# Hidden Forgetting in Continual Multimodal Learning: When Accuracy Survives but Grounding Fails

**Qianyu Chen**[1] **Runxuan Tang**[1] **Canran Xiao**[2*]
[1]Nanyang Technological University
[2]Shenzhen Campus of Sun Yat-sen University

## Abstract

Multimodal large language models continually adapt to new tasks and domains. Standard continual learning metrics track whether earlier answers remain correct and give little insight into the stability of multimodal grounding. We examine whether continually adapted models retain correct answers and continue to use visual, textual, OCR, chart, and document evidence. We identify *hidden evidence-use forgetting*, in which a model retains answer accuracy while shifting to different or less grounded evidence channels, and propose RCL, a replay-free reliance-constrained continual learning framework. RCL freezes the previous checkpoint as a behavioral reference, estimates teacher and student evidence-reliance profiles through counterfactual channel interventions, and jointly optimizes task learning, prediction preservation, and reliance preservation without adding inference-time cost. Across CoIN, COAST, MCITlib, and an evidence-sensitive multimodal stream, RCL consistently improves final performance and reduces forgetting over replay-free, PEFT, routing, and memory-assisted baselines, while substantially lowering modality reliance drift, dominant evidence flips, and hidden forgetting rates. These results suggest that robust continual multimodal learning requires preserving the evidence paths that support correct answers.

## 1 Introduction

Multimodal large language models (MLLMs) support visual question answering, document understanding, chart reasoning, embodied perception, and open-ended image–language interaction. In deployment, new domains, instructions, visual formats, and reasoning skills arrive sequentially. Adapting to these changes can degrade previously acquired abilities, limiting reliability in long-lived and safety-sensitive applications.

Recent studies have begun to formalize this problem as continual vision–language learning and continual multimodal instruction tuning. CLiMB highlights that vision–language systems suffer nontrivial forgetting across sequential multimodal tasks [Srinivasan et al., 2022], while ZSCL and CTP show that continual adaptation can damage the zero-shot transfer and cross-modal topology of pretrained vision–language representations [Zheng et al., 2023, Zhu et al., 2023]. More recent MLLM-oriented benchmarks, such as CoIN and Continual LLaVA, further demonstrate that instruction-tuned MLLMs are vulnerable to capability-level forgetting when exposed to evolving multimodal instruction streams [Chen et al., 2024a, Cao et al., 2024].

Existing approaches have made substantial progress by preserving model parameters, output behavior, or task-specific adaptation modules. For example, Model Tailor mitigates forgetting by identifying sparse parameter modifications that preserve pretrained capabilities [Zhu et al., 2024], while HiDe-LLaVA, BranchLoRA, and ProgLoRA organize task-specific or progressive LoRA modules to reduce

*Corresponding author: xiaocr3@mail.sysu.edu.cn.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

interference during continual instruction tuning [Guo et al., 2025a, Zhang et al., 2025, Yu et al., 2025]. Other methods improve continual multimodal adaptation through prompt routing, expert allocation, or synthetic replay signals, as seen in ModalPrompt, CL-MoE, and GIFT [Zeng et al., 2025, Huai et al., 2025, Wu et al., 2025]. SEFE distinguishes superficial response-format degradation from more essential knowledge loss [Chen et al., 2025]. Despite these advances, most methods still evaluate and constrain forgetting mainly through final answers, output distributions, or parameter changes.

Two multimodal checkpoints may give the same answer while relying on different sources, such as visual regions, OCR tokens, or language priors. Accuracy can then appear stable even as the model relies more on shortcuts. Prior work on language bias in VQA shows that dataset priors can yield correct predictions even when the model makes little use of visual evidence [Agrawal et al., 2018], and counterfactual analysis has been used to reveal causal effects of biased evidence in static VQA settings [Niu et al., 2021]. How these shifts arise during replay-free continual learning for MLLMs remains underexplored.

We study whether continual MLLM adaptation can preserve correct answers and the use of appropriate multimodal evidence. Our framework uses the previous model as a reference, estimates how both models rely on available evidence channels, and constrains the current model to preserve that reliance profile while learning new tasks. This complements output and parameter preservation by addressing evidence-use drift that answer-level evaluation can miss.

We make three contributions:

- We identify and formalize *hidden evidence-use forgetting*: an MLLM may retain the correct answer while drifting toward different or less grounded evidence sources during continual adaptation.
- We propose a replay-free reliance-preserving continual learning framework that constrains evidence-use drift without storing previous multimodal examples or increasing inference-time cost.
- Through continual multimodal instruction streams, we show that preserving reliance profiles improves both answer retention and evidence-grounded stability, providing a stronger notion of continual multimodal robustness than answer accuracy alone.

## 2 Related Work

**Continual vision–language representation learning.** Early multimodal continual learning studies moved beyond unimodal classification by evaluating sequential vision–language tasks and cross-modal representation stability. CLiMB introduced a benchmark for continual vision-and-language tasks, while Mod-X, ZSCL, and CTP showed that sequential adaptation can distort intra-modal geometry, cross-modal alignment, and zero-shot transfer in CLIP-like models [Srinivasan et al., 2022, Ni et al., 2023, Zheng et al., 2023, Zhu et al., 2023]. More recent work scales this direction to practical multimodal streams and stronger VLM backbones: FoMo-in-Flux studies continual multimodal pretraining under realistic compute and data-mixture constraints [Udandarao et al., 2024]; C-CLIP evaluates continual image–text retrieval and zero-shot retention [Liu et al., 2025]; LADA introduces label-specific CLIP adapters [Luo et al., 2025]; and GIFT uses synthetic image–text data for replay-style preservation [Wu et al., 2025]. Pi-CCA preserves image–text alignment by matching canonical-correlation statistics [Zhang et al., 2026d], and COMPO-REALIGN targets compositional forgetting with a reversible primitive-to-composition mapping [Xiao et al., 2026a]. In autoregressive MLLMs, we study whether correct answers can persist even as instance-level reliance on visual, OCR, chart, or language evidence shifts.

**Continual instruction tuning for MLLMs.** Continual instruction tuning has recently become the dominant setting for adapting MLLMs to evolving tasks. CoIN formulates sequential instruction tuning over diverse task categories and shows that MLLMs suffer severe forgetting of instruction following [Chen et al., 2024a]. Continual LLaVA further introduces COAST, covering domain-, capability-, and dataset-incremental LVLM adaptation [Cao et al., 2024]. Recent methods then focus on stronger parameter organization and routing: Model Tailor identifies sparse parameter patches to preserve pretrained capabilities [Zhu et al., 2024]; HiDe-LLaVA uses task-specific expansion and task-general fusion based on layer-wise similarity [Guo et al., 2025a]; BranchLoRA and ProgLoRA

improve LoRA allocation under MCIT [Zhang et al., 2025, Yu et al., 2025]; ModalPrompt uses dual-modality guided prompt fusion and selection [Zeng et al., 2025]; CL-MoE introduces dual momentum experts for continual VQA [Huai et al., 2025]; and MLLM-CL proposes domain/ability continual settings with MLLM-based routing [Zhao et al., 2025]. SEFE separates superficial format forgetting from essential knowledge forgetting [Chen et al., 2025]. These approaches primarily preserve answer accuracy, response format, parameter specialization, or output distributions. Our preliminary study suggests that evidence use can shift even when these properties are preserved. RCL adds a complementary constraint by using the previous checkpoint's counterfactual reliance profile as a training target.

**Grounding, shortcut bias, and counterfactual evidence use.** A related line of VQA research shows that language priors can yield correct answers even when a model overlooks visual evidence. VQA-CP exposes this issue by changing answer priors between training and testing [Agrawal et al., 2018], and Counterfactual VQA uses causal counterfactual reasoning to estimate and reduce language-bias effects [Niu et al., 2021]. These studies motivate channel-level evidence diagnostics and focus mainly on static debiasing or evaluation. In autonomous driving, Hint-AD grounds language explanations in the intermediate outputs of perception, prediction, and planning [Ding et al., 2025]. In optical transport networks, graph attention helps correlate alarms and identify root alarms [Xiao et al., 2025], while YANG schemas ground an LLM agent for configuring equipment from different vendors [Xiao et al., 2026c]. Our work brings this perspective into replay-free MLLM continual learning by turning counterfactual evidence suppression into a training-time reliance-preservation objective, while adding no inference-time overhead.

## 3 Preliminary Study

We examine how continual adaptation affects answer accuracy and the use of multimodal evidence. This distinction matters because two checkpoints may produce the same answer while relying on different channels, such as visual regions, OCR tokens, chart elements, or language priors. We refer to evidence-use drift despite answer retention as *hidden forgetting*.

**Diagnostic setup.** We conduct a replay-free pilot study on an eight-stage multimodal stream and evaluate two simple baselines: sequential LoRA tuning and output-level answer distillation.

After training, we perform post-hoc channel ablations to estimate a diagnostic evidence-use vector $\boldsymbol{q}(f; x, y)$ for each checkpoint $f$, where $q_m$ is proportional to the loss increase caused by suppressing evidence channel $m$. Full protocol details are provided in Appendix B.1.

Let $f_j$ be the checkpoint after learning stage $j$, and $f_k$ a later checkpoint after stage $k > j$. For an old evaluation set $\mathcal{E}_j$, we measure answer forgetting, evidence-sensitivity drift, and answer-preserved dominant-evidence flips as

$$\begin{aligned} A_{j\to k} &= \mathrm{Acc}(f_j; \mathcal{E}_j) - \mathrm{Acc}(f_k; \mathcal{E}_j), \\ G_{j\to k} &= \mathbb{E}_{(x,y)\sim\mathcal{E}_j}\left[\|\boldsymbol{q}(f_j; x, y) - \boldsymbol{q}(f_k; x, y)\|_1\right], \\ H_{j\to k} &= \mathbb{E}\left[\mathbf{1}\left[\arg\max_m q_m(f_j; x, y) \neq \arg\max_m q_m(f_k; x, y)\right] \,\middle|\, f_j(x) = f_k(x) = y\right]. \end{aligned} \tag{1}$$

$A_{j\to k}$ measures answer degradation, while $G_{j\to k}$ and $H_{j\to k}$ characterize changes in evidence use.

**Findings.** Figure 1 shows substantial evidence-use drift even when answers are retained. Many transitions incur less than $2\%$ clean accuracy drop while exhibiting large evidence-sensitivity drift. Output-level distillation reduces the average clean accuracy drop from $2.72\%$ to $1.57\%$. The hidden-drift region still contains $66.3\%$ of diagnostic slices under output distillation, compared with $14.4\%$ under sequential LoRA. The effect is most pronounced on OCR-, chart-, and document-centric stages, where correct answers often coincide with a shift from grounded visual or textual evidence toward easier priors.

These observations motivate preserving both answers and the evidence that supports them during continual adaptation.

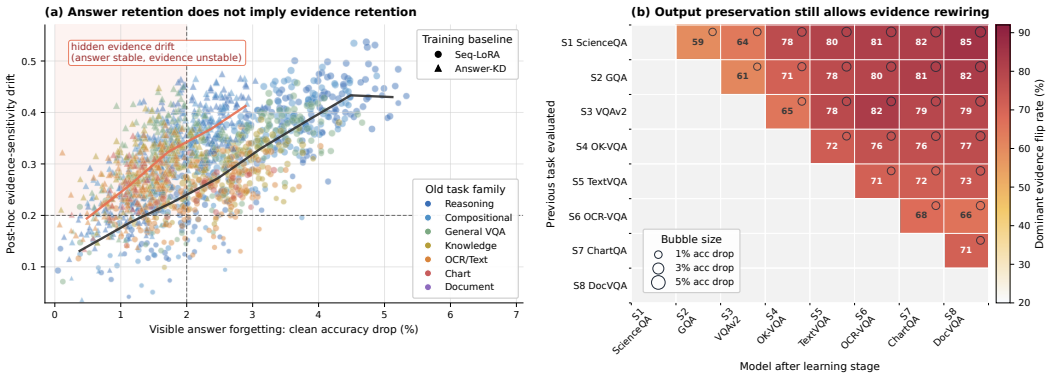


Figure 1: **Hidden evidence-use drift in replay-free multimodal continual adaptation.** **(a)** Answer forgetting and post-hoc evidence drift are weakly aligned. The shaded region marks small answer drops with large evidence drift. **(b)** For Answer-KD, warm cells with small bubbles show that output preservation can hide substantial dominant-evidence flips. Details are in Appendix B.1.

# 4 Method

We propose **Reliance-Constrained Continual Learning** (RCL), a replay-free framework for mitigating hidden forgetting in continual multimodal learning. At each stage, RCL freezes the previous model as a reference and uses counterfactual channel interventions to estimate the modality-reliance profiles of both models. It updates only the adaptation parameters by jointly optimizing task learning, prediction preservation, and reliance preservation. The framework adds no computation at inference time.

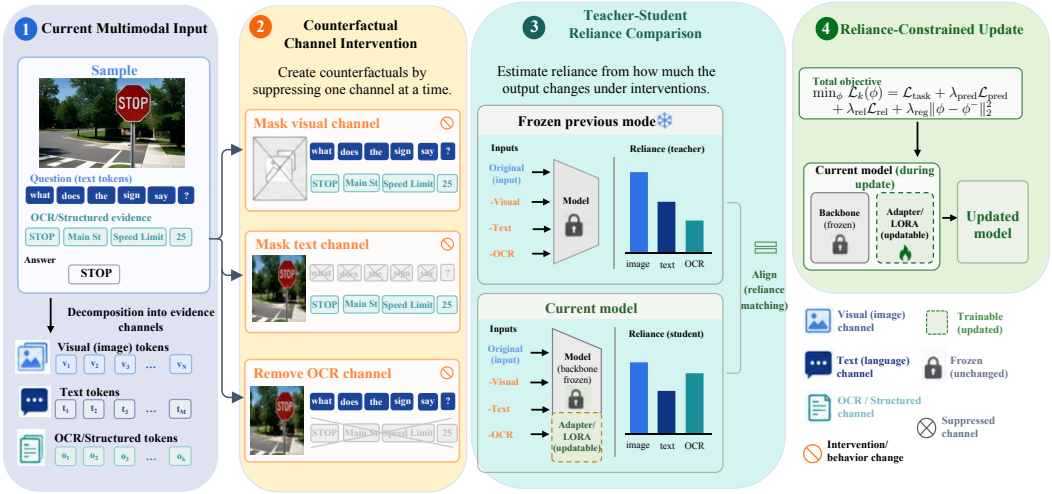


Figure 2: **Overview of Reliance-Constrained Continual Learning.** Given a multimodal input, RCL constructs counterfactual channel interventions, compares teacher–student reliance profiles, and updates the current model with a reliance-preserving objective to mitigate hidden evidence-use forgetting.

## 4.1 Continual Multimodal Learning Setup

We consider a sequence of multimodal learning stages $\{\mathcal{D}_1, \ldots, \mathcal{D}_K\}$, where each stage $\mathcal{D}_k$ contains input-output pairs $(x, y)$. Each input $x$ consists of a set of evidence channels $\mathcal{M}_x$, such as visual

tokens, question tokens, OCR tokens, chart elements, and other task-specific input sources. We write $x = \{x^m : m \in \mathcal{M}_x\}$, where $x^m$ denotes the component of $x$ associated with channel $m$.

Let $f_\theta$ denote an MLLM with autoregressive answer distribution $p_\theta(y \mid x) = \prod_{\ell=1}^{L} p_\theta(y_\ell \mid y_{<\ell}, x)$. We use parameter-efficient adaptation and decompose the model parameters as $\theta = \theta_0 \cup \phi$, where $\theta_0$ is the frozen backbone and $\phi$ denotes trainable adapter parameters. At stage $k$, the previous model is frozen as $f_{\theta^-}$ with $\theta^- = \theta_{k-1}$, and the current model $f_\theta$ is optimized on $\mathcal{D}_k$.

The standard task loss is

$$\mathcal{L}_{\text{task}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_k}\left[-\sum_{\ell=1}^{L} \log p_\theta(y_\ell \mid y_{<\ell}, x)\right]. \tag{2}$$

The task loss supervises current-stage answers and leaves evidence selection unconstrained.

