MAPSELECT: MAPPING PRESERVATION AND ADAPTER SELECTION FOR CONTINUAL MULTIMODAL INSTRUCTION TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual multimodal instruction tuning must learn new tasks while maintaining performance on earlier ones. Retaining task-wise language adapters alone does not ensure this: their effectiveness also depends on the visual-to-language mapping and on which adapters are selected. We investigate these two factors on UCIT. Restoring an earlier visual projector recovers forgotten answers with the later language model unchanged; selecting the corresponding task-wise language adapter improves performance with the visual mapping unchanged. These observations lead to MAPSELECT, a framework that combines *mapping preservation* with *adapter selection*. History-Guided Mapping Adaptation retains projector adapters and constrains new updates using their learned directions. Selection-Guided Adapter Reuse combines current and historical adapters during learning and maintains consistent adapter selection throughout each answer. Pretrained Subspace Guidance further constrains language adaptation using pretrained weights. Extensive experiments on UCIT and MLLM-DCL demonstrate the effectiveness of our method.

1 INTRODUCTION

Visual instruction tuning (Liu et al., 2023; 2024) enables multimodal large language models (MLLMs) to perform image recognition (Radford et al., 2021; Jia et al., 2021), visual question answering (Antol et al., 2015; Hudson & Manning, 2019), and image captioning (Vinyals et al., 2015; Xu et al., 2015) by following natural-language instructions. In practical applications, new tasks arrive sequentially, motivating continual instruction tuning for MLLMs (Chen et al., 2024a; Guo et al., 2025c; Zhao et al., 2025). A central challenge is *catastrophic forgetting* (McCloskey & Cohen, 1989; Ratcliff, 1990), where adapting to new tasks can degrade performance on previously learned tasks.

MLLMs such as LLaVA (Liu et al., 2023; 2024) and InternVL (Chen et al., 2024b) connect a visual encoder to an LLM through a projector that maps image features into the language embedding space. Recent continual instruction-tuning methods, e.g., SMoLoRA (Wang et al., 2025), HiDe-LLaVA (Guo et al., 2025a), and SAME (Xie et al., 2026), use low-rank adaptation (LoRA) (Hu et al., 2022) to learn new tasks and retain earlier language adaptations. With task-wise LoRA adapters, each task has its own adaptation, and adapter selection determines which of these adaptations is used for an input. MVP (Jin et al., 2025) further investigates visual adaptation through instruction-conditioned projectors. However, retaining a language adapter does not preserve the visual mapping on which it relies, nor ensure that it will be selected for an earlier task. We therefore consider two complementary challenges: **(i) mapping preservation**, maintaining visual-to-language mappings that remain effective on earlier tasks; and **(ii) adapter selection**, choosing the appropriate language LoRA adapters for the input. **Mapping preservation and adapter selection together determine how retained adaptations contribute to earlier-task performance.**

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.

We examine these challenges on UCIT (Guo et al., 2025c), a six-task continual multimodal instruction-tuning benchmark.[1] Representative results in Table 1 show how each factor affects earlier-task performance. **(i) Mapping preservation.** In LoRA-FT and O-LoRA, the visual projector changes as the model learns new tasks. We restore the earlier projector while leaving the later LLM unchanged. Previously forgotten examples can then be answered correctly again: for LoRA-FT, 511 of 1,178 selected forgotten examples are recovered. Thus, changes in the visual mapping contribute to forgetting even when the later LLM can still answer these examples with the earlier visual representation. This finding supports *mapping preservation* alongside the preservation of language adapters. **(ii) Adapter selection.** After all six tasks, we keep the visual mapping fixed and compare the model's selection with task-wise selection, which uses the adapter learned for the evaluated task. The examples in Table 1 show that selecting this adapter can improve accuracy without changing its parameters. Retaining task-wise adapters therefore requires effective *adapter selection*.

Table 1: **Mapping preservation and adapter selection on UCIT.** (i) *Forgotten*: selected examples answered correctly before, but incorrectly after, learning a new task. *Recovered*: those answered correctly after restoring the earlier projector, with the later LLM fixed; parentheses give recovery percentages. (ii) Representative final accuracies (%) of MAPSELECT with visual mapping fixed. Model selection uses the input; task-wise selection uses the known task label to choose its adapter.

(i) Mapping preservation		
Baseline	**Forgotten**	**Recovered**
LoRA-FT	1,178	511 (43.38%)
O-LoRA	367	322 (87.74%)
(ii) Adapter selection		
Task	**Model-selected**	**Task-wise**
ImageNet-R	88.73	**90.50**
IconQA	66.20	**73.90**
CLEVR-Math	74.80	**75.33**

To address these challenges, we propose MAPSELECT, a continual multimodal instruction-tuning framework that learns without any replay of earlier training samples. It comprises three modules. For **mapping preservation**, History-Guided Mapping Adaptation (HMA) retains earlier projector adapters and uses their learned directions to constrain new mapping updates. For **adapter selection**, Selection-Guided Adapter Reuse (SAR) learns new tasks with a related historical adapter and maintains a consistent selection of language adapters throughout each answer. The third module, Pretrained Subspace Guidance (PSG), uses directions from pretrained language weights to constrain adapter updates. Besides, both HMA and PSG apply soft gradient constraints, using earlier projector parameters and pretrained language weights as their respective references.

Our contributions are summarized as follows:

- We investigate mapping preservation and adapter selection in continual multimodal instruction tuning, showing that restoring visual mappings and selecting retained adapters can recover earlier-task performance.
- We propose MAPSELECT, a unified framework that combines mapping preservation and adapter selection for continual multimodal instruction tuning.
- Experiments on UCIT and MLLM-DCL demonstrate improved performance over the compared baselines, with task-wise analyses and ablations examining the contributions of the proposed modules.

2 RELATED WORK

Continual multimodal adaptation. CoIN (Chen et al., 2024a), MCITlib (Guo et al., 2025c), and MLLM-DCL (Zhao et al., 2025) evaluate successive multimodal tasks and domains. HiDe-LLaVA (Guo et al., 2025a) combines adapter expansion and fusion; CL-MoE (Huai et al., 2025) and CRAM (Tang et al., 2026) select experts using input-dependent information. Other approaches use task prompts (Zheng et al., 2024; Zeng et al., 2025) or compress adapters (Meng et al., 2026). MVP (Jin et al., 2025) studies instruction-conditioned visual projectors. MAPSELECT addresses both projector adaptation and language-adapter selection through separate projector adapters and historical-adapter reuse.

[1]The task order is ImageNet-R → ArxivQA → VizWiz → IconQA → CLEVR-Math → Flickr30k.

Understanding forgetting. Prior work studies generalizability and adaptivity (Zhou et al., 2025), and distinguishes task performance from retained knowledge. Spurious Forgetting (Zheng et al., 2025) studies disrupted task alignment in language models. SMoLoRA (Wang et al., 2025) separates visual understanding from instruction following, while SAME (Xie et al., 2026) examines how expert updates and expert selection affect forgetting. Our analyses study recoverable earlier-task performance by separately changing the visual mapping and language-adapter selection.

Subspace constraints. O-LoRA (Wang et al., 2023) constrains low-rank adaptation across tasks. Conceptor theory (Jaeger, 2017) supplies a soft spectral operator, and CODE-CL (Apolinario et al., 2025) uses activation references for continual learning. HMA and PSG apply this established operator to historical projector factors and pretrained language weights, respectively. These references come directly from stored parameters and guide projector and language adaptation at their respective sites.

3 Mapping Preservation and Adapter Selection

Retaining task-wise language adapters does not by itself ensure effective use of earlier adaptations. We examine two factors on UCIT: *mapping preservation*, which concerns the visual representations supplied to the LLM, and *adapter selection*, which concerns the language adaptations used to process them. Figure 1 shows that both factors affect earlier-task performance.

