^\top R = R R^\top = I$. For any image i and text j,

$$\langle v_n^i, t_n^j \rangle = (v_o^i R)(t_o^j R)^\top = v_o^i R R^\top (t_o^j)^\top = \langle v_o^i, t_o^j \rangle. \tag{24}$$

The same argument gives $V_n V_n^\top = V_o V_o^\top$ and $T_n T_n^\top = T_o T_o^\top$. Thus, the shared transformation preserves both cross-modal and intra-modal similarities. However, substituting the same transformation into Eq. 20 gives

$$\mathcal{L}_{\text{distill}} = \frac{1}{2N} \left(\|V_o(R - I)\|_F^2 + \|T_o(R - I)\|_F^2 \right), \tag{25}$$

where, I corresponds to leaving the embedding coordinates unchanged. Therefore, even though an orthogonal R preserves the complete similarity geometry, Eq. 25 assigns a positive penalty whenever R moves the visual or textual embeddings.

A.4 Orthogonal Procrustes Formulation of RGP

As established in Appendix A.2, a shared orthogonal transformation preserves the joint similarity geometry of the visual and textual embeddings. RGP therefore searches for the orthogonal transformation that best aligns the previous representations with the current ones. The displacement

explained by this alignment corresponds to geometry-preserving shared motion, while the remaining residual captures the change that cannot be explained by such a transformation. Recall the RGP objective from Eq. 3:

$$\begin{aligned}
\mathcal{L}_{\text{RGP}} &= \min_{R\in O(d)} \frac{1}{2N}\sum_{i=1}^{N}\left(\left\|v_n^i - v_o^i R\right\|_2^2 + \left\|t_n^i - t_o^i R\right\|_2^2\right) \\
&= \min_{R\in O(d)} \frac{1}{2N}\sum_{i=1}^{N}\Big[\|v_n^i\|_2^2 + \|v_o^i R\|_2^2 - 2v_n^i (v_o^i R)^\top \\
&\qquad\qquad + \|t_n^i\|_2^2 + \|t_o^i R\|_2^2 - 2t_n^i (t_o^i R)^\top\Big] \\
&= \min_{R\in O(d)} \frac{1}{2N}\sum_{i=1}^{N}\Big[2 - 2v_n^i R^\top (v_o^i)^\top + 2 - 2t_n^i R^\top (t_o^i)^\top\Big] \\
&= 2 - \frac{1}{N}\max_{R\in O(d)} \sum_{i=1}^{N}\left[v_n^i R^\top (v_o^i)^\top + t_n^i R^\top (t_o^i)^\top\right] \\
&= 2 - \frac{1}{N}\max_{R\in O(d)} \operatorname{tr}\left[R^\top\left(\sum_{i=1}^{N}(v_o^i)^\top v_n^i + \sum_{i=1}^{N}(t_o^i)^\top t_n^i\right)\right] \\
&= 2 - \frac{1}{N}\max_{R\in O(d)} \operatorname{tr}\big(R^\top M\big)
\end{aligned} \tag{26}$$

where $M = M_v + M_t$, $M_v = \sum_{i=1}^{N}(v_o^i)^\top v_n^i$, and $M_t = \sum_{i=1}^{N}(t_o^i)^\top t_n^i$. The last equality follows from the orthogonal Procrustes solution. To solve the remaining orthogonal optimization, take the singular value decomposition $M = U\Sigma W^\top$, where $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_d)$. Then

$$\max_{R\in O(d)} \operatorname{tr}(R^\top M) = \max_{R\in O(d)} \operatorname{tr}(W^\top R^\top U\Sigma). \tag{27}$$

To isolate the contribution of the singular values, we rewrite the orthogonal factors as $Z = W^\top R^\top U$. Because the product of orthogonal matrices is orthogonal, $Z \in O(d)$, and the objective reduces to

$$\max_{Z\in O(d)} \operatorname{tr}(Z\Sigma) = \max_{Z\in O(d)} \sum_{k=1}^{d} z_{kk}\sigma_k. \tag{28}$$

Because $|z_{kk}| \le 1$ and $\sigma_k \ge 0$, the maximum is attained at $Z = I$. Hence,

$$\max_{R\in O(d)} \operatorname{tr}(R^\top M) = \sum_{k=1}^{d}\sigma_k(M), \qquad R^\star = UW^\top. \tag{29}$$