### 4.2 Counterfactual Modality Reliance

RCL estimates modality reliance through counterfactual evidence suppression. For each channel $m \in \mathcal{M}_x$, we define an intervention set $\mathcal{I}_m$. Each intervention $\tau_m \in \mathcal{I}_m$ suppresses access to channel $m$ while keeping the remaining channels unchanged. For example, an image channel is suppressed by masking visual regions, an OCR channel by removing recognized text tokens, and a chart channel by masking chart elements. The complete intervention construction is given in Appendix A.1.

For a model $f_\theta$, we define the counterfactual sensitivity of channel $m$ as

$$\begin{aligned}\Delta_{\theta,m}(x,y) = \frac{1}{|\mathcal{I}_m|}\sum_{\tau_m\in\mathcal{I}_m}\Bigg[\frac{1}{L}\sum_{\ell=1}^{L} D_{\text{KL}}\left(p_\theta(\cdot \mid y_{<\ell}, x) \,\|\, p_\theta(\cdot \mid y_{<\ell}, \tau_m(x))\right) \\ + \beta\left(\ell_\theta(y \mid \tau_m(x)) - \ell_\theta(y \mid x)\right)_+\Bigg],\end{aligned} \tag{3}$$

where $\ell_\theta(y \mid x) = -\sum_{\ell=1}^{L} \log p_\theta(y_\ell \mid y_{<\ell}, x)$, $(z)_+ = \max(z, 0)$, and $\beta \geq 0$ balances output-distribution shift and answer-loss increase. A larger $\Delta_{\theta,m}(x,y)$ indicates that suppressing channel $m$ causes a larger behavioral change, hence the model relies more strongly on that channel for the given sample.

We normalize channel sensitivities into a reliance vector:

$$r_{\theta,m}(x,y) = \frac{\exp\left(\Delta_{\theta,m}(x,y)/\tau_r\right)}{\sum_{n\in\mathcal{M}_x}\exp\left(\Delta_{\theta,n}(x,y)/\tau_r\right)}, \qquad \boldsymbol{r}_\theta(x,y) = \left[r_{\theta,m}(x,y)\right]_{m\in\mathcal{M}_x}. \tag{4}$$

Here $\tau_r > 0$ is a temperature parameter. The vector $\boldsymbol{r}_\theta(x,y)$ lies in the simplex $\Delta^{|\mathcal{M}_x|-1}$ and is always computed over the evidence channels available in the current sample. Teacher and student reliance vectors are therefore compared on the same support $\mathcal{M}_x$.

### 4.3 Hidden Forgetting as Reliance Drift

Accuracy tracks whether answers change. We quantify changes in evidence use through modality reliance drift. Given the frozen reference model $f_{\theta^-}$ and the current model $f_\theta$, the sample-level reliance drift is

$$\text{MRD}(x, y; \theta^-, \theta) = D_{\text{JS}}\left(\boldsymbol{r}_{\theta^-}(x,y) \,\|\, \boldsymbol{r}_\theta(x,y)\right), \tag{5}$$

where $D_{\text{JS}}$ denotes Jensen–Shannon divergence.

Let $\mathcal{S}_j$ index the nonempty diagnostic slices $\mathcal{E}_{j,s}$ of old task $j$. On slice $s$, $\Delta_{\text{acc}}^{j\to k,s}$ is the accuracy at checkpoint $j$ minus that at checkpoint $k$, both evaluated on $\mathcal{E}_{j,s}$. The slice drift $\text{MRD}_{j\to k,s}$ is the mean of Eq. (5) over $\mathcal{E}_{j,s}$. The slice-level hidden forgetting indicator is

$$\text{HF}_{j\to k,s} = \mathbf{1}\left[\Delta_{\text{acc}}^{j\to k,s} \leq 2\%\right]\mathbf{1}\left[\text{MRD}_{j\to k,s} \geq 0.20\right]. \tag{6}$$

The accuracy threshold is two percentage points. The final hidden forgetting rate averages the indicators within each old task and then averages over tasks (Eq. (19)).

### 4.4 Reliance-Constrained Continual Update

RCL constrains the current model to preserve the reference model's reliance profile on the current-stage data. For each sample $(x, y)$, the frozen model $f_{\theta^-}$ produces a reference reliance vector $\boldsymbol{r}_{\theta^-}(x, y)$, while the trainable model produces $\boldsymbol{r}_\theta(x, y)$. The reliance preservation loss is

$$\mathcal{L}_{\mathrm{rel}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_k}\left[w_{\theta^-}(x, y) D_{\mathrm{JS}}\left(\mathrm{sg}[\boldsymbol{r}_{\theta^-}(x, y)] \,\|\, \boldsymbol{r}_\theta(x, y)\right)\right], \tag{7}$$

where $\mathrm{sg}[\cdot]$ denotes stop-gradient. The scalar $w_{\theta^-}(x, y) \in [0, 1]$ is a confidence weight computed from the reference model's predictive entropy and intervention sensitivity; its exact form is provided in Appendix A.2. This gate prevents unreliable reference predictions from dominating the update.

We also preserve the output behavior of the previous model through token-level prediction distillation:

$$\mathcal{L}_{\mathrm{pred}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_k}\left[\frac{\tau_{\mathrm{kd}}^2}{L}\sum_{\ell=1}^{L} D_{\mathrm{KL}}\left(p_{\theta^-}^{\tau_{\mathrm{kd}}}(\cdot \mid y_{<\ell}, x) \,\middle\|\, p_\theta^{\tau_{\mathrm{kd}}}(\cdot \mid y_{<\ell}, x)\right)\right], \tag{8}$$

where $p_\theta^{\tau_{\mathrm{kd}}}$ denotes the softmax distribution over logits divided by the temperature $\tau_{\mathrm{kd}}$.

The full objective at stage $k$ is

$$\min_{\phi} \quad \mathcal{L}_k(\phi) = \mathcal{L}_{\mathrm{task}} + \lambda_{\mathrm{pred}}\mathcal{L}_{\mathrm{pred}} + \lambda_{\mathrm{rel}}\mathcal{L}_{\mathrm{rel}} + \lambda_{\mathrm{reg}}\|\phi - \phi^-\|_2^2, \tag{9}$$

Here $\phi^-$ denotes the previous stage's adapter parameters. The prediction term preserves answers, and the reliance term preserves the reference model's allocation of reliance to visual, textual, and OCR evidence for each input. For inputs grounded in visual evidence, RCL discourages drift toward language shortcuts. For inputs where the reference model relies primarily on text, RCL preserves that allocation.

### 4.5 Training and Inference

Algorithm 1 summarizes one training stage. The frozen reference supplies prediction and reliance targets on current-stage data, and training requires no stored examples from previous stages. The updated checkpoint becomes the reference for the next stage.

**Algorithm 1** RCL training at stage $k$

**Require:** Current-stage data $\mathcal{D}_k$, previous checkpoint $f_{\theta_{k-1}}$

1: Set $\theta^- \leftarrow \theta_{k-1}$, freeze $f_{\theta^-}$, and initialize $\theta \leftarrow \theta^-$
2: **for** each training minibatch $\mathcal{B}_k$ drawn from $\mathcal{D}_k$ **do**
3: Sample shared counterfactual inputs $\tau_m(x)$ for each $(x, y) \in \mathcal{B}_k$ and $m \in \mathcal{M}_x$
4: Estimate $\boldsymbol{r}_{\theta^-}(x, y)$ and $\boldsymbol{r}_\theta(x, y)$ on these inputs and compute $w_{\theta^-}(x, y)$
5: Update only $\phi$ using the minibatch objective in Eq. (9)
6: **end for**
7: **return** $f_{\theta_k}$ with $\theta_k \leftarrow \theta$

During inference, RCL uses the updated MLLM $f_{\theta_k}$ directly. Counterfactual interventions, frozen-reference forward passes, and reliance estimation are used only during training and evaluation, so the deployed model has the same inference cost as the underlying PEFT-adapted MLLM.

## 5 Experiments

### 5.1 Experimental Setup

**Benchmarks and metrics.** We evaluate RCL on CoIN, COAST, MCITlib, and an evidence-sensitive stream covering VQA, OCR/document, and chart/table reasoning [Chen et al., 2024a, Cao et al., 2024, Guo et al., 2025b]. We report official task scores, standard CL metrics, and reliance-aware diagnostics; details are in Appendix B.2.

**Comparison protocol.** We use a replay-free, task-id-free protocol: methods store no previous multimodal examples, use no oracle task identity at inference, and follow the same task order and data

Table 1: **Main results against common baselines and recent SoTAs.** Among continual-learning methods, the best results are bolded and the second-best results are underlined.

| Method | Backbone | Extra signal | CoIN | | | COAST | | MCITlib | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | ACC↑ | MAA↑ | BWT↑ | Avg.↑ | Fgt.↓ | MFN↑ | MAA↑ | BWT↑ |
| **Reference settings** | | | | | | | | | | |
| Zero-shot | LLaVA-1.5-7B | None | — | 7.12 | — | — | — | — | — | — |
| Multi-task / Joint | LLaVA-1.5-7B | All tasks | — | 57.18 | — | 52.59 | — | 58.48 | 63.86 | — |
| **Standard continual learning and PEFT baselines** | | | | | | | | | | |
| Seq-LoRA | LLaVA-1.5-7B | None | 28.74 | 32.97 | -32.62 | 30.29 | 20.74 | 50.22 | 58.09 | -13.41 |
| EWC | LLaVA-1.5-7B | Fisher | 32.90 | 36.93 | -27.46 | 32.25 | 19.16 | 49.51 | 56.38 | -12.68 |
| LwF | LLaVA-1.5-7B | Logit distill. | 30.41 | 34.95 | -27.03 | 32.54 | 20.59 | 49.10 | 56.25 | -13.05 |
| L2P | LLaVA-1.5-7B | Prompt | 41.86 | 46.72 | -18.49 | 45.79 | 6.69 | 50.99 | 57.60 | -8.59 |
| DualPrompt | LLaVA-1.5-7B | Prompt | 42.22 | 47.05 | -18.17 | 46.27 | 6.61 | 51.31 | 57.77 | -8.36 |
| CODA-Prompt | LLaVA-1.5-7B | Prompt | 43.50 | 48.26 | -17.29 | 46.85 | 5.80 | 52.00 | 58.16 | -7.80 |
| O-LoRA | LLaVA-1.5-7B | Orth. LoRA | 45.83 | 50.26 | -16.58 | 46.72 | 5.79 | 52.35 | 58.52 | -9.98 |
| **Recent multimodal continual instruction tuning methods** | | | | | | | | | | |
| MoELoRA | LLaVA-1.5-7B | Expert LoRA | 37.13 | 42.76 | -25.91 | 45.01 | 7.23 | 49.88 | 57.64 | -10.70 |
| Model Tailor | LLaVA-1.5-7B | Sparse patch | 46.67 | 51.22 | -15.12 | 45.56 | 5.17 | 53.05 | 58.98 | -7.48 |
| Continual LLaVA | LLaVA-1.5-7B | Incr. embed. | 48.09 | 52.80 | -13.27 | 48.73 | 4.54 | 53.26 | 59.51 | -6.73 |
| CL-MoE | LLaVA-1.5-7B | MoE | 48.90 | 53.48 | -11.92 | 47.72 | 5.06 | 49.18 | 57.21 | -12.65 |
| HiDe-LLaVA | LLaVA-1.5-7B | Decoupling | 51.56 | 56.41 | -8.97 | 48.30 | 4.68 | 50.18 | 57.53 | -7.58 |
| BranchLoRA | LLaVA-1.5-7B | Router | 44.20 | 49.94 | -20.98 | 49.11 | 4.56 | 54.97 | 60.21 | -5.58 |
| ProgLoRA | LLaVA-1.5-7B | Task recall | 59.09 | 62.38 | -6.59 | 49.79 | 4.11 | 55.78 | 60.55 | -4.94 |
| ModalPrompt | LLaVA-1.5-7B | Prompt | 50.52 | 55.55 | -10.10 | 47.95 | 4.80 | 46.73 | 47.17 | **0.02** |
| SEFE | LLaVA-1.5-7B | ASD+RegLoRA | 53.61 | 58.17 | -8.55 | 49.36 | 4.31 | 54.26 | 59.66 | -7.94 |
| MLLM-CL | LLaVA-1.5-7B | Router | 55.51 | 60.28 | -7.17 | 49.90 | 3.97 | 55.32 | 60.76 | -4.93 |
| **Replay or memory-assisted references** | | | | | | | | | | |
| Rehearsal | LLaVA-1.5-7B | Buffer | 42.88 | 47.25 | -18.43 | 31.58 | 19.52 | 51.56 | 57.19 | -8.11 |
| DER | LLaVA-1.5-7B | Buffer+KD | 44.35 | 48.76 | -16.76 | 33.88 | 17.94 | 52.52 | 57.88 | -7.67 |
| GIFT | LLaVA-1.5-7B | Synthetic replay | 56.26 | 61.51 | -6.89 | 49.76 | 4.07 | 56.06 | 60.72 | -4.74 |
| Ask-and-Remember | LLaVA-1.5-7B | Question replay | 57.22 | 62.07 | -6.72 | 50.02 | 3.90 | 56.35 | 60.88 | -4.56 |
| **Ours** | | | | | | | | | | |
| RCL | LLaVA-1.5-7B | Reliance | 65.01 | 67.56 | -4.23 | 52.55 | 2.76 | 58.86 | 62.29 | -2.03 |
| RCL | LLaVA-1.5-13B | Reliance | **66.87** | **69.30** | **-3.77** | **54.22** | **2.30** | **60.18** | **63.42** | <u>-1.58</u> |
| RCL | InternVL-Chat-7B | Reliance | <u>66.08</u> | <u>68.45</u> | <u>-3.92</u> | <u>53.50</u> | <u>2.47</u> | <u>59.62</u> | <u>62.85</u> | -1.79 |

budget. We compare standard CL and PEFT baselines with recent continual multimodal instruction-tuning methods. We report replay and memory-assisted methods separately because they use stored or generated past-task signals. Appendix B.3 details the baseline groups and matched settings.

**Backbones and implementation.** We use LLaVA-1.5-7B as the default backbone and additionally evaluate LLaVA-1.5-13B and InternVL-Chat-7B to test scale and architecture robustness [Liu et al., 2024, Chen et al., 2024b]. Unless otherwise specified, the vision encoder and language backbone are frozen and only PEFT parameters are updated. All methods are trained with matched data, task order, trainable-parameter budget, and optimization budget whenever possible. For RCL, counterfactual interventions are used only during training and diagnostic evaluation, so the final model has the same inference cost as the underlying PEFT-adapted MLLM. Implementation details, hyperparameters, compute resources, and reproducibility settings are provided in Appendix B.4.

### 5.2 Main Results

Table 1 shows that RCL improves both final performance and retention. With the same LLaVA-1.5-7B backbone, RCL raises CoIN accuracy from ProgLoRA's $59.09$ to $65.01$ and improves backward transfer from $-6.59$ to $-4.23$. On COAST, it raises the average score from $50.02$ to $52.55$ and lowers forgetting from $3.90$ to $2.76$ compared with Ask-and-Remember. On MCITlib, RCL improves final performance (MFN) from $56.35$ to $58.86$ against the same reference. The larger LLaVA model achieves the highest final scores among the three RCL configurations, and the InternVL model also exceeds the scores of the default backbone.