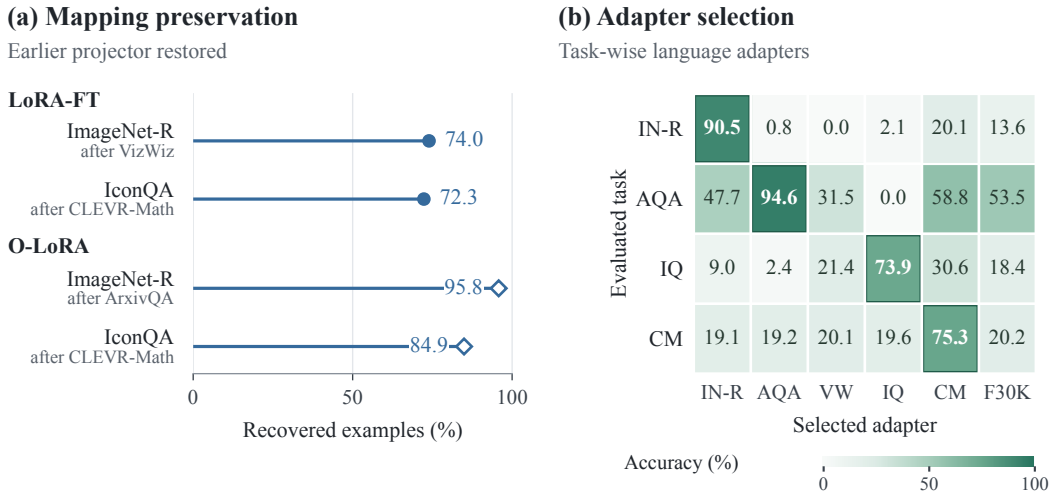

Figure 1: **Mapping preservation and adapter selection.** (a) Representative projector-restoration results on ImageNet-R and IconQA for LoRA-FT and O-LoRA, with the later LLM fixed. Recovery is measured among selected examples that became incorrect after learning the task named below each row label. (b) Final accuracy (%) of the six task-wise language adapters, with visual mapping fixed. For each evaluated task, its corresponding adapter achieves the highest accuracy (outlined cells). IN-R: ImageNet-R; AQA: ArxivQA; VW: VizWiz; IQ: IconQA; CM: CLEVR-Math; F30K: Flickr30k.

3.1 Mapping Preservation

Learning a new task changes the visual mapping used by earlier tasks. To examine its role in forgetting, we evaluate LoRA-FT and O-LoRA with their earlier visual projectors while keeping the later LLM fixed. Figure 1a shows representative results on ImageNet-R and IconQA: restoring the earlier projector recovers answers that had become incorrect after a new task was learned. The later LLM can therefore still use the earlier visual representation to answer these questions.

These results show that some earlier-task errors arise from changes in the visual mapping: the later LLM can still answer the same questions when the earlier visual representation is restored. *Mapping preservation* therefore needs to accompany the preservation of language adaptations. Further results on projector restoration, including all evaluated task transitions, are provided in Appendix D.

3.2 ADAPTER SELECTION

Earlier language adapters can remain effective after their tasks lose performance. In Figure 1b, we compare all six retained task-wise adapters on the four accuracy-scored tasks after the UCIT sequence, with the visual mapping fixed. For every evaluated task, the adapter learned for that task achieves the highest accuracy among the six adapters. Task-wise selection also improves over the model's own selection on ImageNet-R, IconQA, and CLEVR-Math, as shown in Table 1. The retained adapters thus support higher performance when they are selected for their corresponding tasks.

The same relationship holds during the task sequence. After learning VizWiz, selecting the ImageNet-R adapter restores 426 of the 445 previously correct answers that had become incorrect. This recovery uses the same model parameters and therefore reflects how the retained adaptation is used. Together, these findings support *adapter selection* as a necessary complement to retaining task-wise adapters. Further results on adapter selection are provided in Appendix E.

4 MAPSELECT: MAPPING PRESERVATION AND ADAPTER SELECTION

MAPSELECT combines *mapping preservation* and *adapter selection* in a unified framework for continual multimodal instruction tuning without any replay of earlier training samples. HMA preserves visual mappings, SAR learns and selects language adaptations, and PSG guides language adaptation using pretrained weights. Figure 2 shows their roles. For clarity, Algorithm 1 summarizes training and inference after the module definitions.

4.1 TASK-WISE LORA ADAPTERS

We add task-wise LoRA adapters to the visual projector and the language model of a frozen pretrained MLLM. For each task t, the current adapters learn from the task's instruction data, while the completed task-wise factors remain frozen. Earlier projector adapters retain the learned visual mappings, and earlier language adapters remain available for selection and reuse.

For visual-to-language mapping, each projector transformation j has a pretrained weight W^m_{j0} and task-wise factors $(A^m_{j,i}, B^m_{j,i})$. Given visual features z, its output is

$$m_j(z; q^v) = W^m_{j0} z + \eta_m \sum_{i=1}^{t} q^v_i B^m_{j,i} A^m_{j,i} z, \tag{1}$$

where q^v_i weights task i's adapter and η_m is the LoRA scale. The two projector transformations have separate factors and retain their intervening nonlinearity. Training uses the current adapter, $q^v = e_t$; inference determines q^v from image-feature similarity to the retained task features.

For language-adapter selection, each adapted layer ℓ combines a shared input factor $A_{\ell s}$ with task-wise factors $(A_{\ell i}, B_{\ell i})$:

$$\Delta_{\ell i}(h) = B_{\ell i}(A_{\ell s} + \lambda A_{\ell i})h, \qquad y_\ell = W_{\ell 0} h + \eta \sum_{i=1}^{t} r_{\ell i} \Delta_{\ell i}(h). \tag{2}$$

Here $W_{\ell 0}$ is frozen, $A_{\ell s}, A_{\ell i} \in \mathbb{R}^{r \times d_{\text{in}}}$, and $B_{\ell i} \in \mathbb{R}^{d_{\text{out}} \times r}$. The scalars η and λ set the adapter scale and task-wise input-factor weight. The shared factor learns across tasks, while the task-wise factors provide additional adaptation. The coefficients $r_{\ell i}$ determine which language adapters are used for the input. Further details on the adapters, task features, and training schedules are provided in Appendix B.

4.2 HISTORY-GUIDED MAPPING ADAPTATION

HMA uses earlier projector adapters to guide the learning of a new visual mapping. For each transformation j, we form reference matrices $X^m_{j,A,<t}$ and $X^m_{j,B,<t}$ from earlier projector factors. Following Conceptor theory (Jaeger, 2017) and CODE-CL (Apolinario et al., 2025), a reference matrix $X \in \mathbb{R}^{n \times d}$ defines a soft gradient constraint:

$$C_X = X^\top [X X^\top + (n a^{-2} + \epsilon) I_n]^{-1} X, \qquad \mathcal{F}(G; X) = G(I - \rho C_X). \tag{3}$$

