ACTIVATION-GUIDED SPARSE CHANNEL EDITING FOR UNLEARNING IN MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal large language models (MLLMs) may memorize private, copyrighted, or unsafe visual information, making selective unlearning essential for their responsible deployment. Existing methods either update broad parameter regions or remove target-related neurons, which can damage representations shared with retained knowledge. To address this challenge, we propose Visual Activation-Guided Sparse Editing (VASE), a novel method for selective visual unlearning in MLLMs. VASE locates a small set of visual feed-forward network channels using normalized activation energy on forget images and learns sparse residual updates only for these channels. A bounded forgetting objective prevents excessive suppression, while output- and activation-level constraints preserve retained behavior. The learned updates are merged into the visual backbone without additional inference overhead. Experiments across multiple benchmarks and MLLM backbones show that VASE achieves state-of-the-art forgetting while preserving retained knowledge and general multimodal capabilities. Code will be released.

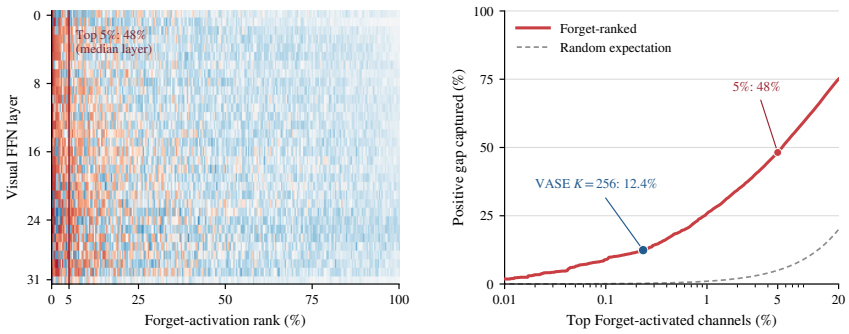
1 INTRODUCTION

Multimodal large language models (MLLMs) acquire broad real-world knowledge through large-scale training, enabling visual recognition, question answering, and reasoning (Liu et al., 2023; Bai et al., 2025). However, their training data may also contain private information, copyrighted content, and harmful knowledge (Liu et al., 2025a; Dontsov et al., 2025). MLLM unlearning seeks to remove specified knowledge without retraining from scratch. The task is difficult because target knowledge can be accessed through both images and text, while visual features are often shared across entities. Forgetting the target must therefore preserve unrelated visual knowledge and the model's general understanding and reasoning abilities (Huo et al., 2025; Wang et al., 2026).

Existing MLLM unlearning methods seek to protect retained abilities in two main ways. Optimization-based methods combine forget and retain objectives with constraints on parameters or visual representations (Huo et al., 2025; Wang et al., 2026; Cai et al., 2026). Localization-based methods identify target-related neurons, knowledge vectors, or cross-modal paths before pruning or editing them (Liu et al., 2025b; Kim et al., 2026; Li et al., 2026). The former can still change visual features shared with retain inputs, and continued optimization may suppress the target beyond the required level. The latter limits the edit region, but pruning or fixed attenuation can damage shared functions. Selective visual unlearning therefore needs to identify a compact support, edit it adaptively, and stop once sufficient forgetting is achieved.

To identify where selective editing may be effective, we examine visual FFN activations on forget and retain samples in Qwen2.5-VL-3B (Bai et al., 2025) before unlearning. Figure 1 shows that the positive Forget–Retain gap is concentrated among channels with high Forget activation: the top 5% capture 48% of the gap, and the 256 channels selected by VASE capture 12.4%. This concentration motivates editing a small set of highly active channels, although their potential contributions to retained concepts call for controlled updates. Together with the limitations of existing methods, this observation raises three questions: *where to edit*, *how to edit*, and *when to stop*. Our approach separates channel selection from update control, using forget samples to locate editable channels and retain samples to constrain changes to their shared functions.