Table 2 shows that strong answer performance can coexist with substantial reliance drift. With LLaVA-1.5-7B, Ask-and-Remember achieves an average answer score of $61.90$ while the dominant evidence

Table 2: **Reliance-aware results on the evidence-sensitive stream.** Avg. is final average answer performance; AF is average forgetting; MRD is modality reliance drift; DEF is dominant evidence flip rate on answer-preserved examples; HFR is hidden forgetting rate; GCR measures grounded-channel retention for visual/OCR/chart/document evidence. The best results are bolded and the second-best results are underlined.

| **Method** | **Backbone** | Avg.↑ | AF↓ | MRD↓ | DEF↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|---|---|
| EWC | LLaVA-1.5-7B | 51.96 | 16.06 | 0.257 | 36.7 | 54.0 | 67.4 |
| LwF | LLaVA-1.5-7B | 50.88 | 16.45 | 0.250 | 36.1 | 52.9 | 67.1 |
| O-LoRA | LLaVA-1.5-7B | 54.30 | 12.60 | 0.234 | 33.9 | 46.7 | 70.7 |
| MoELoRA | LLaVA-1.5-7B | 53.62 | 13.22 | 0.223 | 32.6 | 45.4 | 71.3 |
| Continual LLaVA | LLaVA-1.5-7B | 56.53 | 9.86 | 0.209 | 29.3 | 38.6 | 74.7 |
| HiDe-LLaVA | LLaVA-1.5-7B | 57.26 | 9.38 | 0.196 | 28.0 | 36.3 | 76.0 |
| BranchLoRA | LLaVA-1.5-7B | 58.58 | 8.95 | 0.191 | 26.8 | 35.0 | 77.3 |
| ProgLoRA | LLaVA-1.5-7B | 61.94 | 5.87 | 0.176 | 24.1 | 28.8 | 79.6 |
| ModalPrompt | LLaVA-1.5-7B | 56.99 | 9.12 | 0.184 | 25.0 | 31.4 | 78.2 |
| SEFE | LLaVA-1.5-7B | 59.41 | 8.10 | 0.173 | 23.3 | 27.6 | 80.3 |
| MLLM-CL | LLaVA-1.5-7B | 60.29 | 7.02 | 0.167 | 22.7 | 26.3 | 81.2 |
| GIFT | LLaVA-1.5-7B | 61.62 | 6.06 | 0.162 | 21.9 | 24.7 | 82.6 |
| Ask-and-Remember | LLaVA-1.5-7B | 61.90 | 5.80 | 0.159 | 21.6 | 24.0 | 82.9 |
| RCL | LLaVA-1.5-7B | 64.06 | 4.22 | 0.106 | 14.8 | 12.9 | 89.0 |
| RCL | LLaVA-1.5-13B | **65.28** | **3.76** | **0.093** | **13.0** | **11.1** | **90.6** |
| RCL | InternVL-Chat-7B | <u>64.75</u> | <u>3.93</u> | <u>0.098</u> | <u>13.7</u> | <u>11.8</u> | <u>89.9</u> |

channel still changes on $21.6\%$ of examples answered correctly by both checkpoints. RCL raises the average score to $64.06$ and lowers the flip rate to $14.8\%$. MRD falls from $0.159$ to $0.106$, HFR drops from $24.0\%$ to $12.9\%$, and the fraction of grounded reliance mass retained rises from $82.9\%$ to $89.0\%$. All three RCL configurations improve each reliance metric relative to the listed baselines. These results show how evidence-use metrics complement answer accuracy when evaluating continual adaptation.

### 5.3 Ablation Study and Analysis

**Single-factor ablation.** We ablate each major component of RCL on LLaVA-1.5-7B using the same task order, PEFT configuration, training budget, and evaluation protocol as the full model. As shown in Table 3, removing the reliance loss $\mathcal{L}_{\mathrm{rel}}$ leads to the largest degradation in reliance-aware diagnostics, especially MRD, DEF, and HFR. This indicates that stable outputs and answer accuracy can coexist with changes in evidence use during continual adaptation. Prediction distillation mainly improves conventional retention metrics, while the confidence gate helps filter unreliable teacher reliance targets and evidence-targeted interventions provide more faithful channel-level sensitivity signals. The weaker results for random interventions, single-score reliance variants, and visual-only reliance indicate that RCL's gains depend on preserving instance-specific, multichannel evidence use across stages. Generic perturbation regularization yields smaller gains.

**Answer retention and evidence-use drift.** We analyze all old-to-later transitions in the evidence-sensitive stream and compare clean answer drop, modality reliance drift (MRD), and dominant evidence flip rate (DEF) on answer-preserved examples. In Figure 3, ANSWER-KD shifts many points leftward, reducing the average answer drop from $3.41\%$ to $1.82\%$. Of the low-drop slices, $64.7\%$ fall in the hidden-forgetting region, with an average DEF of $35.9\%$. RCL reduces MRD from $0.238$ to $0.108$, DEF from $35.9\%$ to $14.6\%$, and hidden forgetting from $64.7\%$ to $10.8\%$. The largest reductions occur on OCR-, chart-, and document-centric slices, indicating that reliance preservation is most valuable for answers grounded in structured evidence.

**Training cost and hidden forgetting.** We evaluate how much hidden forgetting decreases with the additional forward passes required for counterfactual reliance estimation. We compare sequential tuning, output distillation, random intervention regularization, sparse RCL, and RCL variants with different intervention budgets using relative GPU cost and HFR reduction. In Figure 4, output

Table 3: **Single-factor ablation on LLaVA-1.5-7B.** We report standard benchmark performance and evidence-use diagnostics. Blue numbers indicate degradation relative to the full RCL model. The best results are bolded and the second-best results are underlined.

| Variant | Standard continual benchmarks | | | Evidence-sensitive stream | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | CoIN ACC↑ | COAST Avg.↑ | MCIT MFN↑ | Avg.↑ | AF↓ | MRD↓ | DEF↓ | HFR↓ | GCR↑ |
| RCL full | **65.01** | **52.55** | **58.86** | **64.06** | **4.22** | **0.106** | **14.8** | **12.9** | **89.0** |
| w/o reliance loss $\mathcal{L}_{\text{rel}}$ | 62.44 (-2.57) | 50.78 (-1.77) | 56.91 (-1.95) | 61.72 (-2.34) | 5.96 (+1.74) | 0.184 (+0.078) | 25.9 (+11.1) | 30.4 (+17.5) | 81.7 (-7.3) |
| w/o prediction distillation $\mathcal{L}_{\text{pred}}$ | 62.98 (-2.03) | 51.10 (-1.45) | 57.29 (-1.57) | 62.24 (-1.82) | 5.31 (+1.09) | 0.125 (+0.019) | 17.9 (+3.1) | 17.2 (+4.3) | 86.8 (-2.2) |
| w/o confidence gate $w_{\theta^-}$ | 63.28 (-1.73) | 51.46 (-1.09) | 57.54 (-1.32) | 62.83 (-1.23) | 5.08 (+0.86) | 0.137 (+0.031) | 19.4 (+4.6) | 20.1 (+7.2) | 85.6 (-3.4) |
| w/o entropy factor in gate | 63.81 (-1.20) | 51.88 (-0.67) | 57.92 (-0.94) | 63.21 (-0.85) | 4.72 (+0.50) | 0.126 (+0.020) | 17.1 (+2.3) | 17.8 (+4.9) | 86.9 (-2.1) |
| w/o sensitivity factor in gate | 63.66 (-1.35) | 51.75 (-0.80) | 57.83 (-1.03) | 63.08 (-0.98) | 4.81 (+0.59) | 0.132 (+0.026) | 18.6 (+3.8) | 18.7 (+5.8) | 86.2 (-2.8) |
| random interventions | 62.90 (-2.11) | 50.96 (-1.59) | 57.08 (-1.78) | 62.34 (-1.72) | 5.42 (+1.20) | 0.153 (+0.047) | 22.0 (+7.2) | 24.8 (+11.9) | 83.9 (-5.1) |
| loss-increase only | 63.95 (-1.06) | 52.05 (-0.50) | 58.06 (-0.80) | 63.41 (-0.65) | 4.63 (+0.41) | 0.124 (+0.018) | 17.0 (+2.2) | 16.3 (+3.4) | 86.9 (-2.1) |
| KL-shift only | 64.12 (-0.89) | 52.12 (-0.43) | 58.11 (-0.75) | 63.59 (-0.47) | 4.54 (+0.32) | 0.119 (+0.013) | 16.4 (+1.6) | 15.7 (+2.8) | 87.4 (-1.6) |
| visual-only reliance | 63.42 (-1.59) | 51.64 (-0.91) | 57.68 (-1.18) | 62.87 (-1.19) | 4.92 (+0.70) | 0.141 (+0.035) | 20.8 (+6.0) | 21.6 (+8.7) | 84.7 (-4.3) |
| w/o adapter regularization | 64.31 (-0.70) | 52.19 (-0.36) | 58.27 (-0.59) | 63.55 (-0.51) | 4.53 (+0.31) | 0.118 (+0.012) | 16.2 (+1.4) | 15.1 (+2.2) | 87.8 (-1.2) |

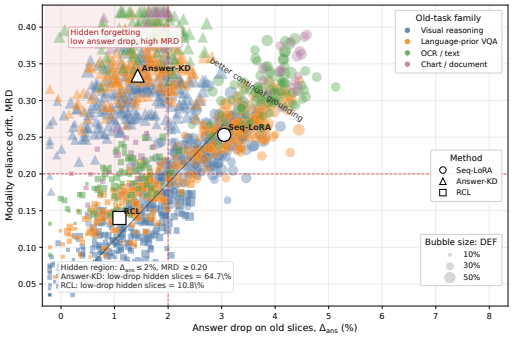


Figure 3: **Hidden evidence-use drift.** Position shows answer drop and MRD, and bubble size shows DEF. The shaded region marks slices with small accuracy drops and large reliance drift. See Appendix B.5 for the protocol.

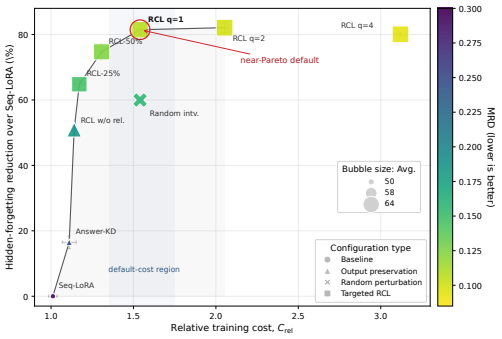


Figure 4: **Training cost and HFR.** Position shows relative cost and HFR reduction over Seq-LoRA. Size shows final average accuracy, and color shows MRD. Default RCL lies near the frontier. See Appendix B.5 for the protocol.

distillation adds little cost and reduces HFR by just $13.9\%$, confirming that output preservation alone is insufficient. Random interventions incur a similar training cost to RCL, with a $59.7\%$ HFR reduction and higher MRD. This pattern indicates that the gains depend on targeted evidence suppression. The default setting $q_{\text{train}} = 1$ reduces HFR by $79.0\%$ at $1.56\times$ the training cost. Increasing the budget to $q_{\text{train}} = 2$ or $q_{\text{train}} = 4$ yields marginal gains at substantially higher cost. Thus, one targeted intervention per available evidence channel already provides a strong reliance signal with no inference-time overhead.

## 6 Conclusion

Continual multimodal adaptation can retain correct answers while changing the evidence that supports them, a failure mode we call hidden evidence-use forgetting. We propose RCL, a replay-free framework that preserves evidence-reliance profiles through counterfactual channel interventions without adding inference-time cost. In our experiments, substantial reliance drift persists even when answers are retained. RCL improves both performance retention and grounding stability. These results suggest evaluating continual MLLMs on both answer accuracy and multimodal evidence use.


## Acknowledgments and Disclosure of Funding

The authors received no third-party funding or support for this work. The authors declare no competing interests.

## References


Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4971–4980, 2018.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: A strong, simple baseline. In *Advances in Neural Information Processing Systems*, volume 33, pages 15920–15930, 2020.

Meng Cao, Yuyang Liu, Yingfei Liu, Tiancai Wang, Jiahua Dong, Henghui Ding, Xiangyu Zhang, Ian Reid, and Xiaodan Liang. Continual llava: Continual instruction tuning in large vision-language models. *arXiv preprint arXiv:2411.02564*, 2024.

Cheng Chen, Junchen Zhu, Xu Luo, Heng T Shen, Jingkuan Song, and Lianli Gao. Coin: A benchmark of continual instruction tuning for multimodel large language models. *Advances in neural information processing systems*, 37:57817–57840, 2024a.

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu, Horace Ip, and Sam Kwong. SEFE: Superficial and essential forgetting eliminator for multimodal continual instruction tuning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 7982–8001. PMLR, 13–19 Jul 2025. URL `https://proceedings.mlr.press/v267/chen25n.html`.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Zhong Muyan, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 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*, pages 24185–24198, 2024b.

Kairui Ding, Boyuan Chen, Yuchen Su, Huan-ang Gao, Bu Jin, Chonghao Sima, Xiaohui Li, Wuqiang Zhang, Paul Barsch, Hongyang Li, and Hao Zhao. Hint-AD: Holistically aligned interpretability in end-to-end autonomous driving. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 3742–3765. PMLR, 06–09 Nov 2025. URL `https://proceedings.mlr.press/v270/ding25a.html`.

Enguang Fan, Anfeng Peng, Matthew Caesar, Jae Kim, Josh Eckhardt, Greg Kimberly, and Denis Osipychev. Towards effective swarm-based GPS spoofing detection in disadvantaged platforms. In *Proceedings of the 2023 IEEE Military Communications Conference (MILCOM)*, pages 722–728. IEEE, 2023. doi: 10.1109/MILCOM58377.2023.10356314.

Enguang Fan, Yifan Chen, Zihan Shan, Klara Nahrstedt, Matthew Caesar, and Jae Kim. Communication-aware multi-agent reinforcement learning for decentralized cooperative UAV deployment. arXiv preprint arXiv:2603.16141, 2026. URL `https://arxiv.org/abs/2603.16141`.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017.

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)*, pages 13572–13586, 2025a.

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. *arXiv preprint arXiv:2508.07307*, 2025b.

Jiani He, Dingyan Shang, Yihua Xu, Shiqi Huang, Yan Lyu, Jize Li, and Shangjing Tang. Semantic signal-assisted inspection and recovery allocation in reverse logistics. arXiv preprint arXiv:2609.02116, 2026. URL https://arxiv.org/abs/2609.02116.

Saisai Hu. Research on security enhancement methods for adversarial robust large language model intelligent agents for medical decision-making tasks. In *2026 7th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*, pages 599–603. IEEE, 2026.

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 computer vision and pattern recognition conference*, pages 19608–19617, 2025.

Shiqi Huang, Jiani He, Dingyan Shang, Yihua Xu, Jize Li, Yan Lyu, and Lakshmi Muraleedharan Nair. DACRI: Decision-aware causal intervention ranking for critical supply chains. arXiv preprint arXiv:2608.11154, 2026. URL https://arxiv.org/abs/2608.11154.

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*, pages 6700–6709, 2019.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

Jize Li, Jiani He, Dishu Yang, Dingyan Shang, Jingjing Liu, and Shiqi Huang. Accuracy-preserving stability regularization for large-scale retail demand forecasting. arXiv preprint arXiv:2607.13331, 2026. URL https://arxiv.org/abs/2607.13331.

Yang Li, Xiaoxue Chen, Hao Zhao, Jiangtao Gong, Guyue Zhou, Federico Rossano, and Yixin Zhu. Understanding embodied reference with Touch-Line Transformer. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ugA1HX69sf.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *Computer Vision – ECCV 2016*, volume 9908 of *Lecture Notes in Computer Science*, pages 614–629. Springer, 2016. doi: 10.1007/978-3-319-46493-0_37.

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*, pages 26296–26306, 2024.

Jingjing Liu, Dishu Yang, Hongbo Qin, and Hansong Zhang. Passing tests, missing semantics: Evidence of interpretive lock-in in plan-based DataFrame code generation. arXiv preprint arXiv:2610.15797, 2026. URL https://arxiv.org/abs/2610.15797.

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

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, volume 35, pages 2507–2521, 2022.

Yao Lu, Tengfei Ma, Zeyu Wang, Zhuangzhi Chen, Dongwei Xu, Yun Lin, Qi Xuan, and Guan Gui. FCOS: A two-stage recoverable model pruning framework for automatic modulation recognition. *IEEE Transactions on Cognitive Communications and Networking*, 12:10764–10778, 2026. doi: 10.1109/TCCN.2026.3706560.

Mao-Lin Luo, Zi-Hao Zhou, Tong Wei, and Min-Ling Zhang. Lada: Scalable label-specific clip adapter for continual learning. In *International Conference on Machine Learning*, pages 41604–41619. PMLR, 2025.

Mingjun Ma, Saisai Hu, Tiantian Zhu, Yitian Lin, Jiachen Sun, Zhengqiu Weng, Dehao Li, and Yajie Zhang. MuSK: Multi-scale knowledge learning for provenance-graph anomaly detection. *Computer Networks*, 289:112728, 2026. doi: 10.1016/j.comnet.2026.112728.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3195–3204, 2019.

Imad Eddine Marouf, Enzo Tartaglione, Stéphane Lathuilière, and Joost Van De Weijer. Ask and remember: A questions-only replay strategy for continual visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18078–18089, 2025.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.findings-acl.177.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2200–2209, 2021.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 947–952, 2019. doi: 10.1109/ICDAR.2019.00156.