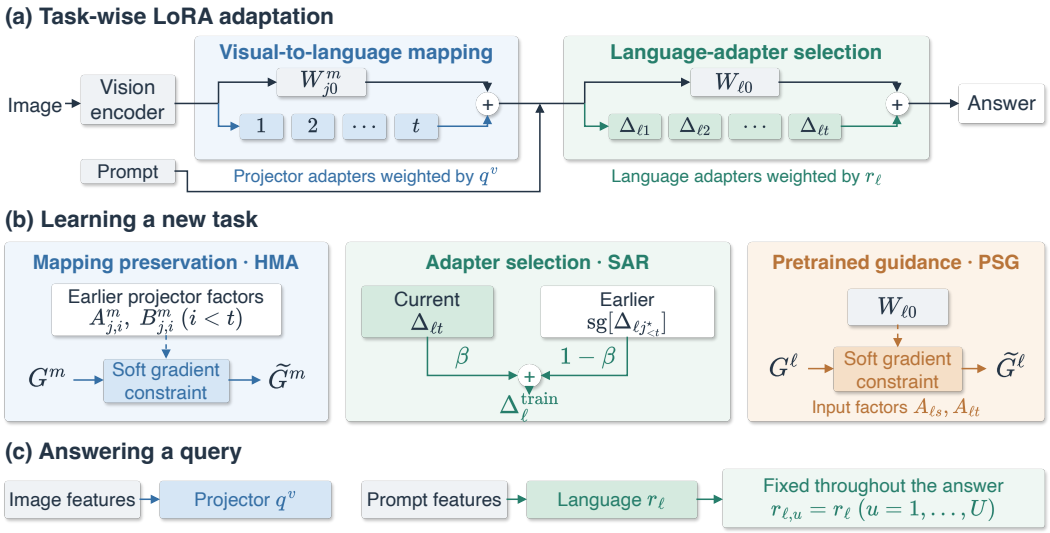

Figure 2: **Overview of MAPSELECT.** (a) Task-wise adapters adapt the projector and language model while pretrained weights remain frozen. (b) HMA constrains projector gradients using earlier factors; SAR combines current and earlier language adapters; PSG constrains language input-factor gradients using pretrained weights. G^m and G^ℓ denote gradients, tildes denote filtered gradients, and sg stops gradient flow. (c) Image and prompt features select the respective adapters. Language selection stays fixed throughout the answer; task identity is not required.

The aperture $a > 0$ controls the response to the reference, $\rho \in [0, 1]$ sets the constraint strength, and $\epsilon > 0$ stabilizes the inverse. C_X gives greater weight to directions strongly represented in X, so $\mathcal{F}$ reduces the gradient along those directions while retaining components outside the reference span. HMA applies this constraint to each current projector factor $P \in \{A, B\}$:

$$\widetilde{\nabla P^m_{j,t}} = \mathcal{F}(\nabla P^m_{j,t}; X^m_{j,P,<t}). \tag{4}$$

The A and B references constrain their respective input and low-rank directions. The resulting updates let the current adapter learn a new mapping while reducing changes along directions represented by earlier projector factors. HMA starts when historical adapters become available. Further details on reference construction are provided in Appendix B.

4.3 SELECTION-GUIDED ADAPTER REUSE

SAR connects learning a new language adapter with using retained ones. During training, a related earlier adapter provides a fixed contribution to the current prediction. Let h^q_ℓ denote the current prompt feature and $\mu_{\ell i}$ the representative prompt feature for task i. At layer ℓ, SAR selects $j^\star_{<t} = \arg\max_{i<t} \cos(h^q_\ell, \mu_{\ell i})$ and uses

$$\Delta^{\text{train}}_\ell(h) = \beta \Delta_{\ell t}(h) + (1 - \beta)\, \text{sg}[\Delta_{\ell j^\star_{<t}}(h)]. \tag{5}$$

Similarity determines which earlier adapter is used; the hyperparameter $\beta = 0.85$ sets the relative contribution of the current adapter. The stop-gradient operator sg keeps the historical contribution fixed during backpropagation. This contribution affects the combined prediction and hence the loss from which the current adapter learns. The first task uses only its current adapter.

At inference, SAR selects language adapters from the prompt and keeps the selection fixed throughout the answer. For scores $s_{\ell i} = \cos(h^q_\ell, \mu_{\ell i})$, let $\mathcal{S}_\ell = \{i : s_{\ell i} = \max_j s_{\ell j}\}$ contain the highest-scoring adapters. Their weights are

$$r_{\ell i} = \frac{\mathbf{1}[i \in \mathcal{S}_\ell] \exp(s_{\ell i}/\tau)}{\sum_{j \in \mathcal{S}_\ell} \exp(s_{\ell j}/\tau)}, \qquad r_{\ell,u} = r_\ell, \quad u = 1, \ldots, U, \tag{6}$$

where τ is the softmax temperature and U is the answer length. Each layer therefore uses the same selected adapters for all response tokens, with the choice determined by the input prompt. Further details on adapter selection and reuse are provided in Appendix B.

4.4 PRETRAINED SUBSPACE GUIDANCE

PSG guides language adaptation with the frozen pretrained weights. For each adapted layer, we form a reference X_ℓ^0 from rows of $W_{\ell 0}$ and use it to constrain the shared and current task-wise input factors:

$$\widetilde{\nabla A_{\ell s}} = \mathcal{F}(\nabla A_{\ell s}; X_\ell^0), \qquad \widetilde{\nabla A_{\ell t}} = \mathcal{F}(\nabla A_{\ell t}; X_\ell^0). \tag{7}$$

Both factors share the reference's input coordinates; the output factor $B_{\ell t}$ uses its ordinary gradient. HMA and PSG thus apply the same soft constraint using complementary references: historical projector factors for visual mapping and pretrained weights for language adaptation.

The effect of this constraint can be characterized through its response to the reference directions. Let $R = X^\top X/n$, $\gamma = a^{-2} + \epsilon/n$, and $E_X(G) = \|GX^\top\|_F^2/n$. The Conceptor has the equivalent form $C_X = R(R + \gamma I)^{-1}$.

Proposition 1. *For eigenpairs (s_k, v_k) of R, $\gamma > 0$, and $0 \le \rho \le 1$,*

$$E_X(\mathcal{F}(G; X)) = \sum_k s_k \left(1 - \rho \frac{s_k}{s_k + \gamma}\right)^2 \|Gv_k\|_2^2 \le E_X(G). \tag{8}$$

The constraint attenuates gradients along the reference directions and leaves components outside their span unchanged. This property applies to both HMA and PSG. We train the adapters with the autoregressive instruction loss $\mathcal{L}_t$, applying these constraints to the gradients before each optimizer update. Further theoretical details and the proof are provided in Appendix G.

Algorithm 1 MAPSELECT: training and inference

Input: Task sequence $\{\mathcal{D}_t\}_{t=1}^T$; pretrained MLLM.
1: **for** task $t = 1, \ldots, T$, with new task-wise adapters **do**
2: **for** minibatch from $\mathcal{D}_t$ **do**
3: Compute $\mathcal{L}_t$ with **SAR** (5) and its gradients.
4: Apply **HMA** (4) and **PSG** (7); update trainable factors.
5: **end for**
6: Retain the task's adapters and feature summaries.
7: **end for**
Inference: Select q^v from the image and r_ℓ from the prompt; keep r_ℓ fixed throughout the answer (6).

5 EXPERIMENTS

We evaluate MAPSELECT on task and domain sequences, then examine how HMA, SAR, and PSG affect learning and retention.

5.1 EXPERIMENTAL SETUP

Benchmarks and metrics. UCIT (Guo et al., 2025c) follows six tasks: ImageNet-R (IN-R), ArxivQA (AQA), VizWiz (VW), IconQA (IQ), CLEVR-Math (CM), and Flickr30k (F30K). MLLM-DCL (Zhao et al., 2025) follows five domains: remote sensing, medicine, autonomous driving, science, and finance. We report mean final performance (MFN), performance when each task is first learned (MFT), average performance across learning stages (MAA), and backward transfer (BWT). UCIT uses accuracy and CaptionAvg for its QA and captioning tasks, respectively; MLLM-DCL uses accuracy. Further details on evaluation and metric definitions are provided in Appendices A and C.