Substituting Eq. 29 into Eq. 26 gives

$$\mathcal{L}_{\text{RGP}} = 2 - \frac{1}{N}\sum_{k=1}^{d}\sigma_k(M). \tag{30}$$

A.5 Derivation of the Modality-Specific RGP Terms

Recall the RGP objective from Eq. 3:

$$\mathcal{L}_{\text{RGP}} = \min_{R\in O(d)} \frac{1}{2N}\sum_{i=1}^{N}\left(\left\|v_n^i - v_o^i R\right\|_2^2 + \left\|t_n^i - t_o^i R\right\|_2^2\right). \tag{31}$$

RGP uses a single orthogonal matrix R to align both modalities. The shared alignment is decomposed into two modality-specific orthogonal alignments, parameterized by R_v and R_t:

$$\min_{R_v, R_t\in O(d)} \frac{1}{2N}\sum_{i=1}^{N}\left(\left\|v_n^i - v_o^i R_v\right\|_2^2 + \left\|t_n^i - t_o^i R_t\right\|_2^2\right). \tag{32}$$

Since R_v and R_t are optimized independently, the objective separates into two modality-specific terms:

$$\begin{aligned}\mathcal{L}_v &= \min_{R_v \in O(d)} \frac{1}{2N} \sum_{i=1}^{N} \left\| v_n^i - v_o^i R_v \right\|_2^2 = 1 - \frac{1}{N} \max_{R_v \in O(d)} \operatorname{tr}(R_v^\top M_v) = 1 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_v), \\ \mathcal{L}_t &= \min_{R_t \in O(d)} \frac{1}{2N} \sum_{i=1}^{N} \left\| t_n^i - t_o^i R_t \right\|_2^2 = 1 - \frac{1}{N} \max_{R_t \in O(d)} \operatorname{tr}(R_t^\top M_t) = 1 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_t),\end{aligned} \tag{33}$$

The resulting $\mathcal{L}_v$ and $\mathcal{L}_t$ quantify the residual deformation that remains after removing the best orthogonal transformation within each modality.

A.6 Derivation of the Asynchronous Drift Term

The modality-specific alignments allow the visual and textual branches to use separate orthogonal transformations R_v and R_t. Their feasible set is therefore $O(d) \times O(d)$. In contrast, RGP requires both modalities to share the same transformation, restricting the feasible solutions to pairs of the form (R, R). Since the shared-alignment solutions form a subset of the independently aligned solutions, the shared RGP residual cannot be smaller than the sum of the two modality-specific residuals:

$$\mathcal{L}_{\text{RGP}} \geq \mathcal{L}_v + \mathcal{L}_t. \tag{34}$$

The difference between the two objectives measures the additional residual introduced by enforcing a common orthogonal alignment across the visual and textual representations. We define this gap as the asynchronous drift term:

$$\begin{aligned}\mathcal{L}_{\text{async}} &= \mathcal{L}_{\text{RGP}} - (\mathcal{L}_v + \mathcal{L}_t) \\ &= \left(2 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_v + M_t)\right) - \left(2 - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_v) - \frac{1}{N} \sum_{k=1}^{d} \sigma_k(M_t)\right) \\ &= \frac{1}{N} \left[\sum_{k=1}^{d} \sigma_k(M_v) + \sum_{k=1}^{d} \sigma_k(M_t) - \sum_{k=1}^{d} \sigma_k(M_v + M_t)\right].\end{aligned} \tag{35}$$

This term is non-negative because the sum of singular values satisfies the triangle inequality. A zero value indicates that the shared alignment introduces no additional residual beyond the two modality-specific alignments, whereas a positive value indicates disagreement between the orthogonal transformations required by the visual and textual representations.