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*, pages 26129–26149. PMLR, 2023.

Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12700–12710, 2021.

Wanru Shao. Interpretable ensemble learning for network traffic anomaly detection: A SHAP-based explainable AI framework for embedded systems security. In *2026 14th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–4. IEEE, 2026a. doi: 10.1109/ISDFS69419.2026.11459069. URL `https://doi.org/10.1109/ISDFS69419.2026.11459069`.

Wanru Shao. Policy-driven vulnerability risk quantification framework for large-scale cloud infrastructure data security. In *Proceedings of the 2026 5th International Conference on Cyber Security, Artificial Intelligence and Digital Economy*, pages 40–45, New York, NY, USA, 2026b. Association for Computing Machinery. doi: 10.1145/3819836.3819842. URL `https://doi.org/10.1145/3819836.3819842`.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11909–11919, 2023.

Tejas Srinivasan, Ting-Yun Chang, Leticia Pinto Alva, Georgios Chochlakis, Mohammad Rostami, and Jesse Thomason. Climb: A continual learning benchmark for vision-and-language tasks. *Advances in Neural Information Processing Systems*, 35:29440–29453, 2022.

Yuan Tang, ChiaHua Chang, Zhe Zhao, Zhizhou Gu, and Zhenglin Li. Intelligent detection and segmentation of gastrointestinal polyps based on YOLOv11 deep learning. In *Proceedings of the 2026 9th International Conference on Advanced Algorithms and Control Engineering (ICAACE)*, pages 825–828. IEEE, 2026. doi: 10.1109/ICAACE69793.2026.11508706.

Beiwen Tian, Mingdao Liu, Huan-ang Gao, Pengfei Li, Hao Zhao, and Guyue Zhou. Unsupervised road anomaly detection with language anchors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7778–7785, 2023. doi: 10.1109/ICRA48891.2023.10160470.

Vishaal Udandarao, Karsten Roth, Sebastian Dziadzio, Ameya Prabhu, Mehdi Cherti, Oriol Vinyals, Olivier Hénaff, Samuel Albanie, Zeynep Akata, and Matthias Bethge. A practitioner's guide to real-world continual multimodal pretraining. *Advances in Neural Information Processing Systems*, 37:133801–133845, 2024.

Mengxiao Wang and Nitesh Saxena. Trading agents or market crashers? dissecting robustness and security failures in academic financial LLM trading schemes. arXiv preprint arXiv:2609.19705, 2026. URL `https://arxiv.org/abs/2609.19705`.

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*, pages 10658–10671. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.715.

Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. DualPrompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*, pages 631–648. Springer, 2022a.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022b.

Zijun Wang, Saisai Hu, Tiantian Zhu, Wenrui Cheng, Zhengqiu Weng, Haiting Chen, Xiangyang Zheng, and Suyu Zhang. Trident: Dual-stream APT attribution over heterogeneous threat knowledge graphs. *Computers & Security*, 171:105094, 2026. doi: 10.1016/j.cose.2026.105094.

Zixuan Wang, Yu Sun, Hongwei Wang, Baoyu Jing, Xiang Shen, Xin Dong, Zhuolin Hao, Hongyu Xiong, and Yang Song. Reasoning-enhanced domain-adaptive pretraining of multimodal large language models for short video content governance. In Saloni Potdar, Lina Rojas-Barahona, and Sebastien Montella, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1104–1112, Suzhou (China), November 2025. Association for Computational Linguistics. ISBN 979-8-89176-333-3. doi: 10.18653/v1/2025.emnlp-industry.77. URL `https://aclanthology.org/2025.emnlp-industry.77/`.

Bin Wu, Wuxuan Shi, Jinqiao Wang, and Mang Ye. Synthetic data is an elegant gift for continual vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2813–2823, 2025.

Canran Xiao, Tianxiang Xu, Siyuan Ma, Yiyang Jiang, Haoyu Gao, and Yuhan Wu. Reversible primitive–composition alignment for continual vision–language learning. In *International Conference on Learning Representations*, 2026a.

Linqi Xiao, Venkateswarlu Gudepu, and Andrea Fumagalli. Joint alarm correlation and root alarm detection in optical transport networks with graph-attention. In *2025 IEEE 30th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pages 1–6. IEEE, 2025. doi: 10.1109/CAMAD67323.2025.11229899.

Linqi Xiao, Venkateswarlu Gudepu, and Andrea Fumagalli. CORA: A conversational AI-agent for intelligent automation of OpenROADM multi-vendor optical transport networks. In *Applications of Machine Learning 2026*, volume 14218 of *Proceedings of SPIE*, page 142180Y. SPIE, 2026b. doi: 10.1117/12.3113861.

Linqi Xiao, Aparaajitha Gomathinayakam Latha, Venkateswarlu Gudepu, and Andrea Fumagalli. A YANG-grounded LLM agent supporting multi-vendor OpenROADM optical transport networks. In *Proceedings of the ACM SIGCOMM 2026 Conference*, pages 2109–2115. ACM, 2026c. doi: 10.1145/3789240.3822606. HotOptics workshop.

Mengzhu Xu, Hanzhi Liu, Ningkang Peng, Qianyu Chen, and Canran Xiao. Affordance-first decomposition for continual learning in video–language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3908–3919, 2026.

Dishu Yang, Jingjing Liu, and Jize Li. Balancing privacy, utility, and safety in LLM alignment through preference optimization. arXiv preprint arXiv:2608.30141, 2026a. URL `https://arxiv.org/abs/2608.30141`.

Dishu Yang, Qi Su, Hongbo Qin, and Hansong Zhang. Measurement-first auditing of agentic leaderboards: Contamination susceptibility, matched-control re-evaluation, and scorer validation. arXiv preprint arXiv:2610.05830, 2026b. URL `https://arxiv.org/abs/2610.05830`.

Junyi Yao, Baichuan Li, Zihao Zheng, and Jiayu Long. Toward workflow-aware benchmarking for healthcare NLP agents. arXiv preprint arXiv:2609.00296, 2026. URL `https://arxiv.org/abs/2609.00296`.

Yahan Yu, Duzhen Zhang, Yong Ren, Xuanle Zhao, Xiuyi Chen, and Chenhui Chu. Progressive lora for multimodal continual instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2779–2796, 2025.

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*, pages 12137–12152, 2025.

Duzhen Zhang, Yong Ren, Zhong-Zhi Li, Yahan Yu, Jiahua Dong, Chenxing Li, Zhilong Ji, and Jinfeng Bai. Enhancing multimodal continual instruction tuning with branchlora. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5743–5756, 2025.

Fan Zhang, Gongguan Chen, Hua Wang, Jinjiang Li, and Caiming Zhang. Multi-scale video super-resolution transformer with polynomial approximation. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4496–4506, 2023. doi: 10.1109/TCSVT.2023.3278131.

Fan Zhang, Gongguan Chen, Hua Wang, and Caiming Zhang. CF-DAN: Facial-expression recognition based on cross-fusion dual-attention network. *Computational Visual Media*, 10(3):593–608, 2024. doi: 10.1007/s41095-023-0369-x.

Fan Zhang, Shiming Fan, and Hua Wang. Rethinking activation function: A simple path to efficient and accurate time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 38 (10):6987–7004, 2026a. doi: 10.1109/TKDE.2026.3716656.

Fan Zhang, Shiming Fan, and Hua Wang. TimeSAF: Towards LLM-guided semantic asynchronous fusion for time series forecasting. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26267–26284, San Diego, California, United States, 2026b. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.1208. URL `https://aclanthology.org/2026.acl-long.1208/`.

Fan Zhang, Zhiwei Gu, and Hua Wang. Decoding with structured awareness: Integrating directional, frequency-spatial, and structural attention for medical image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 12421–12429, 2026c. doi: 10.1609/aaai.v40i15.38235.

Jiayu Zhang, Chuangxin Zhao, Canran Xiao, Ruibo Duan, Wenyi Mo, Haoyu Gao, and Wenshuo Wang. Pi-CCA: Prompt-invariant CCA certificates for replay-free continual multimodal learning. In *International Conference on Learning Representations*, 2026d.

Hongbo Zhao, Fei Zhu, Haiyang Guo, Meng Wang, Rundong Wang, Gaofeng Meng, and Zhaoxiang Zhang. Mllm-cl: Continual learning for multimodal large language models. *arXiv preprint arXiv:2506.05453*, 2025.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19125–19136, 2023.

Zihao Zheng, Baichuan Li, Junyi Yao, and Jiayu Long. Reliable financial named entity recognition under domain shift: Confidence estimation and selective prediction. arXiv preprint arXiv:2608.19558, 2026. URL `https://arxiv.org/abs/2608.19558`.

Didi Zhu, Zhongyi Sun, Zexi Li, Tao Shen, Ke Yan, Shouhong Ding, Chao Wu, and Kun Kuang. Model tailor: mitigating catastrophic forgetting in multi-modal large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 62581–62598, 2024.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22257–22267, 2023.

# A Additional Method Details

## A.1 Counterfactual Intervention Construction

RCL estimates reliance through evidence-suppressing interventions. Each intervention targets one evidence channel while keeping the remaining channels unchanged. We instantiate the intervention set $\mathcal{I}_m$ according to the channel type.

**Visual channel.** For image-based inputs, $\mathcal{I}_{\mathrm{img}}$ contains region masking and visual-token removal. Region masking replaces selected image patches with a neutral value. Visual-token removal removes the corresponding visual embeddings before multimodal fusion. For datasets with object annotations, regions are sampled from annotated objects. For datasets without object annotations, regions are sampled from uniformly partitioned image patches.

**Question-text channel.** For question or instruction text, $\mathcal{I}_{\mathrm{txt}}$ contains content-word masking. Stop words and formatting tokens are preserved. This intervention suppresses semantic cues from the question while keeping the prompt format stable.

**OCR channel.** For OCR-based inputs, $\mathcal{I}_{\mathrm{ocr}}$ contains OCR-token removal and OCR-token masking. OCR-token removal deletes recognized text tokens from the input sequence. OCR-token masking replaces recognized text tokens with a neutral placeholder token.

**Chart and table channels.** For chart and table reasoning, $\mathcal{I}_{\mathrm{chart}}$ contains axis-label masking, legend masking, cell-value masking, and plotted-element masking. These interventions suppress numerical and structural evidence while preserving the global input format.

**Video channel.** For video inputs, $\mathcal{I}_{\mathrm{vid}}$ contains frame masking and sparse frame removal. Frame masking suppresses visual content in selected frames. Sparse frame removal drops selected frame embeddings before multimodal fusion.

For each minibatch, RCL samples a fixed number of interventions from $\mathcal{I}_m$ for every available channel $m \in \mathcal{M}_x$. The same intervened inputs are shared by the frozen reference model and the current model, ensuring that the reliance loss compares teacher and student behavior under identical counterfactual conditions.

## A.2 Confidence Gate for Reliance Targets

The reliance loss in Eq. (7) uses a confidence weight $w_{\theta^-}(x, y)$ to reduce the influence of unreliable reference targets. The gate combines predictive confidence and intervention sensitivity.

The reference predictive entropy is

$$H_{\theta^-}(x, y) = \frac{1}{L} \sum_{\ell=1}^{L} H(p_{\theta^-}(\cdot \mid y_{<\ell}, x)) \,, \tag{10}$$

where $H(\cdot)$ denotes Shannon entropy. The total reference intervention sensitivity is

$$A_{\theta^-}(x, y) = \sum_{m \in \mathcal{M}_x} \Delta_{\theta^-, m}(x, y). \tag{11}$$

The confidence weight is defined as

$$w_{\theta^-}(x, y) = \sigma\left(\frac{\tau_H - H_{\theta^-}(x, y)}{\gamma_H}\right) \cdot \sigma\left(\frac{A_{\theta^-}(x, y) - \tau_A}{\gamma_A}\right), \tag{12}$$

where $\sigma(\cdot)$ is the sigmoid function, $\tau_H$ and $\tau_A$ are fixed thresholds, and $\gamma_H, \gamma_A > 0$ are temperature parameters. The first factor assigns larger weight to low-entropy reference predictions. The second factor assigns larger weight to samples where the reference model exhibits measurable dependence on at least one evidence channel. The reliability of confidence signals can depend on the evaluation domain, as observed in financial named entity recognition under domain shift [Zheng et al., 2026].

### A.3 Diagnostic Attribution Analysis

We use gradient-based attribution only at evaluation to assess agreement between counterfactual reliance and local sensitivity in the model representation.

Let $h_\theta^m(x)$ denote the hidden representation associated with channel $m$. The attribution score for channel $m$ is

$$G_{\theta,m}(x,y) = \frac{\left\| h_\theta^m(x) \odot \nabla_{h_\theta^m(x)} \ell_\theta(y \mid x) \right\|_1}{\dim(h_\theta^m(x))}. \tag{13}$$

Since $G_{\theta,m}$ is never differentiated through during training, RCL avoids second-order optimization. We report attribution agreement by comparing the normalized attribution vector $\boldsymbol{g}_\theta(x,y)$ with the counterfactual reliance vector $\boldsymbol{r}_\theta(x,y)$ using rank correlation and Jensen–Shannon divergence.

### A.4 Task-Preserving Perturbation Analysis

We reserve task-preserving perturbations for robustness analysis and exclude them from the main reliance loss. They test sensitivity to changes that leave the correct answer unchanged.

For each channel $m$, let $\mathcal{P}_m$ be a set of task-preserving perturbations. Examples include mild image noise, small resolution changes, harmless crop-and-resize transformations, question paraphrases that preserve meaning, and OCR formatting changes that preserve recognized text. The invariance sensitivity is

$$B_{\theta,m}(x,y) = \frac{1}{|\mathcal{P}_m|} \sum_{\pi_m \in \mathcal{P}_m} \frac{1}{L} \sum_{\ell=1}^{L} D_{\mathrm{KL}} \left( p_\theta(\cdot \mid y_{<\ell}, x) \,\|\, p_\theta(\cdot \mid y_{<\ell}, \pi_m(x)) \right). \tag{14}$$

A grounded and stable model should have high sensitivity to evidence-suppressing interventions and low sensitivity to task-preserving perturbations.

### A.5 Efficient Training

The computational overhead of RCL comes from counterfactual forward passes through the frozen reference model and the current model. We control this overhead with three design choices.

First, for each minibatch, RCL samples a small fixed number of interventions per channel. Second, the same counterfactual inputs are used for both teacher and student models. Third, reliance estimation is applied only to the channels present in the current sample. Therefore, if a sample contains $|\mathcal{M}_x|$ channels and $q$ interventions are sampled per channel, the reliance module uses $q|\mathcal{M}_x|$ counterfactual views for that sample.

The teacher model is evaluated under stop-gradient. The student model receives gradients only through the reliance vector $\boldsymbol{r}_\theta(x,y)$ computed from counterfactual output shifts. No gradient-based attribution term is used in the training objective.

### A.6 Training Procedure

At continual stage $k$, RCL performs the following procedure.

1. Freeze the previous model $f_{\theta^-} = f_{\theta_{k-1}}$.
2. Initialize the current trainable model with $\theta = \theta^-$ and update only PEFT parameters $\phi$.
3. Sample a minibatch $\mathcal{B}_k$ from the current-stage dataset $\mathcal{D}_k$.
4. For each $(x,y) \in \mathcal{B}_k$ and each channel $m \in \mathcal{M}_x$, construct counterfactual inputs $\tau_m(x)$.
5. Compute reference reliance $\boldsymbol{r}_{\theta^-}(x,y)$ and current reliance $\boldsymbol{r}_\theta(x,y)$ using Eq. (3) and Eq. (4).
6. Compute $\mathcal{L}_{\mathrm{task}}$, $\mathcal{L}_{\mathrm{pred}}$, and $\mathcal{L}_{\mathrm{rel}}$ using Eq. (2), Eq. (8), and Eq. (7).
7. Update $\phi$ by minimizing Eq. (9).
8. After stage $k$ finishes, set $\theta_k = \theta$ and use $f_{\theta_k}$ as the frozen reference for stage $k+1$.

### A.7 Evaluation Metrics

We evaluate continual multimodal learning from two perspectives: answer retention and reliance preservation.

**Final average accuracy.** After the final stage $K$, the final average accuracy is

$$\mathrm{AA}_K = \frac{1}{K}\sum_{j=1}^{K}\mathrm{Acc}(f_{\theta_K};\mathcal{E}_j), \tag{15}$$

where $\mathcal{E}_j$ is the evaluation set for stage $j$.