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



(a) Layer-wise signed activation gap (b) Gap captured by Forget ranking

Figure 1: Concentration of pre-unlearning Forget–Retain activation differences in the visual encoder of Qwen2.5-VL-3B. (a) Signed activation gaps across visual FFN layers, with channels ordered within each layer by their Forget activation. Red and blue indicate stronger responses to Forget and Retain samples, respectively. The top 5% of channels account for a median of 48% of the positive gap across layers. (b) Cumulative positive gap captured by globally ranked channels. Among 109,440 channels, the top 5% capture 48% of the total positive gap, while the $K = 256$ channels selected by VASE capture 12.4%. The dashed curve denotes the expected coverage under uniform random selection.

To address these questions, we propose *Visual Activation-Guided Sparse Editing* (VASE), which localizes a small set of visual FFN channels and confines updates to their input-projection rows. VASE selects the top K channels through a global ranking of normalized activations on forget images, requiring only forward passes. It learns zero-initialized residual updates to the selected rows while keeping all other parameters frozen. A bounded answer-level objective limits target suppression, while token-level reference-margin and activation-preservation losses preserve retain behavior. The learned residuals are merged into the original weights without increasing inference cost. Extensive experiments on multiple multimodal unlearning benchmarks and MLLM backbones demonstrate effective target forgetting while preserving retain-set utility, text-only knowledge, and general multimodal capabilities.

Our contributions are summarized as follows:

- We show that Forget–Retain activation differences concentrate in a small fraction of visual FFN channels. This observation leads to a forget-only, forward-only criterion that selects channels globally across layers.
- We propose VASE, which confines residual updates to the selected input-projection rows. A bounded forgetting objective controls target suppression, while retain-preserving losses limit the effects on shared visual functions.
- We evaluate VASE across multiple multimodal unlearning benchmarks and MLLM backbones, demonstrating effective forgetting while preserving retain-set utility, text-only knowledge, and general multimodal capabilities.

2 Related Work

2.1 Machine Unlearning for Large Language Models

Machine unlearning for large language models (LLMs) removes designated knowledge without retraining from scratch while preserving general utility. Early methods pair forget-set optimization with retain-set training, and TOFU provides a controlled evaluation of this trade-off (Yao et al., 2023; Maini et al., 2024). Fan et al. (2025) remove the reference model from negative preference optimization (Zhang et al., 2024), while gradient rectification offers another way to stabilize updates (Wang et al., 2025b). Shi et al. (2025) broaden evaluation to privacy leakage, utility preservation, and repeated removal requests. These methods operate on text and do not distinguish visual from linguistic access to the same knowledge.

2.2 Machine Unlearning for Multimodal Large Language Models

Multimodal unlearning must consider both image-conditioned and text-only access to target knowledge. MLLMU-Bench and CLEAR evaluate forgetting across these settings (Liu et al., 2025a; Dontsov et al., 2025), while Patil et al. (2025) examine adversarial multimodal extraction. The desired scope of forgetting also varies: MMUnlearner suppresses visual access while preserving textual knowledge, whereas MANU addresses forgetting across modalities (Huo et al., 2025; Liu et al., 2025b). Optimization-based methods constrain updates through saliency, visual representations, or retain subspaces (Huo et al., 2025; Wang et al., 2026). Structure-aware methods localize neurons, knowledge vectors, or cross-modal paths before intervention (Liu et al., 2025b; Kim et al., 2026; Li et al., 2026). Zeng et al. (2025) use retain-guided masking to protect visual understanding. Broad updates can still disturb shared features, while fixed pruning or attenuation may discard useful capacity. This motivates a compact, learnable visual edit controlled by both forgetting and retention requirements.

2.3 Model Editing and Parameter-Efficient Adaptation

Model editing changes selected behaviors without full retraining. ROME and MEMIT modify feed-forward weights for factual updates (Meng et al., 2022; 2023); Gupta et al. (2024) extend this family to batched edits with EMMET. Mitchell et al. (2022) learn efficient gradient-based edits, while LoRA trains compact updates with most pretrained weights frozen (Hu et al., 2022). VLKEB documents the additional challenges of editing vision-language models (Huang et al., 2024). Unlike factual correction or insertion, multimodal unlearning must suppress target access while preserving shared visual functions. Our work combines localized visual support with explicit forgetting and retention constraints.