Baselines and training. We use LLaVA-v1.5-7B (Liu et al., 2024) and compare with LoRA-FT (Hu et al., 2022), O-LoRA (Wang et al., 2023), MoELoRA and Replay (Chen et al., 2024a), CL-MoE (Huai et al., 2025), HiDe (Guo et al., 2025a), DISCO (Guo et al., 2025b), and SMoLoRA (Wang et al., 2025). MLLM-DCL additionally includes ModalPrompt (Zeng et al., 2025) and SEFE (Chen et al., 2025). UCIT uses language/projector ranks 96/16, learning rate 2×10^{-4}, and effective batch size 64. Our models and ablations use seed 42 and the supplied development splits. Benchmark comparisons

Table 2: **Comparison on UCIT after six tasks.** Final task scores and continual-learning metrics; tasks appear in learning order. IN-R/AQA/IQ/CM: accuracy (%); VW/F30K: CaptionAvg. Aggregate metrics combine these task scores. Bold/underline: best/second best; shading identifies MAPSELECT.

	Final task scores ↑						Continual metrics ↑			
Method	**IN-R**	**AQA**	**VW**	**IQ**	**CM**	**F30K**	**MFT**	**MFN**	**MAA**	**BWT**
LoRA-FT	64.87	77.30	44.75	70.03	55.87	57.24	73.44	61.68	76.97	−17.72
O-LoRA	85.27	93.97	37.43	66.73	53.30	52.09	70.80	64.80	78.82	−7.20
MoELoRA	64.43	78.50	44.50	64.17	56.20	57.39	75.30	60.87	76.76	−17.32
CL-MoE	65.57	77.63	44.77	61.87	47.83	<u>57.52</u>	75.89	60.87	76.73	−18.03
HiDe	85.37	91.80	47.44	67.47	55.23	53.96	72.79	66.88	79.78	<u>−7.10</u>
DISCO	<u>87.70</u>	<u>94.50</u>	46.69	70.93	55.67	56.54	<u>76.24</u>	<u>68.67</u>	<u>81.82</u>	−9.08
Replay	75.63	84.50	<u>50.86</u>	71.03	<u>67.10</u>	55.53	75.61	67.78	79.62	−9.40
SMoLoRA	75.53	73.50	45.63	<u>71.10</u>	60.37	56.24	72.36	63.73	75.10	−10.36
MAPSELECT	**89.07**	**94.90**	**64.82**	**67.53**	**74.53**	**56.28**	**74.48**	**74.52**	**83.26**	**0.05**

Table 3: **Component ablations on UCIT.** The full method and three variants are trained with a common configuration and seed (42). Each variant removes one module. ✓ retained, × removed. Removing SAR removes both training-time adapter reuse and the fixed selection during each answer. Bold marks best.

	Components			Continual metrics ↑			
Variant	HMA	SAR	PSG	**MFT**	**MFN**	**MAA**	**BWT**
w/o HMA	×	✓	✓	74.56	73.97	77.60	−1.91
w/o SAR	✓	×	✓	73.38	73.80	80.65	−1.89
w/o PSG	✓	✓	×	74.14	73.84	80.53	−1.55
MAPSELECT	✓	✓	✓	**75.07**	**74.13**	**82.06**	**−1.13**

follow the respective scoring protocols; the adapter-selection analysis uses answer matching. Further details on training and answer matching are provided in Appendices B and E.

5.2 PERFORMANCE ACROSS TASKS AND DOMAINS

MAPSELECT achieves the highest MFN, MAA, and BWT on UCIT (Table 2), with MFN 74.52 compared with 68.67 for DISCO. Its final scores are highest on four of the six tasks, and BWT of $+0.05$ indicates that earlier-task performance is maintained on average. On MLLM-DCL, it achieves the highest final accuracy in all five domains and MFN of 66.28 versus 61.33 for DISCO (Table 4). These results show effective adaptation across both task and domain changes.

The benchmark retention curves in Figure 3(c) show that MAPSELECT maintains near-zero running BWT throughout UCIT. The per-task changes in panel (d) explain the final average. Tables 7 and 8 give every task and domain score throughout learning.

5.3 CONTRIBUTIONS OF HMA, SAR, AND PSG

For component comparisons, we train the full method and three variants with a common configuration, removing one module at a time. Table 3 reports their aggregate performance. Further details on the ablation configuration are provided in Appendix B.

Mapping preservation with HMA. Removing HMA reduces MAA from 82.06 to 77.60. After VizWiz, ImageNet-R accuracy is 76.30% with the full method and 1.40% without HMA (Figure 3(a,b)). Both recover at later stages, so final MFN alone obscures this difference during adaptation.

Table 4: **Comparison on MLLM-DCL after five domains.** Final accuracy is reported for each domain in learning order. RS: remote sensing; Med: medicine; AD: autonomous driving; Sci: science; Fin: finance. Accuracy in %, BWT in percentage points. Bold/underline: best/second best.

	Final domain scores ↑					Continual metrics ↑			
Method	**RS**	**Med**	**AD**	**Sci**	**Fin**	**MFT**	**MFN**	**MAA**	**BWT**
LoRA-FT	67.86	47.53	34.24	41.88	<u>89.89</u>	66.52	56.28	62.24	−12.80
O-LoRA	75.78	36.69	23.05	37.05	81.02	59.33	50.72	59.45	−10.77
MoELoRA	<u>77.41</u>	46.57	31.02	40.58	88.23	65.65	56.76	62.59	−11.12
ModalPrompt	48.65	28.35	12.06	30.12	82.82	54.55	40.40	42.00	−17.68
CL-MoE	71.41	48.56	30.56	41.74	89.79	<u>66.60</u>	56.41	62.39	−12.74
HiDe	71.63	48.73	35.06	40.86	78.06	60.75	54.87	61.12	−7.35
SEFE	77.28	<u>51.97</u>	41.63	42.08	89.36	66.38	60.46	<u>65.14</u>	−7.40
DISCO	76.67	44.85	<u>47.01</u>	<u>49.07</u>	89.07	64.93	<u>61.33</u>	64.29	<u>−4.50</u>
MAPSELECT	**79.59**	**56.77**	**52.49**	**52.42**	**90.13**	**67.39**	**66.28**	**68.28**	**−1.39**

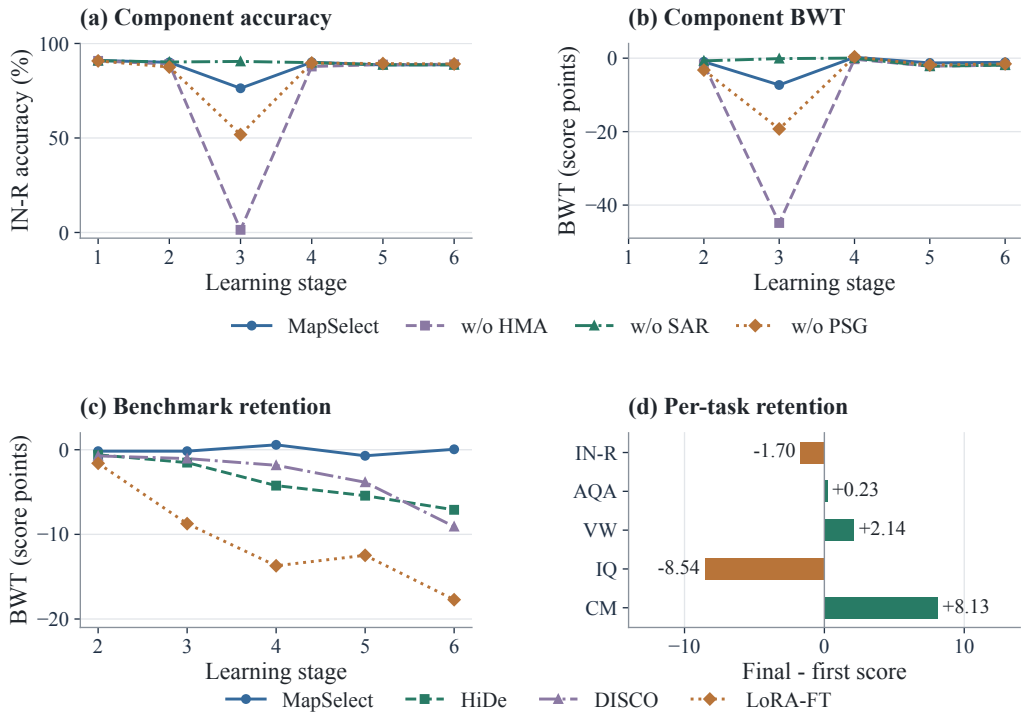

Figure 3: **Learning dynamics on UCIT.** (a,b) Component comparisons under the common configuration in Table 3. (c) Benchmark running BWT for MAPSELECT and three baselines. (d) Each earlier task's final score minus its score when first learned. Running BWT averages changes on previously learned tasks; UCIT combines accuracy and CaptionAvg.