**Average forgetting.** The standard average forgetting score is

$$\mathrm{AF}_K = \frac{1}{K-1}\sum_{j=1}^{K-1}\left[\max_{s\in\{j,\dots,K\}}\mathrm{Acc}(f_{\theta_s};\mathcal{E}_j) - \mathrm{Acc}(f_{\theta_K};\mathcal{E}_j)\right]. \tag{16}$$

**Reliance drift.** The final reliance drift is

$$\mathrm{MRD}_K = \frac{1}{K-1}\sum_{j=1}^{K-1}\frac{1}{|\mathcal{E}_j|}\sum_{(x,y)\in\mathcal{E}_j} D_{\mathrm{JS}}\left(\boldsymbol{r}_{\theta_j}(x,y)\,\|\,\boldsymbol{r}_{\theta_K}(x,y)\right). \tag{17}$$

**Channel-specific drift.** For channel $m$, the channel-specific drift from stage $j$ to the final model is

$$\Delta^{m}_{j\to K} = \frac{1}{|\mathcal{E}_j^m|}\sum_{(x,y)\in\mathcal{E}_j^m}\left[r_{\theta_K,m}(x,y) - r_{\theta_j,m}(x,y)\right], \tag{18}$$

where $\mathcal{E}_j^m = \{(x,y)\in\mathcal{E}_j : m\in\mathcal{M}_x\}$. A negative value for the visual channel indicates a drift away from visual evidence.

**Hidden forgetting rate.** For a single seed, the final hidden forgetting rate is

$$\boxed{\mathrm{HFR}_K = \frac{1}{K-1}\sum_{j=1}^{K-1}\frac{1}{|\mathcal{S}_j|}\sum_{s\in\mathcal{S}_j}\mathbf{1}\left[\Delta_{\mathrm{acc}}^{j\to K,s}\le 2\%\right]\mathbf{1}\left[\mathrm{MRD}_{j\to K,s}\ge 0.20\right]}. \tag{19}$$

Here $\mathcal{S}_j$ indexes the diagnostic slices of old task $j$, and the accuracy drop and reliance drift are evaluated on each slice as in Section 4.3. Each slice has equal weight within its task, and each old task has equal weight in the stream average.

## B Additional Experimental Details

### B.1 Details of the Preliminary Study

This section details the protocol used in Section 3. The preliminary study diagnoses evidence-use drift during standard replay-free multimodal adaptation.

**Continual stream.** We construct an eight-stage multimodal learning stream: ScienceQA [Lu et al., 2022], GQA [Hudson and Manning, 2019], VQAv2 [Goyal et al., 2017], OK-VQA [Marino et al., 2019], TextVQA [Singh et al., 2019], OCR-VQA [Mishra et al., 2019], ChartQA [Masry et al., 2022], and DocVQA [Mathew et al., 2021]. These stages cover visual reasoning, compositional VQA, knowledge-intensive VQA, OCR-based question answering, chart understanding, and document understanding. After learning each stage $j$, we save the resulting checkpoint $f_j$. After learning a later stage $k > j$, we evaluate the current checkpoint $f_k$ on the previous-stage evaluation set $\mathcal{E}_j$. No previous training examples are replayed during adaptation.

**Baselines.** We compare two baselines. The first baseline, **Seq-LoRA**, updates only parameter-efficient adaptation parameters on the current-stage task loss:

$$\mathcal{L}_{\mathrm{Seq}} = \mathbb{E}_{(x,y)\sim\mathcal{D}_k}\left[-\log p_\theta(y \mid x)\right]. \tag{20}$$

The second baseline, **Answer-KD**, additionally preserves the previous checkpoint's output distribution on the current-stage data:

$$\mathcal{L}_{\mathrm{KD}} = \mathcal{L}_{\mathrm{Seq}} + \lambda_{\mathrm{kd}}\mathbb{E}_{(x,y)\sim\mathcal{D}_k}\left[D_{\mathrm{KL}}\left(p^{\tau}_{\theta^-}(\cdot \mid x) \,\|\, p^{\tau}_{\theta}(\cdot \mid x)\right)\right], \tag{21}$$

where $\theta^-$ denotes the previous checkpoint and $\tau$ is the distillation temperature. This baseline preserves output behavior and leaves the evidence supporting each prediction unconstrained.

**Post-hoc channel ablations.** For each example $(x, y)$, we define a set of available evidence channels $\mathcal{C}_x$. Depending on the task, this set may include visual regions, OCR tokens, document text, chart elements, and question-side language priors. For each channel $m \in \mathcal{C}_x$, we construct an ablated input $\tau_m(x)$ that suppresses channel $m$ while keeping other channels unchanged. For example, visual ablation masks image tokens or visual regions; OCR/text ablation removes recognized text or document tokens; chart ablation masks chart elements; and question-prior ablation removes or neutralizes evidence-bearing lexical cues in the question. We apply channel ablations only after training.

For a checkpoint $f$, we define the diagnostic channel sensitivity as

$$s_m(f; x, y) = \left[\ell_f(y \mid \tau_m(x)) - \ell_f(y \mid x)\right]_+, \tag{22}$$

where $\ell_f(y \mid x)$ is the negative log-likelihood of the ground-truth answer and $[z]_+ = \max(z, 0)$. The normalized evidence-use vector is

$$q_m(f; x, y) = \frac{s_m(f; x, y)}{\sum_{n\in\mathcal{C}_x} s_n(f; x, y) + \epsilon}, \qquad \boldsymbol{q}(f; x, y) = \left[q_m(f; x, y)\right]_{m\in\mathcal{C}_x}. \tag{23}$$

The small constant $\epsilon$ prevents numerical instability when all channel sensitivities are close to zero. We use $\boldsymbol{q}(f; x, y)$ as a post-hoc proxy for evidence use.

**Metrics.** For each old-to-new transition $(j, k)$ with $k > j$, we report three quantities. The clean accuracy drop $A_{j\to k}$ measures conventional answer forgetting. The evidence-sensitivity drift $G_{j\to k}$ measures how much the normalized evidence-use vector changes between checkpoints. The dominant-evidence flip rate $H_{j\to k}$ is computed only on examples for which both $f_j$ and $f_k$ predict the correct answer. $H_{j\to k}$ measures how often the dominant channel changes among these examples. In Figure 1, we mark a transition as falling into the hidden-drift region when

$$A_{j\to k} \le 2\%, \qquad G_{j\to k} \ge 0.20. \tag{24}$$

This pilot diagnostic uses the post-hoc evidence-sensitivity drift as its measure of reliance change.

**Evaluation slices and aggregation.** We partition each previous-stage evaluation set into diagnostic slices to capture variation within tasks. These include easy, medium, hard, rare, compositional, and format-sensitive subsets. Each point in Figure 1(a) corresponds to one tuple of old task, later task, random seed, evaluation slice, and training baseline. The x-axis is $A_{j\to k}$, the y-axis is $G_{j\to k}$, the marker size is $H_{j\to k}$, the marker shape denotes the baseline, and the color denotes the old-task family. Figure 1(b) focuses on Answer-KD. Each cell corresponds to evaluating an old task after a later stage is learned; color indicates the dominant-evidence flip rate, and bubble size indicates the clean accuracy drop.

**Training and diagnosis.** Both baselines use task supervision, with output distillation added for Answer-KD. Reliance preservation, intervention consistency, and evidence-channel matching are absent from the pilot training objectives. We apply channel ablations only to the resulting checkpoints. The observed evidence-use drift motivates the reliance-preservation objective in Section 4.

### B.2 Detailed Experimental Setup

#### B.2.1 Benchmark Suites

We evaluate RCL on four benchmark suites that together cover standard multimodal continual learning, recent multimodal continual instruction tuning, and the evidence-use drift targeted by our method.

Table 4: **Benchmark overview.** Task composition and evaluation goals for each benchmark suite.

| Suite | Continual setting | Representative tasks | Purpose |
|---|---|---|---|
| CoIN | Continual instruction tuning | VQA, OCR-QA, recognition, classification, referring expression comprehension | Main MCIT comparison |
| COAST | Domain-, capability-, and dataset-incremental adaptation | ChartQA, DocVQA, IconQA, medical QA, conversation, reasoning, REC, VQA | Cross-setting robustness |
| MCITlib | Leakage-aware MCIT protocol | UCIT, MLLM-DCL, MLLM-ACL | Reproducible benchmark comparison |
| Evidence-sensitive stream | Reliance-aware continual adaptation | ScienceQA, GQA, VQAv2, OK-VQA, TextVQA, OCR-VQA, ChartQA, DocVQA | Hidden evidence-use forgetting |

**CoIN.** We use CoIN [Chen et al., 2024a] as a primary continual instruction-tuning benchmark for MLLMs. It contains diverse multimodal tasks, including visual question answering, OCR reasoning, object recognition, image classification, and referring expression comprehension. Following the official protocol, we train the model sequentially over the task stream and evaluate all seen tasks after each stage. CoIN also provides a shared benchmark for comparison with recent MCIT methods.

**COAST.** We use the COAST benchmark from Continual LLaVA [Cao et al., 2024] to evaluate three complementary continual adaptation scenarios: domain-incremental learning, capability-incremental learning, and dataset-incremental learning. The domain-incremental setting evaluates robustness across visual domains such as natural images, documents, charts, medical-style images, and scientific visual content. The capability-incremental setting evaluates whether the model can sequentially acquire different multimodal abilities, including recognition, OCR, chart reasoning, visual conversation, and complex reasoning. The dataset-incremental setting evaluates whether the model can adapt to new datasets without catastrophically forgetting previously learned ones.

**MCITlib.** We further evaluate on MCITlib [Guo et al., 2025b], which provides leakage-aware and reproducible multimodal continual instruction tuning protocols. We use its UCIT, MLLM-DCL, and MLLM-ACL settings. UCIT evaluates unified continual instruction tuning across heterogeneous task families. MLLM-DCL emphasizes domain continual learning, where the task format may remain similar but the visual or semantic domain changes. MLLM-ACL emphasizes ability continual learning, where the model must sequentially acquire different reasoning and perception abilities.

**Evidence-sensitive stream.** To directly evaluate hidden evidence-use forgetting, we construct an eight-stage stream:

$$\begin{aligned}&\text{ScienceQA} \rightarrow \text{GQA} \rightarrow \text{VQAv2} \rightarrow \text{OK-VQA}\\ &\quad \rightarrow \text{TextVQA} \rightarrow \text{OCR-VQA} \rightarrow \text{ChartQA} \rightarrow \text{DocVQA}.\end{aligned}$$

The stages cover distinct sources of evidence. ScienceQA and GQA emphasize visual reasoning and compositional grounding; VQAv2 and OK-VQA include strong language-prior and external-knowledge effects; TextVQA and OCR-VQA require OCR grounding; ChartQA requires chart-element and numerical reasoning; DocVQA requires document text and layout understanding. This makes the stream suitable for testing whether a continual MLLM preserves the evidence path behind retained answers.

### B.2.2 Dataset Preprocessing and Answer Normalization

For all benchmarks, we follow the official train/validation/test splits whenever available. If a benchmark provides multiple task orders, we report the average over the official orders. For datasets without an official continual split, we use the original training set for the corresponding continual stage and reserve the official validation set for evaluation. To prevent dataset size from dominating the sequential dynamics, we cap each stage at the same maximum number of training examples unless the official benchmark protocol specifies otherwise.

For closed-form VQA tasks, we apply standard VQA answer normalization, including lowercasing, punctuation removal, article removal, and number normalization. For OCR- and document-centric tasks, we additionally normalize whitespace, line breaks, and common OCR artifacts. For chart and table QA, we normalize numerical answers by allowing minor formatting differences such as commas, percentage signs, and decimal-place variants. For open-ended generative tasks, we use the official evaluator of the corresponding benchmark. When GPT-assisted scoring is required by the benchmark, we use it only for evaluation and never for training or hyperparameter selection.

#### B.2.3 Task-Level Metrics

We report the official metric for each task family:

- **VQA-style tasks:** exact-match or soft VQA accuracy, depending on the dataset evaluator.
- **Classification and recognition:** top-1 accuracy.
- **Referring expression comprehension:** $\mathrm{Acc@0.5}$ based on box IoU.
- **OCR and document QA:** normalized exact match and ANLS when supported by the official evaluator.
- **Chart and table QA:** relaxed exact match for numerical answers and official ChartQA-style accuracy.
- **Open-ended instruction following:** official GPT-assisted or rule-based score, following the benchmark protocol.

When a benchmark reports a single aggregated score, we use the official aggregation. When a benchmark reports separate task-family scores, we report both the task-family score and the macro-average across task families.

#### B.2.4 Continual Learning Metrics

Let $a_{i,j}$ denote the score on evaluation set $\mathcal{E}_i$ after training stage $j$. For a $K$-stage stream, we report immediate task acquisition, final retained performance, average forgetting, backward transfer, and mean accuracy over the learning trajectory:

$$\begin{aligned}
\mathrm{MFT} &= \frac{1}{K}\sum_{i=1}^{K} a_{i,i}, \\
\mathrm{MFN} &= \frac{1}{K}\sum_{i=1}^{K} a_{i,K}, \\
\mathrm{AF} &= \frac{1}{K-1}\sum_{i=1}^{K-1}\left(\max_{j\in\{i,\ldots,K\}} a_{i,j} - a_{i,K}\right), \\
\mathrm{BWT} &= \frac{1}{K-1}\sum_{i=1}^{K-1}\left(a_{i,K} - a_{i,i}\right), \\
\mathrm{MAA} &= \frac{1}{K}\sum_{j=1}^{K}\frac{1}{j}\sum_{i=1}^{j} a_{i,j}.
\end{aligned} \tag{25}$$

Here MFT measures how well the model learns each new stage when it is first trained; MFN measures final retained performance; AF measures the amount of forgetting; BWT measures whether later training improves or harms previous tasks; and MAA measures average performance throughout the whole continual trajectory.

#### B.2.5 Reliance-Aware Metrics

Standard continual learning metrics track answer performance. Reliance-aware metrics track how evidence use changes across checkpoints.

For each example $(x, y)$ and checkpoint $\theta_j$, we compute a normalized reliance vector $\boldsymbol{r}_{\theta_j}(x, y)$ over available evidence channels. The evidence channels may include image tokens, question tokens,

OCR tokens, document text, chart elements, table cells, layout regions, and language-prior cues. The channel set is sample-dependent and is denoted by $\mathcal{M}_x$.

**Modality reliance drift.** The reliance drift from stage $i$ to the final checkpoint $K$ is

$$\mathrm{MRD}_{i\to K} = \frac{1}{|\mathcal{E}_i|} \sum_{(x,y)\in\mathcal{E}_i} D_{\mathrm{JS}}\left(\boldsymbol{r}_{\theta_i}(x,y) \| \boldsymbol{r}_{\theta_K}(x,y)\right). \tag{26}$$

We report the stream-level score

$$\mathrm{MRD}_K = \frac{1}{K-1} \sum_{i=1}^{K-1} \mathrm{MRD}_{i\to K}. \tag{27}$$

Lower MRD indicates that the final model preserves the evidence-use pattern of earlier checkpoints.

**Dominant evidence flip rate.** The dominant evidence flip rate isolates examples where the answer remains correct but the dominant evidence channel changes:

$$\begin{aligned}\mathrm{DEF}_{i\to K} = \mathbb{E}_{(x,y)\in\mathcal{E}_i}\Big[ & \\ & \mathbf{1}\Big[\arg\max_{m\in\mathcal{M}_x} r_{\theta_i,m}(x,y) \neq \arg\max_{m\in\mathcal{M}_x} r_{\theta_K,m}(x,y)\Big] \\ & \Big| \; f_{\theta_i}(x) = f_{\theta_K}(x) = y\Big].\end{aligned} \tag{28}$$

Higher DEF indicates more frequent changes in the dominant evidence source among correct answers.

**Hidden forgetting rate.** We apply Eq. (19) at the final checkpoint $K$. The thresholds of two percentage points for accuracy drop and $0.20$ for reliance drift are applied to each diagnostic slice before aggregation. Both thresholds are fixed before evaluation.