3 VASE: Visual Activation-Guided Sparse Editing

We propose *Visual Activation-Guided Sparse Editing* (VASE) for selective visual knowledge unlearning in multimodal large language models. VASE follows a locate-then-edit procedure. It first identifies a small set of visual feed-forward network channels from their normalized activation statistics on the forget set. Then, it learns sparse residual updates restricted to the projections that generate the selected channels. The language model and all parameters outside the selected visual support remain frozen throughout the editing process. After optimization, the learned residuals are merged into the original model weights without introducing additional inference-time modules.

3.1 Problem Setup

A multimodal large language model $\mathcal{M}$ usually consists of a vision encoder $\mathcal{V}$, a modality projector $\mathcal{P}$, and a large language model $\mathcal{W}$. Let $\mathcal{D} = \{(v_i, q_i, a_i)\}_{i=1}^{N}$ denote a visual instruction dataset, where v_i, q_i, and a_i are an image, a textual question, and its answer. Given (v_i, q_i, a_i), the vision encoder and projector produce the visual embedding $\mathcal{P}(\mathcal{V}(v_i))$. The question q_i is converted into a textual embedding $\mathbf{q}_i^e$ and combined with the visual embedding for response generation:

$$\hat{a}_i = \mathcal{W}\left(\mathbf{q}_i^e \oplus \mathcal{P}\big(\mathcal{V}(v_i)\big)\right), \tag{1}$$

where $\oplus$ denotes sequence concatenation.

The dataset is partitioned into a forget set $\mathcal{D}_f$ and a retain set $\mathcal{D}_r$, with $\mathcal{D} = \mathcal{D}_f \cup \mathcal{D}_r$ and $\mathcal{D}_f \cap \mathcal{D}_r = \varnothing$. The forget set contains samples associated with the target entities, while the retain set contains all other samples. In this work, we consider *selective visual knowledge unlearning*. The goal is to suppress image-conditioned knowledge of the target entities while preserving the model's behavior on non-target samples.

Let $\mathcal{M}'$ denote the model obtained after unlearning, and let $p_{\mathcal{M}}(a \mid v, q)$ denote the conditional probability assigned by model $\mathcal{M}$ to answer a. For a forget sample $(v^f, q^f, a^f) \in \mathcal{D}_f$ and a retain sample $(v^r, q^r, a^r) \in \mathcal{D}_r$, the desired behavior is

$$\begin{aligned} p_{\mathcal{M}'}\left(a^f \mid v^f, q^f\right) &\ll p_{\mathcal{M}}\left(a^f \mid v^f, q^f\right), \\ p_{\mathcal{M}'}\left(\cdot \mid v^r, q^r\right) &\approx p_{\mathcal{M}}\left(\cdot \mid v^r, q^r\right). \end{aligned} \tag{2}$$

Since the edit targets the visual pathway, $\mathcal{M}'$ should also preserve the text-only behavior of $\mathcal{M}$.