Adapter selection and reuse with SAR. Removing SAR reduces final MFN from 74.13 to 73.80 and BWT from −1.13 to −1.89 (Table 3), supporting its role in retaining performance at the end of the sequence.

Language adaptation with PSG. PSG increases MAA from 80.53 to 82.06 and final MFN from 73.84 to 74.13. After VizWiz, ImageNet-R accuracy improves from 51.80% to 76.30% (Figure 3(a)). HMA and PSG thus support retention by constraining visual and language adaptation, respectively.

5.4 Pretrained-Task Performance and Computational Cost

Pretrained-task performance. Table 5 compares the full method and w/o PSG on 2,374 MME (Fu et al., 2025) and 4,377 MMBench (Liu et al., 2025) examples. PSG improves the MME score by 121.12 after VizWiz and 164.26 after the full sequence; the corresponding MMBench differences are -0.16 and -1.01 percentage points. Further details on candidate scoring are provided in Appendix F.

Table 5: **Pretrained-task performance.** Full and w/o PSG use identical scoring. MME: official total; MMBench: accuracy (%). T3 follows VizWiz, T6 all six tasks.

Stage	Full	w/o PSG
MME		
Pretrained	1745.60	
T3	1611.47	1490.34
T6	1558.13	1393.88
MMBench		
Pretrained	73.70	
T3	55.86	56.02
T6	70.92	71.92

Table 6: **Computational cost on UCIT.** Counts exclude the backbone. Storage: six tasks, bf16, decimal units. Training: two GPUs, 30.97 GPU-hours, excluding loading and evaluation.

Quantity	Value
Adapter parameters	
Trainable per task	349.39M
Added per task	240.07M
Retained after six tasks	1.55B
Storage and training	
Adapter storage	3.10 GB
Task features	13.7 MB
Training time	15.49 h

Computational cost. Table 6 summarizes parameters, storage, and training time. Each task adds 240.07M adapter parameters; shared factors contribute 109.31M. Language selection compares the prompt with task features once, costing $O(td)$ per adapted linear, then reuses the selected weights throughout the answer. Gradient filtering is used only during training. Further details on parameter counts and filter complexity are provided in Appendix H.

Table 7: **Complete UCIT trajectory.** Task scores follow Table 2; Mean averages seen tasks. A dash marks a task not yet learned.

Stage	IN-R	AQA	VW	IQ	CM	F30K	Mean
T1	90.77	–	–	–	–	–	90.77
T2	90.60	94.67	–	–	–	–	92.63
T3	90.47	94.63	62.68	–	–	–	82.59
T4	90.23	95.00	64.63	76.07	–	–	81.48
T5	89.03	95.03	64.97	72.30	66.40	–	77.55
T6	89.07	94.90	64.82	67.53	74.53	56.28	74.52

Table 8: **Complete MLLM-DCL trajectory.** Accuracy (%); Mean averages seen domains. A dash marks a task not yet learned.

Stage	RS	Med	AD	Sci	Fin	Mean
T1	80.47	–	–	–	–	80.47
T2	79.19	61.16	–	–	–	70.18
T3	79.09	60.23	52.73	–	–	64.02
T4	79.58	57.18	52.62	52.46	–	60.46
T5	79.59	56.77	52.49	52.42	90.13	66.28

6 Conclusion

We investigate mapping preservation and adapter selection in continual multimodal instruction tuning. Restoring an earlier visual mapping and selecting retained language adapters can recover earlier-task performance without further training. These observations motivate MapSelect, which combines history-guided projector adaptation, selection-guided adapter reuse, and pretrained subspace guidance. Experiments on UCIT and MLLM-DCL show improved learning and retention over the compared baselines. Together, the analysis and results highlight the importance of preserving visual mappings alongside selecting effective language adapters.

Reproducibility statement

The appendix provides evaluation protocols, model settings, metric definitions, additional mapping and adapter-selection comparisons, the gradient-constraint proof, and computational costs. The supplementary data contain the recorded measurements and figure-generation code. Further details on training and evaluation are provided in Appendices B and A.

AI Use Statement

In this work, we used generative AI tools for assisting with literature search and locating potentially relevant prior work. We have not used generative AI tools for other substantive research tasks, and the remaining tasks are not applicable to this work. Additionally, we used generative AI tools for language editing and polishing of the manuscript, including improving grammar, wording, clarity, and readability. We have reviewed all AI-assisted work. References located with AI assistance were checked against the original sources, and all AI-assisted language edits were reviewed before inclusion in the manuscript. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2425–2433, 2015.

Marco P. E. Apolinario, Sakshi Choudhary, and Kaushik Roy. CODE-CL: Conceptor-based gradient projection for deep continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 775–784, 2025. URL https://arxiv.org/abs/2411.15235.

Cheng Chen, Junchen Zhu, Xu Luo, Heng Tao Shen, Jingkuan Song, and Lianli Gao. CoIN: A benchmark of continual instruction tuning for multimodel large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 57817–57840. Curran Associates, Inc., 2024a. doi: 10.52202/079017-1844.

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu, Horace Ho Shing Ip, and Sam Kwong. SEFE: Superficial and essential forgetting eliminator for multimodal continual instruction tuning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 7982–8001. PMLR, 2025. URL https://proceedings.mlr.press/v267/chen25n.html.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024b. URL https://arxiv.org/abs/2312.14238.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. MME: A comprehensive evaluation benchmark for multimodal large language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4899.

Haiyang Guo, Fanhu Zeng, Ziwei Xiang, Fei Zhu, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin Liu. HiDe-LLaVA: Hierarchical decoupling for continual instruction tuning of multimodal large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13572–13586. Association for Computational Linguistics, 2025a. doi: 10.18653/v1/2025.acl-long.666.

Haiyang Guo, Fanhu Zeng, Fei Zhu, Wenzhuo Liu, Da-Han Wang, Jian Xu, Xu-Yao Zhang, and Cheng-Lin Liu. Federated continual instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1325–1335, 2025b. URL https://arxiv.org/abs/2503.12897.

Haiyang Guo, Fei Zhu, Hongbo Zhao, Fanhu Zeng, Wenzhuo Liu, Shijie Ma, Da-Han Wang, and Xu-Yao Zhang. MCITlib: Multimodal continual instruction tuning library and benchmark, 2025c. URL https://arxiv.org/abs/2508.07307.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. CL-MoE: Enhancing multimodal large language model with dual momentum mixture-of-experts for continual visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19608–19617, 2025. URL https://arxiv.org/abs/2503.00413.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6700–6709, 2019.