**Channel-specific drift.** For a channel $m$, we report

$$\Delta^m_{i\to K} = \frac{1}{|\mathcal{E}_i^m|} \sum_{(x,y)\in\mathcal{E}_i^m} \left[r_{\theta_K,m}(x,y) - r_{\theta_i,m}(x,y)\right], \tag{29}$$

where $\mathcal{E}_i^m = \{(x,y)\in\mathcal{E}_i : m\in\mathcal{M}_x\}$. Negative visual, OCR, chart, or document drift indicates that the final model relies less on the corresponding grounded channel than the earlier checkpoint.

**Grounded channel retention.** Grounded channel retention (GCR) measures the fraction of the original grounded reliance mass preserved at the final checkpoint. Let $f_{\theta_j}$ denote the model after learning task $j$ and $f_{\theta_K}$ the final model. For input $x$, the grounded channels are

$$\mathcal{G}_x = \mathcal{M}_x \cap \{\text{Image}, \text{OCR}, \text{Chart/Table}, \text{Document}\}.$$

The question and language-prior channels are excluded from $\mathcal{G}_x$. For each old task $j$, we evaluate examples with at least one grounded channel that both checkpoints answer correctly:

$$\Omega_j^G = \left\{(x,y)\in\mathcal{E}_j : f_{\theta_j}(x) = f_{\theta_K}(x) = y,\ \mathcal{G}_x \neq \varnothing\right\}. \tag{30}$$

The retained grounded reliance mass $N_j$ and the original grounded reliance mass $D_j$ are

$$\begin{aligned} N_j &= \sum_{(x,y)\in\Omega_j^G} \sum_{m\in\mathcal{G}_x} \min\big(r_{\theta_j,m}(x,y),\ r_{\theta_K,m}(x,y)\big), \\ D_j &= \sum_{(x,y)\in\Omega_j^G} \sum_{m\in\mathcal{G}_x} r_{\theta_j,m}(x,y). \end{aligned} \tag{31}$$

The channel-wise minimum caps the retained mass at the amount originally assigned to that channel. For each task with $D_j > 0$, we compute

$$\mathrm{GCR}_{j\to K} = 100\% \times \frac{N_j}{D_j}. \tag{32}$$

Since $0 \leq N_j \leq D_j$, GCR ranges from $0\%$ to $100\%$. Within each seed, we average over eligible old tasks with equal weight. When all old tasks are eligible, the stream-level score is

$$\mathrm{GCR}_K = \frac{1}{K-1} \sum_{j=1}^{K-1} \mathrm{GCR}_{j \to K}, \tag{33}$$

where $K = 8$ for the evidence-sensitive stream. We compute the stream-level GCR separately for each seed and report the arithmetic mean over three seeds.

### B.3 Comparison Protocol

#### B.3.1 Method Groups

We group compared methods according to the information and memory they require.

**Replay-free and task-id-free methods.** This is the primary comparison group. All methods in this group are trained sequentially without storing previous examples and are evaluated without oracle task identity. This group includes: Seq-LoRA, EWC [Kirkpatrick et al., 2017], LwF [Li and Hoiem, 2016], L2P [Wang et al., 2022b], DualPrompt [Wang et al., 2022a], CODA-Prompt [Smith et al., 2023], O-LoRA [Wang et al., 2023], MoELoRA, Model Tailor, Continual LLaVA, CL-MoE, HiDe-LLaVA, BranchLoRA, ProgLoRA, ModalPrompt, SEFE, MLLM-CL, and RCL.

**Replay- or memory-assisted methods.** Some methods store old examples, generate replay data, or preserve past-task questions. We report them separately because they use additional past-task signals unavailable to strictly replay-free methods. This group includes rehearsal, DER [Buzzega et al., 2020], GIFT [Wu et al., 2025], and Ask-and-Remember [Marouf et al., 2025]. For rehearsal and DER, we use a $1\%$ memory buffer unless otherwise specified by the official protocol. For synthetic replay methods, we follow the official generation budget and report the additional generated-sample count.

**Oracle or task-aware methods.** If a method requires oracle task identifiers, a task-specific expert selector, or separate task-specific heads at inference time, we report it in an oracle table. Oracle results serve as upper references and are reported separately from the main comparison. When the official implementation provides both oracle and non-oracle variants, we report both.

#### B.3.2 Matched Comparison Settings

Directly compared methods follow these settings:

- **Same task order.** All methods are trained on the same continual stream and evaluated after the same stages.
- **Same data budget.** Each method sees the same current-stage training examples. Replay-free methods do not access previous-stage examples.
- **Same backbone.** Direct comparisons are made under the same MLLM backbone.
- **Matched trainable parameters.** When a method allows rank, prompt length, or expert-count adjustment, we tune it so that the trainable parameter count is within $5\%$ of RCL.
- **Matched optimization budget.** Methods are trained with the same number of epochs, batch size, and total update steps whenever their official protocol permits.
- **Task-id-free inference.** Methods in the main comparison infer without oracle task IDs.
- **Separate memory accounting.** Methods using stored or generated replay are reported with their memory or generation budget.
- **Same evaluation interventions.** Reliance-aware diagnostics use the same post-hoc counterfactual intervention set for every method.

#### B.3.3 Baseline Implementation Notes

**Seq-LoRA.** Seq-LoRA updates LoRA adapters sequentially on each stage using only the current-stage task loss. Seq-LoRA shares RCL's backbone and adapter placement and trains on the task loss alone.

**Answer-KD / LwF.** Answer-KD preserves the previous checkpoint's output distribution on current-stage samples using token-level KL divergence. It tests whether output preservation alone is sufficient to prevent hidden evidence-use forgetting.

**Prompt-based continual learning.** L2P, DualPrompt, and CODA-Prompt are adapted to the multimodal instruction tuning setting by inserting learnable prompts into the language-side input or adapter-compatible prompt slots. When task identifiers are not available, prompt selection uses the official key-query matching mechanism.

**Adapter- and LoRA-based MCIT methods.** O-LoRA, BranchLoRA, ProgLoRA, Model Tailor, HiDe-LLaVA, ModalPrompt, CL-MoE, SEFE, and MLLM-CL are implemented following their official designs whenever code is available. If official code is unavailable, we reproduce the method from the paper using the same backbone and training schedule used for other baselines.

**Replay-based methods.** For rehearsal and DER, we maintain a fixed-size buffer selected uniformly from previous stages. For GIFT and Ask-and-Remember, we follow the original synthetic-data or question-only replay protocol and report the replay budget separately.

### B.4 Implementation Details

#### B.4.1 Backbones

We use three MLLM backbones:

- **LLaVA-1.5-7B** [Liu et al., 2024]: the default backbone for main experiments because it is widely used in recent MCIT work.
- **LLaVA-1.5-13B** [Liu et al., 2024]: used to evaluate whether reliance preservation remains effective at a larger model scale.
- **InternVL-Chat-7B** [Chen et al., 2024b]: used to test whether RCL generalizes beyond the LLaVA model family.

Unless otherwise specified, the visual encoder, language backbone, and multimodal projector are frozen. Only PEFT parameters are updated during continual adaptation.

#### B.4.2 PEFT Configuration

For RCL, we use LoRA adapters on attention projection layers. The default configuration is:

$$r = 16, \qquad \alpha = 32, \qquad p_{\text{dropout}} = 0.05.$$

For larger backbones, we keep the same LoRA rank unless otherwise specified. For baselines with tunable parameter budgets, such as prompt length, expert number, or adapter rank, we adjust their configuration to match the trainable parameter count of RCL within $5\%$.

#### B.4.3 Optimization

We train all methods using AdamW with bfloat16 mixed precision. The default optimization configuration is:

$$\text{global batch size} = 32, \quad \text{warm-up ratio} = 3\%, \quad \text{gradient clipping} = 1.0, \quad \text{weight decay} = 0.$$

For LLaVA-style backbones, the default learning rate is $2 \times 10^{-4}$ for LoRA parameters. For InternVL-style backbones, we use $1 \times 10^{-4}$ for stability. For COAST and MCITlib settings with official hyperparameters, we additionally report the official-setting result when it differs from the matched-budget setting. Each continual stage is trained for two epochs unless the official benchmark protocol requires a fixed update budget.

#### B.4.4 RCL Hyperparameters

The default RCL objective is

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{pred}}\mathcal{L}_{\text{pred}} + \lambda_{\text{rel}}\mathcal{L}_{\text{rel}} + \lambda_{\text{reg}}\|\phi - \phi^{-}\|_2^2.$$

We use the following default values:

$$\tau_r = 0.5, \quad \tau_{\mathrm{kd}} = 2.0, \quad \beta = 1.0, \quad \lambda_{\mathrm{pred}} = 0.5, \quad \lambda_{\mathrm{rel}} = 1.0, \quad \lambda_{\mathrm{reg}} = 10^{-4}.$$

We select these values using the validation split of the first stream only and keep them fixed for all later stages and all benchmark suites.

#### B.4.5 Counterfactual Intervention Budget

For each sample, RCL identifies the available evidence channels $\mathcal{M}_x$. During training, we sample one intervention per available channel:

$$q_{\mathrm{train}} = 1.$$

During diagnostic evaluation, we use four interventions per available channel:

$$q_{\mathrm{eval}} = 4.$$

The same intervened inputs are used for the previous checkpoint and current checkpoint, so teacher–student reliance vectors are compared under identical counterfactual views.

#### B.4.6 Evidence Channel Taxonomy

We instantiate evidence channels according to the input structure of each dataset:

- **Image channel:** image patches, visual tokens, or annotated regions.
- **Question channel:** instruction tokens and content-bearing question tokens.
- **OCR channel:** recognized text tokens and OCR bounding boxes.
- **Document channel:** document text spans, line blocks, and layout regions.
- **Chart/table channel:** axes, legends, plotted marks, table cells, and numerical entries.
- **Language-prior channel:** lexical cues in the question that can induce shortcut answers.

We omit OCR, document, and chart channels when the corresponding annotations are unavailable. Reliance vectors are always normalized over the channels available in the current sample.

#### B.4.7 Counterfactual Intervention Construction

We estimate reliance using evidence-suppressing interventions. Task-preserving perturbations are reserved for robustness analysis. Suppressing a channel lets us measure changes in the output distribution or answer likelihood.

**Image interventions.** We mask image patches, remove selected visual tokens, or replace visual regions with neutral image statistics. If object annotations are available, regions are sampled from annotated objects; otherwise, patches are sampled from a uniform grid.

**OCR interventions.** We remove OCR tokens, replace them with a neutral placeholder, or mask OCR spans while preserving the prompt template. When OCR bounding boxes are available, visual regions corresponding to OCR spans are also masked in an aligned intervention.

**Document interventions.** We mask document lines, paragraph blocks, or layout regions. For DocVQA-style inputs, we preserve page order and prompt structure while suppressing the selected document evidence.

**Chart and table interventions.** We mask axis labels, legends, plotted elements, cell values, and numerical entries. For tables, we distinguish header masking from body-cell masking because these two perturbations often affect different reasoning paths.

**Question-prior interventions.** We mask content words in the question while preserving stop words, formatting tokens, and instruction templates. This intervention estimates reliance on question-side language priors relative to grounded visual or textual evidence.

### B.4.8 Compute Resources

All main experiments are run on $8$ NVIDIA A100-80GB GPUs. For LLaVA-1.5-7B experiments, each continual stage uses approximately $6$–$10$ GPU hours depending on the dataset size and intervention budget. For LLaVA-1.5-13B, each stage uses approximately $12$–$18$ GPU hours. For InternVL-Chat-7B, each stage uses approximately $8$–$12$ GPU hours. The full main benchmark suite requires approximately $1{,}200$–$1{,}600$ A100 GPU hours, including three random seeds, ablations, and diagnostic evaluation. Reliance diagnostics require additional counterfactual forward passes during training and evaluation.

### B.4.9 Random Seeds and Reporting

Unless otherwise specified, we report the mean and standard deviation over three random seeds. The seeds affect model initialization of PEFT parameters, task sampling order when multiple official orders are available, data shuffling, and intervention sampling. For main tables, we report mean performance. For HFR, we compute Eq. (19) separately for each seed and report the arithmetic mean of the resulting rates.

All reliance-aware metrics are computed on the same evaluation examples across methods to avoid intervention-sampling artifacts.

### B.4.10 Inference Cost

RCL uses counterfactual interventions and the frozen previous checkpoint only during training and diagnostic evaluation. The updated PEFT-adapted MLLM is deployed directly after each stage.

RCL therefore retains the inference-time architecture, memory footprint, and latency of the underlying LoRA-adapted backbone.

### B.4.11 Hyperparameter Sensitivity

We evaluate sensitivity to three key hyperparameters:

$$\lambda_{\mathrm{rel}} \in \{0.25, 0.5, 1.0, 2.0\}, \qquad \tau_r \in \{0.25, 0.5, 1.0\}, \qquad q_{\mathrm{train}} \in \{1, 2, 4\}.$$

We report both answer-level metrics and reliance-aware metrics. This analysis tests whether reliance preservation is robust to the strength of the reliance loss, the sharpness of the reliance distribution, and the number of counterfactual interventions used during training.

### B.4.12 Ablation Settings

We evaluate the following ablations:

- **w/o reliance loss:** removes $\mathcal{L}_{\mathrm{rel}}$ and keeps only task learning and prediction distillation.
- **w/o prediction distillation:** removes $\mathcal{L}_{\mathrm{pred}}$ and keeps task learning and reliance preservation.
- **w/o confidence gate:** sets $w_{\theta^-}(x, y) = 1$ for all examples.
- **visual-only reliance:** preserves reliance only for visual channels.
- **text/OCR-only reliance:** preserves reliance only for text and OCR channels.
- **random interventions:** replaces evidence-targeted interventions with random token or patch masking.
- **larger intervention budget:** increases $q_{\mathrm{train}}$ from $1$ to $2$ or $4$.

These variants assess the roles of reliance preservation, output distillation, confidence weighting, and evidence-targeted interventions.

## B.5 Main Experiment Details

**Answer-drop–MRD bubble protocol.** Figure 3 compares answer drop, reliance drift, and dominant evidence flips across diagnostic slices. For every method $\mathcal{A}$, random seed, old stage $j$, later checkpoint

$k > j$, and diagnostic slice $\mathcal{E}_{j,s}$, we compute the clean answer drop

$$\Delta^{(\mathcal{A})}_{\text{ans}}(j,k,s) = \text{Acc}\Big(f^{(\mathcal{A})}_{\theta_j}; \mathcal{E}_{j,s}\Big) - \text{Acc}\Big(f^{(\mathcal{A})}_{\theta_k}; \mathcal{E}_{j,s}\Big). \tag{34}$$

We compute modality reliance drift using the reliance vectors defined in Section 4.2:

$$\text{MRD}^{(\mathcal{A})}_{j\to k,s} = \frac{1}{|\mathcal{E}_{j,s}|} \sum_{(x,y)\in\mathcal{E}_{j,s}} D_{\text{JS}}\left(\boldsymbol{r}^{(\mathcal{A})}_{\theta_j}(x,y) \Big\| \boldsymbol{r}^{(\mathcal{A})}_{\theta_k}(x,y)\right). \tag{35}$$

The dominant evidence flip rate is computed only on examples whose answers are preserved:

$$\begin{aligned}\text{DEF}^{(\mathcal{A})}_{j\to k,s} = \mathbb{E}_{(x,y)\in\mathcal{E}_{j,s}}\Big[ \\ \mathbf{1}\Big[\arg\max_{m\in\mathcal{M}_x} r^{(\mathcal{A})}_{\theta_j,m}(x,y) \neq \arg\max_{m\in\mathcal{M}_x} r^{(\mathcal{A})}_{\theta_k,m}(x,y)\Big] \\ \Big| \; f^{(\mathcal{A})}_{\theta_j}(x) = f^{(\mathcal{A})}_{\theta_k}(x) = y\Big].\end{aligned} \tag{36}$$

Each bubble in Figure 3 corresponds to one tuple

$$(\mathcal{A},\ \text{seed},\ j,\ k,\ s).$$

The horizontal coordinate is Eq. (34), the vertical coordinate is Eq. (35), and the bubble area is proportional to Eq. (36). We mark a slice as hidden forgetting when

$$\Delta^{(\mathcal{A})}_{\text{ans}}(j,k,s) \le 2\%, \qquad \text{MRD}^{(\mathcal{A})}_{j\to k,s} \ge 0.20. \tag{37}$$

The bubble plot applies the slice criterion in Eq. (6) to transitions across later checkpoints. Bubble-plot summaries average over old-to-later transitions, diagnostic slices, and seeds. For the stream-level HFR in Eq. (19), only transitions to the final checkpoint enter the average.