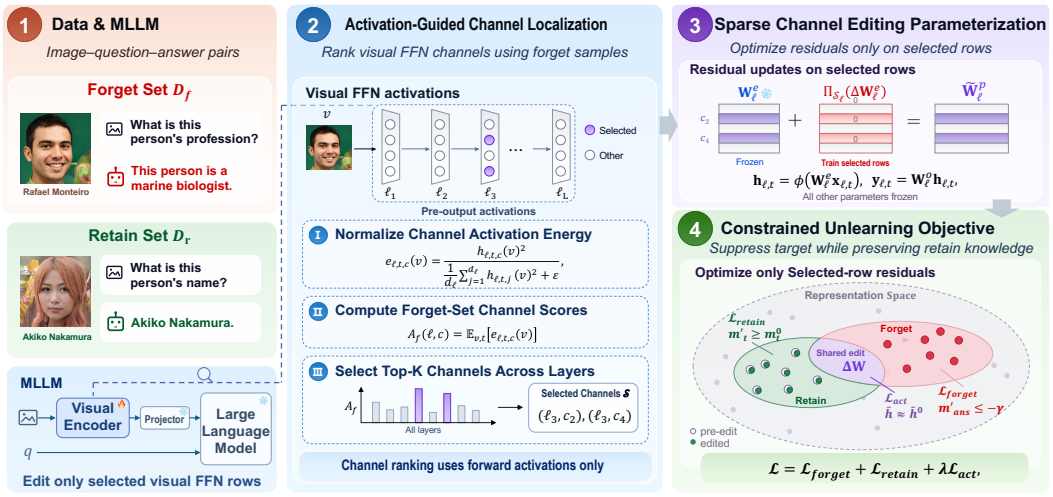


Figure 2: Overview of VASE. Forget-set activations identify visual FFN channels for sparse editing. Residual updates to the selected input-projection rows suppress target-answer confidence while preserving retain-sample margins and activations. The residuals are merged into the original weights without additional inference-time modules.

3.2 Visual Activation-Guided Channel Localization

We rank visual feed-forward network channels by their relative activation energy on the forget set and use the top-ranked channels as candidate editing locations. Consider the L feed-forward network layers in the vision encoder. For layer ℓ, let $\mathbf{h}_{\ell,t}(v) \in \mathbb{R}^{d_\ell}$ denote the intermediate activation of visual token t after the nonlinear transformation or gating operation and before the output projection. Its c-th channel is denoted by $h_{\ell,t,c}(v)$, where $c \in \{1, \ldots, d_\ell\}$.

Raw activation scales can differ across layers. We measure the squared response of each channel relative to the mean squared response of all channels for the same token:

$$e_{\ell,t,c}(v) = \frac{h_{\ell,t,c}(v)^2}{\frac{1}{d_\ell}\sum_{j=1}^{d_\ell} h_{\ell,t,j}(v)^2 + \varepsilon}, \tag{3}$$

where $\varepsilon > 0$ ensures numerical stability. This normalization places channel responses from different layers on a common relative scale.

We aggregate the normalized energy over the visual tokens and samples in the forget set:

$$A_f(\ell, c) = \frac{1}{|\mathcal{D}_f|} \sum_{(v,q,a)\in\mathcal{D}_f} \frac{1}{T_\ell(v)} \sum_{t=1}^{T_\ell(v)} e_{\ell,t,c}(v), \tag{4}$$

where $T_\ell(v)$ denotes the number of visual tokens at layer ℓ. Let

$$\mathcal{C} = \{(\ell, c) \mid \ell \in \{1, \ldots, L\},\ c \in \{1, \ldots, d_\ell\}\} \tag{5}$$

be the set of channels across all visual feed-forward network layers. Given an editing budget K, we select

$$\mathcal{S} = \mathrm{TopK}_{(\ell,c)\in\mathcal{C}}\big(A_f(\ell, c)\big), \qquad |\mathcal{S}| = K. \tag{6}$$

By aggregating normalized activation energy across visual tokens and forget samples, $A_f(\ell, c)$ highlights channels that respond consistently to the visual patterns associated with the target entities, while reducing sample-specific fluctuations and cross-layer scale bias. This produces a compact and stable editing support for subsequent sparse updates, focusing the intervention on relevant visual computations while limiting unnecessary changes elsewhere in the model.

3.3 Sparse Channel Editing Parameterization

After identifying the global channel support $\mathcal{S}$ in Eq. 6, we map the selected activations to the weight rows that generate them. For layer ℓ, the selected channel indices are

$$\mathcal{S}_\ell = \{c \mid (\ell, c) \in \mathcal{S}\}. \tag{7}$$

A layer remains unchanged when $\mathcal{S}_\ell = \varnothing$.

Let $\mathbf{x}_{\ell,t} \in \mathbb{R}^{d_\ell^{\text{in}}}$ denote the input of visual token t to the feed-forward network at layer ℓ. Bias terms are omitted for clarity. A standard feed-forward network, such as the vision encoder in InternVL3.5 (Wang et al., 2025a), computes

$$\mathbf{h}_{\ell,t} = \phi\left(\mathbf{W}_\ell^e \mathbf{x}_{\ell,t}\right), \qquad \mathbf{y}_{\ell,t} = \mathbf{W}_\ell^o \mathbf{h}_{\ell,t}, \tag{8}$$

where $\mathbf{W}_\ell^e$ and $\mathbf{W}_\ell^o$ are the expansion and output projections.