Herbert Jaeger. Using conceptors to manage neural long-term memories for temporal patterns. *Journal of Machine Learning Research*, 18(13):1–43, 2017. URL https://jmlr.org/papers/v18/15-449.html.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4904–4916. PMLR, 2021. URL https://proceedings.mlr.press/v139/jia21b.html.

Hyundong Jin, Hyung Jin Chang, and Eunwoo Kim. Instruction-grounded visual projectors for continual learning of generative vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3466–3476, 2025. URL https://arxiv.org/abs/2508.00260.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. URL https://arxiv.org/abs/2304.08485.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26296–26306, 2024. URL https://arxiv.org/abs/2310.03744.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *Computer Vision – ECCV 2024*, volume 15064 of *Lecture Notes in Computer Science*, pp. 216–233. Springer Nature Switzerland, 2025. doi: 10.1007/978-3-031-72658-3_13.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pp. 109–165. Academic Press, 1989. doi: 10.1016/S0079-7421(08)60536-8.

Weicheng Meng, Jingyang Qiao, Zhizhong Zhang, Shaohui Liu, and Yuan Xie. PCLR: Progressively compressed LoRA for multimodal continual instruction tuning. In *International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10009038.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Roger Ratcliff. Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological Review*, 97(2):285–308, 1990. doi: 10.1037/0033-295X.97.2.285.

Jun-Tao Tang, Zhen-Hao Xie, Yu-Cheng Shi, and Da-Wei Zhou. CRAM: Centroid-routing and adaptive MoE for multimodal continual instruction tuning. In *Conference on Empirical Methods in Natural Language Processing*, 2026. URL https://arxiv.org/abs/2606.02502. Accepted, Main Conference.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164, 2015.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10658–10671. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.715.

Ziqi Wang, Chang Che, Qi Wang, Yangyang Li, Zenglin Shi, and Meng Wang. SMoLoRA: Exploring and defying dual catastrophic forgetting in continual visual instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 177–186, 2025. URL https://arxiv.org/abs/2411.13949.

Zhen-Hao Xie, Jun-Tao Tang, Yu-Cheng Shi, Han-Jia Ye, De-Chuan Zhan, and Da-Wei Zhou. SAME: Stabilized mixture-of-experts for multimodal continual instruction tuning. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.01990.

Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2048–2057. PMLR, 2015. URL https://proceedings.mlr.press/v37/xuc15.html.

Fanhu Zeng, Fei Zhu, Haiyang Guo, Xu-Yao Zhang, and Cheng-Lin Liu. ModalPrompt: Towards efficient multimodal continual instruction tuning with dual-modality guided prompt. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12126–12141. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.609.

Hongbo Zhao, Fei Zhu, Haiyang Guo, Meng Wang, Rundong Wang, Gaofeng Meng, and Zhaoxiang Zhang. MLLM-CL: Continual learning for multimodal large language models, 2025. URL https://arxiv.org/abs/2506.05453.

Junhao Zheng, Qianli Ma, Zhen Liu, Binquan Wu, and Huawen Feng. Beyond anti-forgetting: Multimodal continual instruction tuning with positive forward transfer, 2024. URL https://arxiv.org/abs/2401.09181.

Junhao Zheng, Xidi Cai, Shengjie Qiu, and Qianli Ma. Spurious forgetting in continual learning of language models. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2501.13453.

Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision*, 133(3):1012–1032, 2025. doi: 10.1007/s11263-024-02218-0.

SUPPLEMENTARY MATERIAL

We provide the experimental settings and supporting evidence for MAPSELECT. Appendices A and B describe evaluation and training; Appendix C gives metric definitions; complete trajectories appear in the main text. Appendices D and E expand the mapping and adapter-selection comparisons, and Appendix F describes pretrained-task evaluation. Appendices G and H provide the gradient-constraint proof and computational costs. Appendix I provides additional recovery analyses and visual examples.

A EVALUATION SPLITS AND SCORING

Table 9 lists the task sequences and supplied splits used in our experiments. A training image can appear with several instructions. Shared images are byte-identical images in both splits; their questions and answers may differ. UCIT scores informed method design and hyperparameters, and MLLM-DCL scores informed capacity and protocol comparisons. We retain the supplied prompts, including rationale text in some ArxivQA annotations.

Table 9: Training and evaluation splits in stream order. Shared images are byte-identical images present in both splits.

		Training split		Evaluation / overlap	
Stream	**Task / domain**	Entries	Unique IDs	Entries	Shared images
UCIT	ImageNet-R	23,998	23,998	3,000	9
	ArxivQA	40,000	40,000	3,000	1,259
	VizWiz	40,000	20,559	3,000	99
	IconQA	29,859	29,859	3,000	411
	CLEVR-Math	40,000	39,387	3,000	0
	Flickr30k	40,000	21,632	3,000	0
MLLM-DCL	Remote sensing	60,000	60,000	10,000	0
	Medicine	22,842	22,842	9,790	3,696
	Autonomous driving	60,000	60,000	10,000	3,681
	Science	30,747	30,747	7,687	3,741
	Finance	60,000	60,000	10,000	1,214

Task scores. ImageNet-R, ArxivQA, IconQA, CLEVR-Math, and all MLLM-DCL domains use exact-match accuracy after uppercasing predictions and labels. The main benchmark scorer preserves whitespace and punctuation. VizWiz and Flickr30k use the following average of seven caption metrics:

$$\text{CaptionAvg} = \frac{100}{7}\left(\sum_{k=1}^{4}\text{BLEU}_k + \text{METEOR} + \text{ROUGE}_L + \text{CIDEr}\right). \tag{9}$$

All scores are computed from predictions and reference answers without an LLM judge. Further details on aggregate continual-learning metrics are provided in Appendix C.

Controlled comparisons. Adapter-selection comparisons use the same examples and fixed parameters across conditions. Their answer matching additionally strips surrounding whitespace. Projector-restoration results use the recorded correctness of the original predictions. MME and MMBench use fixed candidate scoring. Further details on these comparisons are provided in Appendices D, E, and F.

B MODEL AND TRAINING SETTINGS

Table 10 specifies the two benchmark configurations. LoRA adapters are attached to seven linear transformations in each of the 32 language blocks and to both visual projector transformations. Completed task-wise factors remain frozen.

Table 10: Backbone, adapter, and training settings for UCIT and MLLM-DCL.

Setting	UCIT	MLLM-DCL
Parameterization		
Backbone	LLaVA-v1.5-7B	LLaVA-v1.5-7B
Language rank / scale	96 / 2	128 / 2
Cross-modal rank / scale	16 / 2	16 / 2
Training		
Language / cross-modal LR	2×10^{-4} / 2×10^{-4}	2×10^{-5} / 2×10^{-5}
Effective batch size	64	64
Epochs per task	1, 1, 1, 1, 1, 1	1, 3, 1, 2, 1

Optimization. UCIT uses AdamW with a cosine learning-rate schedule, warmup ratio 0.03, zero weight decay, and bf16 training. Benchmark runs use eight workers and component comparisons use two, with effective batch size 64 in both cases. All reported runs use seed 42. The shared language input factor is trained throughout both sequences. The component configuration uses $\lambda = 0.35$ in Equation (2). For image v, instruction q, and response y, the training objective is

$$\mathcal{L}_t = -\mathbb{E}_{(v,q,y)\sim\mathcal{D}_t} \sum_{u=1}^{U} \log p_\theta(y_u \mid v, q, y_{<u}). \tag{10}$$

Gradient constraints. HMA uses $(a_1, \rho_1, \epsilon_1) = (5, 0.5, 10^{-6})$; PSG uses $(a_3, \rho_3, \epsilon_3) = (10, 0.2, 10^{-6})$ on both benchmarks. For each projector transformation, HMA concatenates earlier A factors and earlier B factors into separate references, starting at task 2. PSG takes rows from the frozen pretrained language weights, transposing when needed to match the adapted factor's input coordinates. Each reference is limited to 2,048 rows by taking every $\lceil n/2048 \rceil$-th row when it originally has $n > 2048$ rows. The references are used only during training.