**Dominant-channel Sankey protocol.** Figure 5 summarizes changes in dominant evidence channels. For each method $\mathcal{A}$ and old stage $j$, we retain only answer-preserved examples:

$$\Omega^{(\mathcal{A})}_j = \Big\{(x,y)\in\mathcal{E}_j : f^{(\mathcal{A})}_{\theta_j}(x) = f^{(\mathcal{A})}_{\theta_K}(x) = y\Big\}, \tag{38}$$

where $K$ denotes the final continual stage. For each example, we identify the dominant evidence channel before and after continual adaptation:

$$c^{(\mathcal{A})}_j(x,y) = \arg\max_{m\in\mathcal{M}_x} r^{(\mathcal{A})}_{\theta_j,m}(x,y), \qquad c^{(\mathcal{A})}_K(x,y) = \arg\max_{m\in\mathcal{M}_x} r^{(\mathcal{A})}_{\theta_K,m}(x,y). \tag{39}$$

The Sankey transition probability from old dominant channel $u$ to final dominant channel $v$ is

$$T^{(\mathcal{A})}_{u\to v} = \frac{\sum_{j=1}^{K-1}\sum_{(x,y)\in\Omega^{(\mathcal{A})}_j} \mathbf{1}\Big[c^{(\mathcal{A})}_j(x,y)=u\Big]\,\mathbf{1}\Big[c^{(\mathcal{A})}_K(x,y)=v\Big]}{\sum_{j=1}^{K-1}\sum_{(x,y)\in\Omega^{(\mathcal{A})}_j} \mathbf{1}\Big[c^{(\mathcal{A})}_j(x,y)=u\Big] + \epsilon}. \tag{40}$$

Diagonal retention is computed as

$$\text{DiagRet}^{(\mathcal{A})} = \frac{\sum_{j=1}^{K-1}\sum_{(x,y)\in\Omega^{(\mathcal{A})}_j} \mathbf{1}\Big[c^{(\mathcal{A})}_j(x,y)=c^{(\mathcal{A})}_K(x,y)\Big]}{\sum_{j=1}^{K-1}|\Omega^{(\mathcal{A})}_j| + \epsilon}. \tag{41}$$

Shortcut-oriented flip rate into the question-prior channel $q$ is

$$\text{QFlip}^{(\mathcal{A})} = \frac{\sum_{j=1}^{K-1}\sum_{(x,y)\in\Omega^{(\mathcal{A})}_j} \mathbf{1}\Big[c^{(\mathcal{A})}_j(x,y)\neq q\Big]\,\mathbf{1}\Big[c^{(\mathcal{A})}_K(x,y)=q\Big]}{\sum_{j=1}^{K-1}|\Omega^{(\mathcal{A})}_j| + \epsilon}. \tag{42}$$

In Figure 5, ribbon width is proportional to Eq. (40). Red-tinted ribbons correspond to transitions whose final channel is the question-prior channel. We additionally report channel-specific diagonal retention for OCR, chart, and document channels to quantify whether structured evidence remains the dominant support for answer-preserved examples.

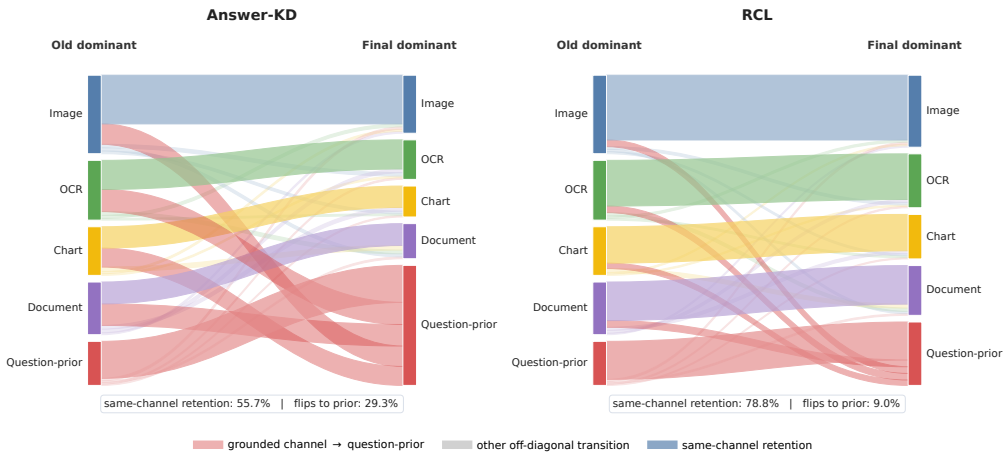


Figure 5: **Dominant evidence-channel transitions on answer-preserved examples.** Ribbons show how the dominant evidence channel changes from the old checkpoint to the final checkpoint, conditioned on both checkpoints answering correctly. Red-tinted ribbons denote shortcut-oriented flips into the question-prior channel. Protocol details are in Appendix B.5, *Dominant-channel Sankey protocol*.

**Cost–performance Pareto protocol.** Figure 4 compares training cost and hidden forgetting across configurations. We compare the following configurations: Seq-LoRA, ANSWER-KD, random intervention regularization, sparse RCL, and RCL with intervention budgets $q_{\text{train}} \in \{1, 2, 4\}$. For each configuration $\mathcal{A}$, the relative training cost is

$$C_{\text{rel}}(\mathcal{A}) = \frac{\text{GPUHours}(\mathcal{A})}{\text{GPUHours}(\text{Seq-LoRA})}. \tag{43}$$

The hidden-forgetting reduction over Seq-LoRA is

$$\text{HFR-Red.}(\mathcal{A}) = \frac{\text{HFR}(\text{Seq-LoRA}) - \text{HFR}(\mathcal{A})}{\text{HFR}(\text{Seq-LoRA})} \times 100. \tag{44}$$

A method is Pareto-dominated if another method achieves both lower or equal relative cost and higher or equal HFR reduction, with at least one strict improvement:

$$\mathcal{A} \prec \mathcal{B} \quad \Longleftrightarrow \quad C_{\text{rel}}(\mathcal{B}) \le C_{\text{rel}}(\mathcal{A}), \quad \text{HFR-Red.}(\mathcal{B}) \ge \text{HFR-Red.}(\mathcal{A}), \tag{45}$$

with one inequality strict. In Figure 4, the $x$-axis is Eq. (43), the $y$-axis is Eq. (44), bubble area is proportional to final average answer performance, and color encodes MRD. The default RCL configuration uses $q_{\text{train}} = 1$, i.e., one targeted intervention per available evidence channel. Because counterfactual interventions are used only during training and diagnostic evaluation, all points in the figure have the same inference-time architecture and latency as their corresponding PEFT-adapted backbone.

# C Additional Experimental Results

## C.1 Dominant Evidence-Channel Transitions

For each old-stage example answered correctly by both the old and final checkpoints, we compute the dominant channel before and after adaptation and aggregate the resulting transition matrix. In Figure 5, ANSWER-KD retains many correct answers. Among these examples, $55.7\%$ keep the same dominant channel and $29.3\%$ switch to the question-prior channel. The shortcut-oriented transitions OCR$\rightarrow$question-prior, chart$\rightarrow$question-prior, and document$\rightarrow$question-prior together account for $18.6\%$ of all answer-preserved examples. RCL increases diagonal retention to $78.8\%$ and reduces question-prior flips to $9.0\%$, with especially large gains for OCR, chart, and document evidence. Preserving the evidence behind correct answers accounts for the lower DEF and HFR of RCL.

Table 5: **Task-order robustness on the evidence-sensitive stream.** RCL shows stable reliance preservation across the default, reversed, and random task orders.

| Method | Task order | Avg.↑ | AF↓ | MRD↓ | DEF↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|---|---|
| Seq-LoRA | Default | 48.51 | 18.58 | 0.291 | 42.0 | 61.5 | 64.9 |
| Seq-LoRA | Reversed | 47.84 | 19.21 | 0.304 | 44.3 | 64.2 | 63.8 |
| Seq-LoRA | Random avg. | $48.23_{\pm 0.62}$ | $18.96_{\pm 0.71}$ | $0.298_{\pm 0.012}$ | $43.1_{\pm 1.6}$ | $62.7_{\pm 2.0}$ | $64.2_{\pm 1.1}$ |
| Answer-KD | Default | 50.88 | 16.45 | 0.250 | 36.1 | 52.9 | 67.1 |
| Answer-KD | Reversed | 50.19 | 16.88 | 0.261 | 37.4 | 55.0 | 66.4 |
| Answer-KD | Random avg. | $50.61_{\pm 0.55}$ | $16.70_{\pm 0.62}$ | $0.256_{\pm 0.010}$ | $36.8_{\pm 1.3}$ | $53.8_{\pm 1.8}$ | $66.8_{\pm 0.9}$ |
| RCL | Default | **64.06** | **4.22** | **0.106** | **14.8** | **12.9** | **89.0** |
| RCL | Reversed | 63.52 | 4.61 | 0.114 | 15.7 | 14.1 | 88.2 |
| RCL | Random avg. | $63.78_{\pm 0.41}$ | $4.44_{\pm 0.32}$ | $0.111_{\pm 0.006}$ | $15.3_{\pm 0.8}$ | $13.6_{\pm 1.0}$ | $88.6_{\pm 0.6}$ |

Table 6: **Validation of counterfactual reliance against gradient×activation attribution.** RCL produces reliance profiles that are more consistent with independent local attribution, especially on OCR-, chart-, and document-centric examples.

| Task family | Spearman $\rho \uparrow$ | | | JS divergence $\downarrow$ | | |
|---|---|---|---|---|---|---|
| | Seq-LoRA | Answer-KD | RCL | Seq-LoRA | Answer-KD | RCL |
| Visual reasoning | 0.52 | 0.55 | **0.68** | 0.184 | 0.169 | **0.112** |
| Language-prior VQA | 0.45 | 0.48 | **0.61** | 0.206 | 0.193 | **0.137** |
| OCR / text reasoning | 0.49 | 0.51 | **0.70** | 0.198 | 0.181 | **0.105** |
| Chart reasoning | 0.46 | 0.50 | **0.69** | 0.213 | 0.190 | **0.109** |
| Document reasoning | 0.44 | 0.47 | **0.67** | 0.221 | 0.204 | **0.118** |
| Overall | 0.48 | 0.51 | **0.67** | 0.204 | 0.187 | **0.116** |

### C.2 Task-Order Robustness

**Protocol.** We assess how RCL performs across different task orders. In addition to the default stream used in the main paper, we evaluate the reversed stream and three random task orders. For each order, we report final average performance, average forgetting, and reliance-aware diagnostics. The random-order row reports the mean and standard deviation over three independently sampled task permutations.

**Analysis.** Table 5 shows that the performance of RCL is stable across task orders. The average performance varies by less than $0.6$ points between the default and reversed streams, and HFR remains below $15\%$ in all settings. Seq-LoRA and Answer-KD are more sensitive to order changes, especially in MRD and HFR. This suggests that reliance preservation stabilizes old evidence use across different orderings of OCR-, chart-, and document-centric tasks.

### C.3 Reliance Metric Validation

**Protocol.** MRD is computed from counterfactual evidence suppression. We assess whether the counterfactual channel ranking agrees with an independent gradient×activation attribution signal. For each answer-preserved example, we compute the Spearman rank correlation between the two channel rankings. We also report the Jensen–Shannon divergence between the normalized counterfactual reliance vector and the normalized attribution vector. Higher rank correlation and lower JS divergence indicate better agreement between the two diagnostic views.

**Analysis.** Table 6 supports the validity of counterfactual reliance as a diagnostic signal. Across all task families, counterfactual reliance correlates positively with gradient-based attribution, and the agreement is strongest under RCL. The largest improvements occur on OCR, chart, and document tasks, where evidence channels are structured and channel flips are more harmful. This agreement suggests that MRD tracks patterns of channel use reflected in local attribution.

Table 7: **Sensitivity to evidence-suppressing versus task-preserving perturbations.** RCL is more sensitive to true evidence removal while remaining stable under harmless perturbations.

| **Method** | ES-KL↑ | TP-KL↓ | Sep. ratio↑ | ES Acc. drop↑ | TP Acc. drop↓ | Stable-grounded score↑ |
|---|---|---|---|---|---|---|
| Seq-LoRA | 0.312 | 0.118 | 2.64 | 14.8 | 5.7 | 56.3 |
| Answer-KD | 0.284 | 0.102 | 2.78 | 12.9 | 4.9 | 59.1 |
| Random interventions | 0.331 | 0.151 | 2.19 | 15.6 | 7.4 | 55.8 |
| RCL | **0.386** | **0.081** | **4.77** | **18.2** | **3.1** | **72.6** |

Table 8: **Hidden forgetting under different levels of question-prior bias.** The gain of RCL is largest on high-prior-bias examples, where answer-level metrics are most likely to hide shortcut reliance.

| **Bias group** | **Method** | Avg.↑ | DEF↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|
| Low prior bias | Seq-LoRA | 51.8 | 31.5 | 43.7 | 72.4 |
| | Answer-KD | 53.2 | 27.9 | 36.8 | 74.6 |
| | RCL | **64.9** | **12.4** | **9.8** | **90.7** |
| Medium prior bias | Seq-LoRA | 49.1 | 41.8 | 60.9 | 64.1 |
| | Answer-KD | 51.0 | 35.7 | 52.1 | 67.3 |
| | RCL | **64.2** | **14.6** | **12.7** | **89.3** |
| High prior bias | Seq-LoRA | 45.7 | 52.6 | 78.4 | 56.9 |
| | Answer-KD | 48.3 | 44.8 | 69.3 | 60.2 |
| | RCL | **62.7** | **17.3** | **16.5** | **86.8** |

### C.4 Task-Preserving Perturbation Test

**Protocol.** A grounded model should respond to evidence removal and remain stable under harmless input variations. We compare two perturbation families. The first family contains evidence-suppressing interventions, such as masking OCR tokens, chart elements, document spans, or visual regions. The second family contains task-preserving perturbations, such as mild image noise, question paraphrases, OCR whitespace changes, and chart style changes that do not alter the answer. For each method, we report output KL shift and accuracy drop under both perturbation types, as well as the separation ratio between evidence-suppressing and task-preserving sensitivity.

**Analysis.** Table 7 shows that RCL increases sensitivity to evidence removal and reduces sensitivity to task-preserving changes.

Random interventions increase both ES-KL and TP-KL, consistent with sensitivity to perturbations in general. RCL achieves the highest separation ratio and the lowest task-preserving accuracy drop. This pattern supports selective sensitivity to task-relevant evidence and stability under changes in formatting or style.

### C.5 Bias-Stratified Hidden Forgetting

**Protocol.** We assess how the effect of RCL varies with the strength of language shortcuts. For each example, we estimate a question-prior score using a question-only model and divide the evaluation set into low-, medium-, and high-prior-bias groups. We then compare HFR, DEF, and GCR in each group.

**Analysis.** Table 8 shows that hidden forgetting becomes more severe as question-prior bias increases. For Seq-LoRA and Answer-KD, high-prior-bias examples have substantially higher DEF and HFR, indicating that correct answers are often preserved through shortcut reliance. RCL reduces HFR in all groups and achieves the largest absolute reduction in the high-bias group, decreasing HFR from $69.3\%$ under Answer-KD to $16.5\%$. This confirms that reliance preservation is particularly useful when answer correctness alone is least reliable as a grounding indicator.