A gated feed-forward network, such as the vision encoder in Qwen2.5-VL (Bai et al., 2025), computes

$$\mathbf{h}_{\ell,t} = \phi\left(\mathbf{W}_\ell^g \mathbf{x}_{\ell,t}\right) \odot \left(\mathbf{W}_\ell^u \mathbf{x}_{\ell,t}\right), \qquad \mathbf{y}_{\ell,t} = \mathbf{W}_\ell^o \mathbf{h}_{\ell,t}, \tag{9}$$

where $\mathbf{W}_\ell^g$ and $\mathbf{W}_\ell^u$ are the gating and up projections. In both architectures, $\mathbf{h}_{\ell,t}$ is the pre-output activation used for channel localization in Eq. 3. Since its c-th channel is generated by the c-th row of the input-side projection, we restrict the residual updates to the rows indexed by $\mathcal{S}_\ell$. For $p = e$ in a standard feed-forward network and $p \in \{g, u\}$ in a gated feed-forward network, we define

$$\widetilde{\mathbf{W}}_\ell^p = \mathbf{W}_\ell^p + \Pi_{\mathcal{S}_\ell}\left(\Delta \mathbf{W}_\ell^p\right), \qquad \Delta \mathbf{W}_\ell^p \in \mathbb{R}^{|\mathcal{S}_\ell| \times d_\ell^{\text{in}}}, \tag{10}$$

where $\Pi_{\mathcal{S}_\ell}(\cdot)$ places the residual rows at the selected indices and sets all other rows to zero. For a gated feed-forward network, independent residuals are learned for the gating and up projections using the same channel support $\mathcal{S}_\ell$. All residuals are initialized to zero and are the only trainable parameters.

3.4 Constrained Unlearning Objective

We optimize the sparse residuals using three objectives: suppressing the target-answer likelihood on forget samples, preserving token-level margins on retain samples, and limiting retain-side activation drift. Let $\mathcal{T}(a)$ denote the supervised answer positions. For each $t \in \mathcal{T}(a)$, let $p_{t,a_t} = p(a_t \mid v, q, a_{<t})$ denote the target-token probability. Superscripts 0 and $'$ refer to quantities produced by the original model $\mathcal{M}$ and the edited model $\mathcal{M}'$, respectively.

Bounded target-likelihood suppression. For a forget sample, we define the length-normalized target-answer likelihood as

$$\bar{p}'(a \mid v, q) = \exp\left(\frac{1}{|\mathcal{T}(a)|} \sum_{t \in \mathcal{T}(a)} \log p'_{t,a_t}\right). \tag{11}$$

The geometric mean removes the dependence of answer likelihood on sequence length. We convert this likelihood into an answer-level log-odds score:

$$m'_{\text{ans}} = \log \frac{\bar{p}'(a \mid v, q)}{1 - \bar{p}'(a \mid v, q)}. \tag{12}$$

The forgetting loss imposes a negative boundary on this score:

$$\mathcal{L}_{\text{forget}} = \mathbb{E}_{(v,q,a)\sim\mathcal{D}_f}\left[\left[m'_{\text{ans}} + \gamma\right]_+\right], \tag{13}$$

where $[x]_+ = \max(x, 0)$ and $\gamma > 0$ is the forgetting margin. The loss becomes zero when $m'_{\text{ans}} \leq -\gamma$, or equivalently, $\bar{p}'(a \mid v, q) \leq \sigma(-\gamma)$. It suppresses the target answer only to the prescribed boundary, avoiding unnecessary updates after the forgetting criterion is met. The answer-level constraint also avoids forcing every target token below the same threshold.