Task features and selection. For UCIT projector selection, we mean-pool and normalize the visual tokens entering the projector, then maintain a task-wise exponential moving average with momentum 0.95. Each adapted language linear stores the mean of its inputs at the last valid prompt position for each task; answer tokens and padding are excluded. Feature dimensions match the linear's input dimension. Cosine similarity to these normalized task features determines selection. Language selection retains the highest score at temperature 0.1, and projector selection retains scores at or above the second-highest score at temperature 0.07. Ties are retained and normalized by softmax; only tasks with an available feature mean are eligible. When fewer than two projector adapters are available, all available adapters are used. Training reuses the most similar prior language adapter with $\beta = 0.85$; the prior contribution is detached in its entirety. MLLM-DCL also uses top-1 language selection at temperature 0.1 and projector temperature 0.07.

Generation. UCIT evaluation uses greedy decoding, at most 128 new tokens, and a no-repeat 3-gram constraint. Image and prompt features select their respective adapters; the language selection is held fixed throughout each answer. Algorithm 1 summarizes the method.

C Metric Definitions

Tables 7 and 8 in the main text report performance after every task or domain for the models in Tables 2 and 4. Let $A_{t,i}$ denote the score on task i after learning task t, and let $\bar{A}_t = t^{-1} \sum_{i=1}^{t} A_{t,i}$.

The aggregate metrics are

$$\begin{aligned} \mathrm{MFT} &= \frac{1}{T}\sum_{i=1}^{T} A_{i,i}, \quad \mathrm{MFN} = \frac{1}{T}\sum_{i=1}^{T} A_{T,i}, \\ \mathrm{MAA} &= \frac{1}{T}\sum_{t=1}^{T} \bar{A}_t, \quad \mathrm{BWT} = \frac{1}{T-1}\sum_{i=1}^{T-1}(A_{T,i} - A_{i,i}). \end{aligned} \tag{11}$$

MFT measures initial learning, MFN measures final performance, and MAA averages performance throughout the sequence. BWT measures the change on earlier tasks since they were first learned. Stage means and aggregate metrics are calculated before rounding. UCIT combines accuracy and CaptionAvg without rescaling, following its benchmark protocol. The component trajectories in Figure 3 use the common ablation configuration in Table 3.

D Projector Restoration

We test whether changing the visual projector can recover earlier answers while keeping the later language model unchanged. For a transition from task $t-1$ to task t, we select examples that the earlier model answered correctly under both the original prompt and a prompt that explicitly states the required answer format, but that the later model answered incorrectly under the original prompt. Writing $a = 1$ for a correct answer, the selected set for earlier task i is

$$\mathcal{F}_{i,t} = \{z : a^{\mathrm{orig}}(\theta_{t-1}, z) = 1,\ a^{\mathrm{fmt}}(\theta_{t-1}, z) = 1,\ a^{\mathrm{orig}}(\theta_t, z) = 0\}. \tag{12}$$

We restore both linear transformations of the earlier projector, including their biases, and evaluate the selected examples again. Recovery is the proportion answered correctly after restoration. Table 11 includes all five evaluated transitions; Figure 1a displays the ImageNet-R and IconQA comparisons. Each example–transition pair is counted once.

Table 11: **Recovery after restoring the earlier projector.** For transition $T_{t-1} \to T_t$, the projector uses its parameters before task t, while the LLM keeps its parameters after task t. Counts refer to example–transition pairs.

			Case-transition counts		
Learner	**Transition**	**Old task**	Forgotten	Recovered	Rate (%)
LoRA-FT	$T_2 \to T_3$	IN-R	334	247	73.95
	$T_3 \to T_4$	AQA	544	47	8.64
	$T_4 \to T_5$	IQ	300	217	72.33
O-LoRA	$T_1 \to T_2$	IN-R	95	91	95.79
	$T_4 \to T_5$	IQ	272	231	84.93
Pooled, same intervention scope			1545	833	53.92

E Adapter Selection

We evaluate task-wise adapter selection in the full method, using the component configuration and seed 42.

Each task has a fixed set of 3,000 examples. Inputs use the Vicuna-v1 conversation template and CLIP center-crop preprocessing at resolution 336, with float16 inference and the generation settings in Appendix B. Answer matching strips surrounding whitespace and uppercases both prediction and reference. All model parameters and task features remain fixed across selection conditions.

Task-wise adapter comparisons. Tables 12 and 13 compare the model's selection with each retained task-wise adapter after VizWiz and after all tasks. A named adapter is selected at every

adapted language linear; the visual mapping stays unchanged. The corresponding task-wise adapter uses task identity for this diagnostic comparison. It gives the highest final score among the six adapters for every evaluated task in the full model. After VizWiz, selecting the ImageNet-R adapter also recovers 426 of 445 ImageNet-R answers that were correct at task 2 but incorrect with model selection at task 3.

Table 12: **Language-adapter selection at T3.** Accuracy (%) on 3,000 examples per task with unchanged visual mapping. Results are for the full method. Each named adapter was learned on that task and is selected at every adapted language linear. Green cells mark the corresponding task-wise adapter.

Selection	IN-R	AQA
Model-selected	76.30	93.97
Uniform mixture	74.37	91.47
ImageNet-R adapter	90.87	34.87
ArxivQA adapter	0.07	93.97
VizWiz adapter	0.00	2.67

Table 13: **Language-adapter selection at T6.** Accuracy (%) on 3,000 examples per task with unchanged visual mapping. Results are for the full method. Each named adapter was learned on that task and is selected at every adapted language linear. Green cells mark the corresponding task-wise adapter.

Selection	IN-R	AQA	IQ	CM
Model-selected	88.73	94.57	66.20	74.80
Uniform mixture	51.60	78.17	60.10	37.80
ImageNet-R adapter	90.50	47.73	9.03	19.13
ArxivQA adapter	0.77	94.57	2.40	19.17
VizWiz adapter	0.00	31.53	21.40	20.10
IconQA adapter	2.10	0.00	73.90	19.63
CLEVR-Math adapter	20.10	58.80	30.57	75.33
Flickr30k adapter	13.63	53.50	18.37	20.23

Further recovery analyses and examples are provided in Appendix I.

F Pretrained-Task Evaluation

Table 5 in the main text compares the full method and w/o PSG on the same 2,374 MME (Fu et al., 2025) and 4,377 MMBench (Liu et al., 2025) examples before continual learning, after VizWiz (T3), and after the full sequence (T6). We select the candidate with the highest mean conditional log-probability over its answer tokens: yes/no for MME and the available option letters for MMBench. Routing and scoring are identical across conditions. MME uses its official total score; MMBench uses option-letter accuracy.

G Analysis of the Gradient Constraint

Following Jaeger (Jaeger, 2017), we first state the Conceptor spectral form and then derive its gradient response for the parameter references used by HMA and PSG.

G.1 Conceptor Form and Spectrum

Lemma 1 (Standard Conceptor representation). *Let $X \in \mathbb{R}^{n\times d}$, $n > 0$, $a > 0$, and $\epsilon > 0$. Set $R = X^\top X/n$ and $\gamma = a^{-2} + \epsilon/n$. Then*

$$X^\top [XX^\top + (na^{-2} + \epsilon)I_n]^{-1} X = R(R + \gamma I_d)^{-1}. \tag{13}$$

If $R = V \operatorname{diag}(s_k)V^\top$ is an eigendecomposition, then $C_X = V \operatorname{diag}(s_k/(s_k+\gamma))V^\top$. Consequently $0 \preceq C_X \prec I_d$, with zero response precisely on $\ker X$.