Table 9: **Dominant evidence-channel retention by task family.** RCL improves retention of structured evidence channels and reduces flips into question-side priors.

| Task family | Method | Image Ret.↑ | OCR Ret.↑ | Chart Ret.↑ | Doc Ret.↑ | Flip to prior↓ |
|---|---|---|---|---|---|---|
| Visual reasoning | Seq-LoRA | 58.6 | — | — | — | 24.9 |
| | Answer-KD | 62.0 | — | — | — | 26.0 |
| | RCL | **82.5** | — | — | — | **8.5** |
| Language-prior VQA | Seq-LoRA | 51.3 | — | — | — | 34.6 |
| | Answer-KD | 55.8 | — | — | — | 31.2 |
| | RCL | **77.4** | — | — | — | **11.8** |
| OCR / text reasoning | Seq-LoRA | 49.2 | 43.5 | — | 50.1 | 39.8 |
| | Answer-KD | 53.6 | 48.7 | — | 52.4 | 37.0 |
| | RCL | **76.8** | **76.4** | — | **73.9** | **10.9** |
| Chart reasoning | Seq-LoRA | 46.7 | — | 39.1 | — | 42.7 |
| | Answer-KD | 50.8 | — | 43.9 | — | 39.0 |
| | RCL | **74.1** | — | **74.8** | — | **11.1** |
| Document reasoning | Seq-LoRA | 45.9 | 46.8 | — | 38.6 | 45.1 |
| | Answer-KD | 49.7 | 51.3 | — | 41.5 | 40.9 |
| | RCL | **72.9** | **73.5** | — | **73.2** | **12.8** |

Table 10: **Sensitivity to reliance-loss weight $\lambda_{\text{rel}}$.** The default value $\lambda_{\text{rel}} = 1.0$ provides the best balance between answer performance and reliance preservation.

| $\lambda_{\text{rel}}$ | Avg.↑ | AF↓ | MRD↓ | DEF↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|---|
| 0.00 | 61.72 | 5.96 | 0.184 | 25.9 | 30.4 | 81.7 |
| 0.25 | 62.91 | 5.13 | 0.142 | 20.8 | 21.7 | 85.2 |
| 0.50 | 63.63 | 4.62 | 0.119 | 16.9 | 15.8 | 87.6 |
| 1.00 | **64.06** | **4.22** | **0.106** | **14.8** | **12.9** | **89.0** |
| 2.00 | 63.74 | 4.35 | 0.101 | 14.5 | 12.4 | 89.2 |

### C.6 Evidence-Channel Retention by Task Family

**Protocol.** We further analyze which evidence channels benefit most from RCL. For each old-task family, we compute the retention rate of the originally dominant grounded channel on answer-preserved examples. For example, OCR retention measures the fraction of answer-preserved examples whose dominant channel remains OCR after continual adaptation, conditioned on OCR being the old dominant channel. We also report the shortcut-flip rate into the question-prior channel. These rates track the identity of the highest-reliance channel. The GCR metric measures the fraction of grounded reliance mass preserved, including when the dominant channel changes.

**Analysis.** Table 9 shows that the largest improvements occur on structured evidence channels. For Answer-KD, OCR, chart, and document retention remains below $50\%$ in several task families, even though the answers are preserved. Output distillation thus often retains correct answers alongside shifts toward question priors. RCL improves OCR retention to $76.4\%$, chart retention to $74.8\%$, and document retention to $73.2\%$, while reducing flips into question-prior reliance to around 9–$13\%$. The channel-level results link RCL's gains to preservation of grounded evidence use beyond parameter stability alone.

### C.7 Hyperparameter Sensitivity

**Protocol.** We evaluate sensitivity to the reliance-loss weight $\lambda_{\text{rel}}$ and the reliance temperature $\tau_r$. The first parameter controls the strength of reliance preservation, while the second controls the sharpness of the normalized reliance vector. All other hyperparameters are fixed to the default setting. We report both answer-level and reliance-aware metrics.

Table 11: **Sensitivity to reliance temperature $\tau_r$.** Performance is stable around the default $\tau_r = 0.5$; overly sharp or overly smooth reliance distributions slightly weaken the trade-off.

| $\tau_r$ | Avg.↑ | AF↓ | MRD↓ | DEF↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|---|
| 0.25 | 63.58 | 4.47 | 0.113 | 16.1 | 14.8 | 87.9 |
| 0.50 | **64.06** | **4.22** | **0.106** | **14.8** | **12.9** | **89.0** |
| 1.00 | 63.79 | 4.39 | 0.116 | 16.6 | 15.4 | 87.8 |

Table 12: **Confidence-gate analysis.** The gate is most helpful when teacher targets are unreliable, especially in high-entropy or low-sensitivity bins.

| **Teacher bin** | **Variant** | Avg.↑ | MRD↓ | HFR↓ | GCR↑ |
|---|---|---|---|---|---|
| Low entropy, high sensitivity | w/o gate | 65.1 | 0.092 | 9.8 | 91.3 |
| | full RCL | **65.4** | **0.087** | **8.6** | **92.0** |
| Low entropy, low sensitivity | w/o gate | 63.4 | 0.126 | 18.2 | 85.9 |
| | full RCL | **64.1** | **0.108** | **13.4** | **88.4** |
| High entropy, high sensitivity | w/o gate | 61.8 | 0.146 | 23.7 | 82.6 |
| | full RCL | **63.0** | **0.119** | **16.5** | **86.8** |
| High entropy, low sensitivity | w/o gate | 59.6 | 0.171 | 31.5 | 77.8 |
| | full RCL | **61.7** | **0.132** | **20.9** | **83.5** |

**Analysis.** Tables 10 and 11 show that RCL is stable around its default hyperparameters. Removing reliance preservation entirely produces the largest degradation in MRD, DEF, and HFR. Increasing $\lambda_{\text{rel}}$ from $0.25$ to $1.0$ consistently improves reliance preservation, while $\lambda_{\text{rel}} = 2.0$ gives only a small additional reduction in MRD but slightly lowers average answer performance. This suggests a mild over-conservation effect when the reliance constraint is too strong. For $\tau_r$, the default value $0.5$ performs best, but both neighboring settings remain competitive.

### C.8 Confidence-Gate Analysis

**Protocol.** The confidence gate downweights unreliable teacher reliance targets. To analyze its effect, we bin samples by teacher predictive entropy and total intervention sensitivity. Low entropy indicates confident teacher predictions, while high intervention sensitivity indicates that the teacher has a measurable evidence-use pattern. For each bin, we compare full RCL with the variant that removes the gate.

**Analysis.** Table 12 shows that the confidence gate is most useful when the teacher's reliance estimates are unreliable. When the teacher is confident and evidence-sensitive, removing the gate causes only a small degradation. The gap widens when teacher entropy is high or intervention sensitivity is low. In the high-entropy, low-sensitivity bin, the gate reduces HFR from $31.5\%$ to $20.9\%$ and improves GCR from $77.8\%$ to $83.5\%$. The pattern indicates that the gate helps prevent the student from inheriting noisy or weakly grounded teacher reliance profiles.

### C.9 Failure-Case Taxonomy

**Protocol.** We categorize the remaining high-MRD examples under the default RCL model. We sample 300 answer-preserved examples with the highest MRD and manually categorize the dominant failure source.

**Analysis.** Table 13 shows several distinct sources of error under RCL. Incorrect teacher reliance accounted for 30 of the 300 reviewed high-MRD cases ($10.0\%$), suggesting that the confidence gate is largely effective. The largest categories involve ambiguous evidence, OCR noise, chart parsing ambiguity, and document layout mismatch. These categories reflect the difficulty of separating evidence channels in multimodal reasoning. Question-prior shortcuts are uncommon among these cases, supporting the conclusion that RCL substantially mitigates hidden evidence-use forgetting.

Table 13: **Failure-case taxonomy for high-MRD examples under RCL.** Ambiguous or noisy evidence accounts for most remaining failures. Systematic shifts to language priors are less common.

| Failure type | Count | Share | Typical pattern | Interpretation |
|---|---|---|---|---|
| Ambiguous multi-channel evidence | 83 | 27.7% | The same answer can be supported by both visual context and text/OCR evidence. | Reliance drift is partly benign because multiple evidence channels are valid. |
| Noisy or incomplete OCR | 64 | 21.3% | The old checkpoint relies on OCR tokens that are partially missing or incorrectly recognized. | Teacher reliance can be imperfect when the extracted evidence is noisy. |
| Chart parsing ambiguity | 55 | 18.3% | Axis labels, legends, or plotted marks are visually crowded and hard to isolate. | Structured visual evidence remains difficult under coarse channel interventions. |
| Document layout mismatch | 45 | 15.0% | The answer depends on layout relations between nearby text blocks. Isolated spans provide insufficient context. | Better layout-aware interventions may further improve reliance estimation. |
| Incorrect teacher reliance | 30 | 10.0% | The previous checkpoint answers correctly but already relies on a shortcut channel. | The gate reduces but cannot fully remove imperfect teacher targets. |
| Answer alias or evaluator ambiguity | 23 | 7.7% | Multiple textual forms are accepted, but reliance vectors differ across aliases. | Answer normalization accounts for some apparent reliance drift. |
| **Total** | **300** | **100.0%** | | |

# D Additional Discussion

## D.1 Limitations

The proposed RCL framework assumes that each input can be decomposed into meaningful evidence channels, such as image tokens, question tokens, OCR tokens, chart elements, document spans, or table cells. The benchmarks used here have identifiable evidence channels. Applications with more implicit or weakly structured inputs [Zhang et al., 2024] may require task-specific channel construction. For instance, embodied reference understanding draws on both linguistic descriptions and pointing gestures to locate a referent [Li et al., 2023]. In cybersecurity, threat knowledge graphs organize evidence for threat attribution [Wang et al., 2026], while provenance graphs represent system activity for anomaly detection [Ma et al., 2026].

RCL's counterfactual interventions add computation during training and diagnostic evaluation. The deployed model retains the same architecture, memory footprint, and latency. We control this cost by sampling a small fixed number of interventions per available channel and sharing the intervened inputs between the teacher and student.

Our empirical evaluation covers standard continual multimodal instruction tuning benchmarks, an evidence-sensitive multimodal stream, and three representative MLLM backbones.

Further applications include longer streams [Zhang et al., 2026a,b], additional modalities such as video [Zhang et al., 2023] and audio, and more specialized deployment domains [Wang et al., 2025, Fan et al., 2026, He et al., 2026, Huang et al., 2026, Xiao et al., 2026b, Lu et al., 2026, Tang et al., 2026, Zhang et al., 2026c]. For video, the affordance tokens used in continual video–language learning [Xu et al., 2026] could inform the construction of temporally localized evidence channels. In healthcare, episode-level benchmarks have been proposed to assess state continuity and evidence traceability across multi-step workflows [Yao et al., 2026].

## D.2 Broader Impacts

Reliance-aware evaluation can improve the reliability and interpretability of continual multimodal adaptation. In deployed multimodal assistants, preserving only the final answer may be insufficient [Agrawal et al., 2018, Liu et al., 2026, Li et al., 2026]: a model can appear accurate while gradually relying less on grounded visual, textual, OCR, chart, or document evidence. By explicitly measuring and constraining evidence-use drift, RCL encourages models to preserve the evidence path behind

correct answers, which may benefit applications such as document understanding, chart reasoning, educational assistants, accessibility tools, and visual question answering systems.

Preserving a reference checkpoint's reliance pattern can also carry forward its biases, spurious correlations, or factual errors. High-stakes applications should pair reliance-aware continual learning with task-specific evaluations [Shao, 2026a,b, Yang et al., 2026b] for safety, fairness, privacy, and robustness [Hu, 2026, Wang and Saxena, 2026, Yang et al., 2026a, Fan et al., 2023]. In road-scene perception, language anchors have been used to detect unfamiliar obstacles without labeled anomaly examples [Tian et al., 2023]. RCL serves as a diagnostic and a training-time reliability mechanism.

The study uses existing assets and requires no new human-subject data, dataset scraping, or release of a high-risk pretrained model. The main environmental cost comes from additional training-time counterfactual forward passes. We mitigate this cost by keeping the backbone frozen, updating only PEFT parameters, using a small intervention budget, and removing all reliance-estimation modules at inference time.

### D.3 Existing Assets and Licenses

We cite the original sources of the public datasets, pretrained backbones, and baseline implementations and use them under their licenses, research restrictions, and access terms.

Raw datasets, pretrained weights, and third-party code are not redistributed.

The main benchmark assets include CoIN, COAST, MCITlib, ScienceQA, GQA, VQAv2, OK-VQA, TextVQA, OCR-VQA, ChartQA, and DocVQA. The main pretrained backbones include LLaVA-1.5-7B, LLaVA-1.5-13B, and InternVL-Chat-7B. Reproduction requires obtaining datasets and checkpoints from their official sources under the applicable license or access terms.

We implement baselines from published descriptions or adapt official code under its licensing and attribution terms. Assets without an explicit license are used under their stated research-access terms and are not redistributed.

### D.4 LLM Usage

The experiments use LLaVA-1.5 and InternVL-Chat. The main method uses no closed-source LLM for oracle evaluation, label generation, or hidden supervision.

LLM assistance may include wording, grammar correction, LaTeX formatting, drafting code snippets, debugging, and organizing notes. The authors determined the scientific claims, method, protocol, and reported results, and reviewed and approved the technical content and conclusions.

### D.5 Compute Resources

Main experiments use NVIDIA A100-80GB GPUs with bfloat16 mixed precision. Each job uses up to 8 GPUs, with CPU workers for loading, preprocessing, and evaluation. A typical node has 64–128 vCPUs and 512GB–1TB of host memory. Temporary storage totals approximately 4–6TB for datasets, checkpoints, logs, and counterfactual evaluation outputs.

Each stage takes approximately 6–10 A100 GPU hours for LLaVA-1.5-7B, 12–18 A100 GPU hours for LLaVA-1.5-13B, and 8–12 A100 GPU hours for InternVL-Chat-7B, depending on dataset size and channel count. These estimates cover task learning, prediction distillation, reliance estimation, and counterfactual forward passes.

The default budget is one intervention per available channel during training and four during diagnosis. These forward passes add cost relative to standard answer evaluation. Inference uses the underlying LoRA-adapted MLLM with the same architecture and latency, without the intervention module, reference model, or reliance estimator.

The reported experiments require approximately 1,200–1,600 A100 GPU hours, including the main comparisons, three random seeds, ablations, sensitivity analyses, and diagnostic plots. Preliminary diagnostics and configuration checks require approximately 200–300 additional A100 GPU hours.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction identify hidden evidence-use forgetting and describe RCL, including channel interventions, teacher–student reliance profiles, prediction preservation, and reliance preservation. Experiments cover CoIN, COAST, MCITlib, and the evidence-sensitive stream.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix D.1 discusses the need for identifiable evidence channels, the cost of training-time interventions, and the evaluation scope across public continual multimodal benchmarks and three MLLM backbones. The experiments examine how answer retention can hide evidence-use drift.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The mathematical content defines the continual learning setup, reliance estimator, drift metrics, and training objective. The results are empirical.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method and appendix specify task order, preprocessing, answer normalization, baseline settings, backbones, PEFT, interventions, hyperparameters, evaluation metrics, random seeds, and compute requirements.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code is not publicly available. The experiments use public benchmarks, and the appendix describes training, interventions, and evaluation.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main text and appendix describe the benchmark suites, task order, splits, normalization, metrics, matched baseline settings, and training configuration, including PEFT, optimizer, learning rate, batch size, epochs, and intervention budget.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The main tables report mean performance over three seeds. Appendix B.4.9 identifies the sources of variation: PEFT initialization, data shuffling, task-order sampling where applicable, and intervention sampling. The task-order analysis reports the mean and standard deviation over three sampled permutations.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D.5 lists GPU type and memory, estimated cost per stage and backbone, diagnostic overhead, and total A100 GPU hours for the main experiments, ablations, and diagnostics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research follows the NeurIPS Code of Ethics and uses existing datasets and backbones under their intended research terms. It involves no new human-subject data or release of a high-risk model or scraped dataset.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix D.2 discusses how preserving evidence use can improve reliability and how it can carry forward biases or errors from the reference checkpoint. High-stakes applications require task-specific evaluations of safety, bias, and privacy.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The contribution is a method and diagnostic protocol evaluated on existing benchmarks. No new high-risk model, generator, or scraped dataset is released.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: Appendix D.3 identifies the datasets, backbones, and baseline implementations, credits their creators, and describes licensing and access terms. Raw data and pretrained weights are not redistributed.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The evidence-sensitive stream is an evaluation protocol built from existing public datasets. The appendix documents its composition, preprocessing, and metrics. The paper releases no new standalone dataset, benchmark, checkpoint, or code asset.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The study involves no crowdsourcing, human participants, new annotations, or compensation to human workers.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Institutional review is inapplicable because the study involves no human participants, crowdsourcing, private participant data, or user studies.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Appendix D.4 describes the use of MLLMs as experimental backbones and the scope of LLM assistance. The experiments use open multimodal backbones, including LLaVA-1.5 and InternVL-Chat. The authors checked and approved the scientific claims, experimental design, and reported results.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.