Reference-margin preservation. For retain samples, we preserve the original model's token-level log-odds margins. Given the logits $\mathbf{z}_t$ at answer position t, the margin is

$$m_t = \log \frac{p_{t,a_t}}{1 - p_{t,a_t}} = z_{t,a_t} - \log \sum_{j \neq a_t} \exp\left(z_{t,j}\right). \tag{14}$$

This margin measures the preference for the target token over the aggregate probability mass of all alternatives. Using the original margin as a lower bound, we define

$$\mathcal{L}_{\text{retain}} = \mathbb{E}_{(v,q,a)\sim\mathcal{D}_r} \left[\frac{1}{|\mathcal{T}(a)|} \sum_{t \in \mathcal{T}(a)} \left[m_t^0 - m_t' \right]_+ \right]. \tag{15}$$

This one-sided loss penalizes only reductions from the reference margins, preserving the target-token preference without constraining margin increases.

Retain-activation preservation. The output-level loss preserves target-token margins but does not constrain changes within the edited visual channels. We therefore anchor their retain-set activations to the corresponding pre-edit values. Let $h^0_{\ell,t,c}$ and $h'_{\ell,t,c}$ denote the original and edited activations of channel (ℓ, c), and let $\mathbb{E}_r[\cdot]$ denote averaging over retain samples and their visual tokens. We define

$$r_{\ell,c} = \sqrt{\mathbb{E}_r\left[\left(h^0_{\ell,t,c}\right)^2\right] + \varepsilon}, \qquad \mathcal{L}_{\text{act}} = \frac{1}{|\mathcal{S}|} \sum_{(\ell,c)\in\mathcal{S}} \mathbb{E}_r \left[\left(\frac{h'_{\ell,t,c} - h^0_{\ell,t,c}}{r_{\ell,c}} \right)^2 \right]. \tag{16}$$

The RMS normalization prevents channels with large activation scales from dominating the loss. Combined with the forgetting loss, this term directs the edit toward forget-set behavior while preserving the selected-channel responses on retain inputs.

Overall objective. The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{forget}} + \mathcal{L}_{\text{retain}} + \lambda \mathcal{L}_{\text{act}}, \tag{17}$$

where λ controls the strength of retain-activation preservation. We minimize $\mathcal{L}$ only over the sparse residuals defined in Eq. 10. Together, these objectives suppress image-conditioned target knowledge while preserving predictive behavior and internal activations on retain samples.

4 Experiments

4.1 Experimental Setup

Datasets and models. We evaluate on MLLMU-Bench (Liu et al., 2025a) and CLEAR (Dontsov et al., 2025) under their official 5% forget setting. Experiments use Qwen2.5-VL-3B (Bai et al., 2025) and InternVL3.5-8B (Wang et al., 2025a), which differ in scale and visual FFN design. All methods start from the same backbone-specific vanilla checkpoint trained on the full training set.

Baselines and evaluation. We compare VASE with GA (Thudi et al., 2022), GA Diff (Liu et al., 2022), KL Min (Nguyen et al., 2020), NPO (Zhang et al., 2024), MANU (Liu et al., 2025b), and MMUnlearner (Huo et al., 2025). Forget and Retain performance is measured using multiple-choice accuracy and open-ended ROUGE-L. We also report text-only QA and benchmark-specific real-world evaluations to measure textual knowledge preservation and general utility.

Implementation details. Vanilla checkpoints follow the official LoRA configurations. All methods share the same data splits, initial checkpoints, prompts, and decoding settings. VASE optimizes only sparse residuals on the selected visual channels, and all open-ended evaluations use greedy decoding. Dataset statistics, metric definitions, and complete hyperparameters are provided in Appendix A.

Table 1: Main results on MLLMU-Bench and CLEAR under the 5% forget setting. F and R denote Forget and Retain, while RW and RF denote Real-World and Real-Face. Acc and R-L denote accuracy and ROUGE-L, respectively. Arrows indicate the preferred direction. Bold and underlined values indicate the best and second-best results, respectively. Green and red deltas indicate favorable and unfavorable changes relative to Vanilla, respectively.