Proof. Write the reduced singular value decomposition as $X = U\Sigma V_q^\top$, with positive singular values σ_k. The left-hand side equals $V_q \operatorname{diag}(\sigma_k^2/(\sigma_k^2 + n\gamma))V_q^\top$. The nonzero eigenvalues of R are $s_k = \sigma_k^2/n$, and both expressions vanish on $\ker X$. Completing V_q to an orthonormal basis establishes the identity and the eigenvalue bounds. □

G.2 Response Attenuation in the Reference Directions

Proof of Proposition 1. For $G \in \mathbb{R}^{k\times d}$, $E_X(G) = \operatorname{tr}(GRG^\top)$. Let $F = I - \rho C_X$. Lemma 1 shows that F and R share eigenvectors, with $f_k = 1 - \rho s_k/(s_k + \gamma)$. Hence

$$E_X(GF) = \operatorname{tr}(GFRFG^\top) = \sum_k s_k f_k^2 \|Gv_k\|_2^2, \tag{14}$$

$$E_X(G) - E_X(GF) = \sum_k s_k(1 - f_k^2)\|Gv_k\|_2^2 \geq 0. \tag{15}$$

The inequality follows from $1 - \rho \leq f_k \leq 1$. If $s_k = 0$, then $f_k = 1$ and $GFv_k = Gv_k$. For $\rho > 0$, attenuation is strict whenever G has a nonzero component in a reference direction with $s_k > 0$. □

Application to HMA and PSG. For HMA's A factors, X has the projector input dimension; for its B factors, X has the low-rank dimension. PSG uses the language linear's input dimension. The proof applies in each coordinate space. The reference is defined by the stored factorization, and the attenuation statement concerns each factor gradient before the optimizer step.

For an SGD update, the same response identity scales by the square of the learning rate. With AdamW, the proposition describes the filtered gradient supplied to the optimizer, before moment accumulation and coordinate-wise rescaling.

H Computational Cost

Table 6 reports adapter parameters and retained storage for UCIT, excluding the frozen backbone. The parameter count after t tasks is $240.07t + 109.31$ million: task-wise factors grow with the sequence, while the shared factors have fixed size. Each task trains 240.07M task-wise parameters and 109.31M shared parameters, for 349.39M in total. Conceptor references are used only during training.

Measured training time. The full six-task sequence takes 15.49 hours on two GPUs (30.97 GPU-hours) under the component configuration in Table 3, with effective batch size 64 and seed 42. This measures training and excludes initial model loading and separate evaluation.

Gradient filtering. For a reference $X \in \mathbb{R}^{n\times d}$, we evaluate the constraint in dual form to avoid a dense $d \times d$ Conceptor:

$$D = \left[XX^\top + (na^{-2} + \epsilon)I_n\right]^{-1} X, \qquad \widetilde{G} = G - \rho(GX^\top)D. \tag{16}$$

Computing D once costs $O(n^2 d + n^3)$; filtering each $G \in \mathbb{R}^{k\times d}$ then costs $O(knd)$. Storing X and D requires $2nd$ scalars, or 67.1 MB in float32 for $n = 2048$ and $d = 4096$. This is the reference storage per filter.

I Recovery Analyses and Examples

Figure 4 examines how retained adapters affect earlier-task performance. For ImageNet-R, 445 answers are correct after task 2 but incorrect after task 3. Selecting the ImageNet-R language adapter at task 3 recovers 426 of these answers; 423 are answered correctly again with the model's selection at task 4. The comparison uses the fixed evaluation examples in Appendix E.

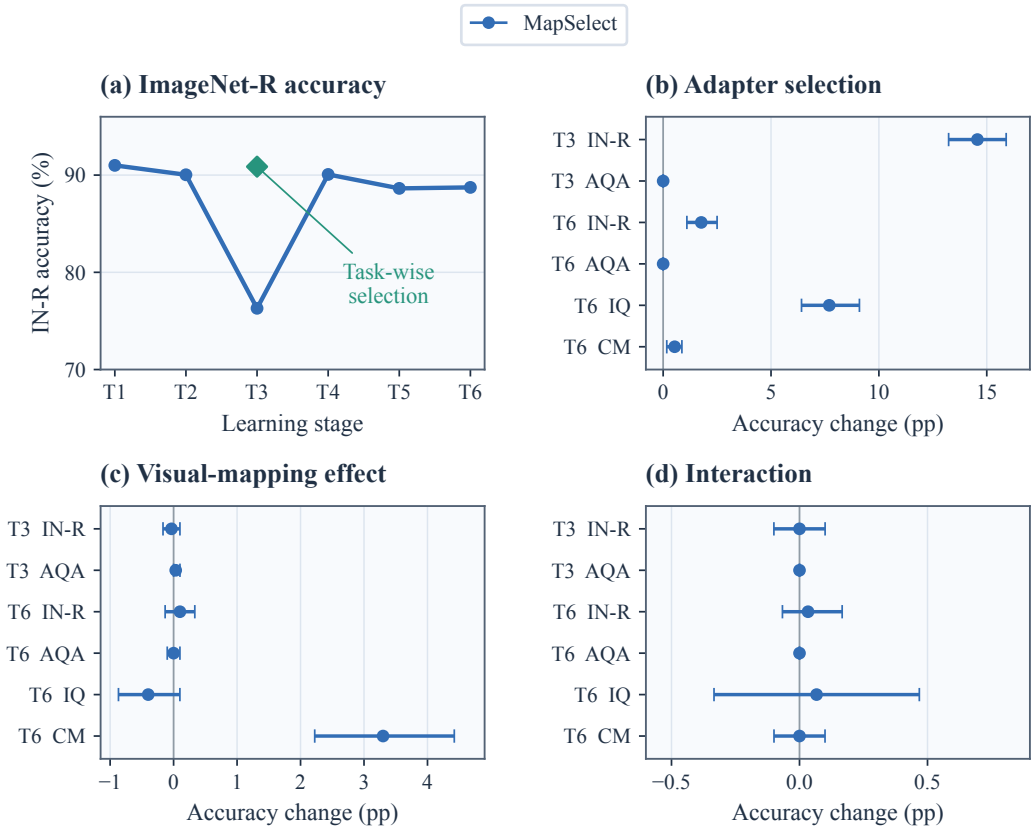

Figure 4: **Recovery through adapter selection and visual mapping.** (a) ImageNet-R accuracy during learning; the diamond selects the ImageNet-R adapter after task 3, keeping the visual mapping unchanged. (b) Accuracy change when selecting the evaluated task's language adapter instead of the model's selection. (c) Accuracy change when selecting the highest-weight projector adapter instead of the weighted projector combination, holding language-adapter weights fixed. (d) Interaction between the projector change in (c) and the task-wise language-adapter change in (b), measured as a difference of differences. Error bars are paired 95% bootstrap intervals over evaluation examples. Results are for the full method with prompt-based selection. T3 follows VizWiz; T6 follows all six tasks. IN-R: ImageNet-R; AQA: ArxivQA; IQ: IconQA; CM: CLEVR-Math.



Figure 5: **Examples of recovered answers.** (a) Restoring the earlier visual projector corrects IconQA answers in LoRA-FT and O-LoRA after learning CLEVR-Math; language-model parameters stay fixed. (b) Keeping language-adapter selection fixed throughout an answer corrects two answers from the w/o SAR model after all six tasks; the comparison changes selection at inference, without retraining. (c) With the same yes/no candidate-scoring rule, the full method answers two MME questions correctly after VizWiz while the model without PSG does not. Gold denotes the reference answer. These are selected examples of the respective comparisons, rather than a shared evaluation subset.