A.7 ALGORITHM

Algorithm 1 Decomp-RGP: Continual Learning with Decomposed Relative Geometry Preservation

Require: Task sequence $\{\mathcal{D}_1, \ldots, \mathcal{D}_T\}$; pretrained CLIP model θ_0; budgets ρ_v, ρ_t, ρ_a; EMA factor β; drift tolerance ϵ

Ensure: Continually adapted model θ_T

for task $t = 1, \ldots, T$ **do**
  $\theta_{\text{ref}} \leftarrow \theta_{t-1}$ frozen reference model
  for mini-batch s with $(I, T) \in \mathcal{D}_t$ **do**
    $V_n, T_n \leftarrow f_{\theta_t}(I, T)$; $V_o, T_o \leftarrow f_{\theta_{\text{ref}}}(I, T)$
    $M_v \leftarrow V_o^\top V_n$; $M_t \leftarrow T_o^\top T_n$
    $\mathcal{L}_v^{(s)} \leftarrow 1 - \frac{1}{N}\sum_i \sigma_i(M_v)$
    $\mathcal{L}_t^{(s)} \leftarrow 1 - \frac{1}{N}\sum_i \sigma_i(M_t)$
    $\mathcal{L}_{\text{async}}^{(s)} \leftarrow \frac{1}{N}\left[\sum_i \sigma_i(M_v) + \sum_i \sigma_i(M_t) - \sum_i \sigma_i(M_v + M_t)\right]$
    $\mathcal{L}_{\text{CLIP}}^{(s)} \leftarrow \text{InfoNCE}(V_n, T_n)$
    Initialize $\mu_j^{(1)} \leftarrow \mathcal{L}_j^{(1)}$ for $j \in \{\text{CLIP}, v, t, a\}$ at the first mini-batch; otherwise $\mu_j^{(s)} \leftarrow \beta\mu_j^{(s-1)} + (1-\beta)\mathcal{L}_j^{(s)}$
    $\rho \leftarrow \rho_v + \rho_t + \rho_a$; $S^{(s)} \leftarrow \mu_{\text{CLIP}}^{(s)}/(1-\rho)$
    Compute
    for $x \in \{v, t, a\}$ **do**
      if $\mu_x^{(r)} \le \epsilon$ for all $r \le s$ **then**
        $\lambda_x^{(s)} \leftarrow 0$
      else
        $$\lambda_x^{(s)} \leftarrow \frac{\rho_x S^{(s)}}{\max\left(\mu_x^{(s)}, \epsilon\right)}$$
      end if
    end for
    $\mathcal{L}_{\text{total}}^{(s)} \leftarrow \mathcal{L}_{\text{CLIP}}^{(s)} + \lambda_v^{(s)}\mathcal{L}_v^{(s)} + \lambda_t^{(s)}\mathcal{L}_t^{(s)} + \lambda_a^{(s)}\mathcal{L}_{\text{async}}^{(s)}$
    Update θ_t using $\nabla_{\theta_t}\mathcal{L}_{\text{total}}^{(s)}$
  end for
end for
return θ_T

A.8 CONSISTENCY OF MINI-BATCH AND TASK-LEVEL ORTHOGONAL DRIFT

Table 5: Consistency between task-level and mini-batch Procrustes alignment across continual transitions. Tasks follow the sequence Flickr30K, COCO, Kream, Visual Genome, Wikido, Lexica, and SUN397. Each transition is evaluated over 10 random mini-batch repartitionings.

Metric	T1→T2	T2→T3	T3→T4	T4→T5	T5→T6	T6→T7	Mean ± Std.
Task-level explained drift ↑	61.51	76.21	73.83	60.76	73.47	69.13	69.15 ± 6.62
Batch-specific extra drift ↓	5.53	3.46	2.23	5.80	4.24	5.30	4.43 ± 1.39
Global consistency ↑	91.76	95.66	97.07	91.28	94.54	92.88	93.86 ± 2.28

We further examine whether the orthogonal drift estimated from different mini-batches is consistent at the task level. We report three metrics in Table 5. Task-level explained drift measures the representation drift accounted for by a shared task-level orthogonal drift. Batch-specific extra drift measures the additional drift explained by treating each mini-batch independently, while global consistency measures how much batch-wise explainable orthogonal drift is captured at the task level. Across continual transitions, task-level orthogonal drift explains 69.15% of the representation drift on average, indicating that a substantial portion follows a shared geometry-preserving pattern. Batch-specific alignment adds only 4.43% explainable drift, showing limited batch-specific

variation. Global consistency reaches 93.86%, indicating that nearly all mini-batch orthogonal drift is consistent with the task-level shared drift. Together, these results validate Decomp-RGP by showing that mini-batch Procrustes reliably captures shared orthogonal drift while regularization focuses on residual drift that alters cross-modal structure.