Method	Qwen2.5-VL-3B													
	MLLMU-Bench								CLEAR					
	F-VQA Acc ↓	F-VQA R-L ↓	F-QA Acc ↑	R-VQA Acc ↑	R-VQA R-L ↑	R-QA Acc ↑	RW-VQA Acc ↑	RW-QA Acc ↑	F-VQA Acc ↓	F-QA R-L ↑	R-VQA Acc ↑	R-QA R-L ↑	RW-VQA Acc ↑	RF-VQA Acc ↑
Vanilla	62.4%	0.631	**64.0%**	53.2%	0.555	**63.3%**	**69.7%**	**71.2%**	70.7%	**0.405**	63.5%	0.401	57.2%	91.4%
GA	52.0%	0.491	62.4%	49.8%	0.442	61.2%	67.9%	68.4%	67.6%	0.402	63.3%	0.396	55.6%	90.7%
GA Diff	59.2%	0.591	62.4%	52.8%	**0.573**	62.3%	67.4%	69.8%	60.6%	0.398	60.3%	0.401	57.5%	91.4%
KL Min	60.8%	0.579	62.4%	51.7%	0.548	61.9%	67.5%	69.9%	64.4%	0.401	63.5%	**0.403**	58.3%	92.1%
NPO	52.8%	0.507	63.2%	50.3%	0.437	62.6%	66.8%	69.9%	68.1%	0.402	62.3%	0.397	57.2%	91.4%
MANU	58.4%	0.619	60.8%	50.8%	0.527	58.4%	68.0%	69.8%	67.0%	0.400	64.3%	0.399	**58.6%**	92.1%
MMUnlearner	48.8%	0.520	63.2%	52.5%	0.548	62.5%	68.8%	70.5%	55.0%	0.400	63.0%	0.400	57.5%	91.4%
Ours	**17.6%**	**0.369**	**64.0%**	**53.7%**	0.554	**63.3%**	**69.7%**	**71.2%**	**32.4%**	**0.405**	**64.8%**	0.401	57.8%	**92.7%**
Δ *vs. Vanilla*	-44.8 pp	-0.262	0.0 pp	+0.5 pp	-0.001	0.0 pp	0.0 pp	0.0 pp	-38.3 pp	0.000	+1.3 pp	0.0 pp	+0.6 pp	+1.3 pp

Method	InternVL3.5-8B													
	MLLMU-Bench								CLEAR					
	F-VQA Acc ↓	F-VQA R-L ↓	F-QA Acc ↑	R-VQA Acc ↑	R-VQA R-L ↑	R-QA Acc ↑	RW-VQA Acc ↑	RW-QA Acc ↑	F-VQA Acc ↓	F-QA R-L ↑	R-VQA Acc ↑	R-QA R-L ↑	RW-VQA Acc ↑	RF-VQA Acc ↑
Vanilla	66.4%	0.625	63.2%	65.2%	0.580	62.8%	**70.1%**	77.1%	69.7%	**0.414**	66.1%	**0.414**	68.4%	**60.9%**
GA	64.0%	**0.229**	64.0%	63.5%	0.228	60.2%	60.3%	72.8%	68.6%	0.413	66.8%	0.413	**68.7%**	56.3%
GA Diff	64.0%	0.590	64.0%	64.4%	**0.589**	63.4%	67.9%	**77.6%**	64.4%	**0.414**	65.6%	0.411	67.6%	59.6%
KL Min	64.8%	0.533	62.4%	65.7%	0.480	61.0%	62.4%	75.6%	64.4%	**0.414**	62.3%	0.411	67.6%	59.6%
NPO	64.8%	0.581	63.2%	**67.6%**	0.541	63.3%	69.3%	77.3%	70.7%	0.411	66.6%	0.413	**68.7%**	56.3%
MANU	64.0%	0.611	**65.6%**	65.2%	0.568	**64.9%**	70.0%	76.2%	67.0%	0.409	66.3%	0.413	**68.7%**	60.3%
MMUnlearner	54.4%	0.540	63.2%	64.4%	0.570	62.5%	69.3%	76.5%	58.0%	0.410	65.8%	0.410	68.1%	58.9%
Ours	**24.0%**	0.476	63.2%	65.1%	0.579	62.8%	**70.1%**	77.1%	**36.7%**	**0.414**	**67.3%**	**0.414**	68.1%	58.3%
Δ *vs. Vanilla*	-42.4 pp	-0.149	0.0 pp	-0.1 pp	-0.001	0.0 pp	0.0 pp	0.0 pp	-33.0 pp	0.000	+1.2 pp	0.000	-0.3 pp	-2.6 pp

4.2 Main Results

Table 1 compares our method with representative multimodal unlearning baselines under the 5% forget setting. Across both backbones and datasets, our method achieves the strongest targeted visual forgetting. On MLLMU-Bench, it reduces F-VQA accuracy by 44.8 and 42.4 percentage points relative to Vanilla on Qwen2.5-VL-3B and InternVL3.5-8B, respectively, outperforming the strongest competing baseline by 31.2 and 30.4 points. Meanwhile, Retain and real-world performance remains close to the Vanilla model. Although GA obtains a lower F-VQA ROUGE-L on InternVL3.5-8B, it substantially degrades general utility, reducing RW-VQA accuracy from 70.1% to 60.3%.

The same pattern holds on CLEAR, where our method improves F-VQA forgetting over the strongest baseline by 22.6 points on Qwen2.5-VL-3B and 21.3 points on InternVL3.5-8B. On Qwen2.5-VL-3B, it also improves Retain, real-world, and real-face performance over Vanilla. On InternVL3.5-8B, these gains incur only minor decreases of 0.3 and 2.6 points on RW-VQA and RF-VQA, respectively. Overall, our method delivers a substantially stronger forgetting–utility trade-off while preserving textual knowledge and out-of-domain generalization.

4.3 Ablation Study

Table 2 evaluates the contribution of each component on Qwen2.5-VL-3B and InternVL3.5-8B. Replacing the proposed channel selector with random selection substantially increases Forget accuracy from 17.6% to 39.2% on Qwen2.5-VL-3B and from 24.0% to 42.4% on InternVL3.5-8B, while providing negligible utility gains. This confirms that effective unlearning depends on identifying influential visual channels rather than editing an arbitrary sparse subset. GA forgetting is consistently weaker than the full method. KL retaining is more competitive and achieves a slightly lower Forget ROUGE-L on Qwen2.5-VL-3B, but sacrifices Retain and real-world performance; the full model therefore provides a better overall forgetting–utility trade-off.

The preservation objectives are also essential. Removing $\mathcal{L}_{\text{retain}}$ reduces Retain accuracy from 53.7% to 28.7% on Qwen2.5-VL-3B and from 65.1% to 39.6% on InternVL3.5-8B. Removing

Table 2: Ablation study on MLLMU-Bench with a 5% forget ratio.

Variant	Qwen2.5-VL-3B					InternVL3.5-8B				
	F-VQA Acc ↓	F-VQA R-L ↓	R-VQA Acc ↑	R-VQA R-L ↑	RW-VQA Acc ↑	F-VQA Acc ↓	F-VQA R-L ↓	R-VQA Acc ↑	R-VQA R-L ↑	RW-VQA Acc ↑
VASE (Full)	**17.6%**	0.369	**53.7%**	**0.554**	69.7%	**24.0%**	**0.476**	**65.1%**	**0.579**	70.1%
Random Channel Selection	39.2%	0.538	53.3%	0.551	**70.0%**	42.4%	0.590	64.4%	0.571	**70.2%**
GA Forgetting	31.2%	0.517	52.5%	0.550	67.8%	36.0%	0.570	63.8%	0.569	68.3%
KL Retaining	20.0%	**0.349**	52.7%	0.546	67.0%	28.8%	0.484	64.8%	0.564	67.4%
w/o $\mathcal{L}_{\text{retain}}$	19.2%	0.436	28.7%	0.470	62.9%	28.0%	0.540	39.6%	0.488	63.3%
w/o $\mathcal{L}_{\text{act}}$	21.6%	0.429	52.1%	0.539	56.9%	31.2%	0.536	63.4%	0.557	57.3%

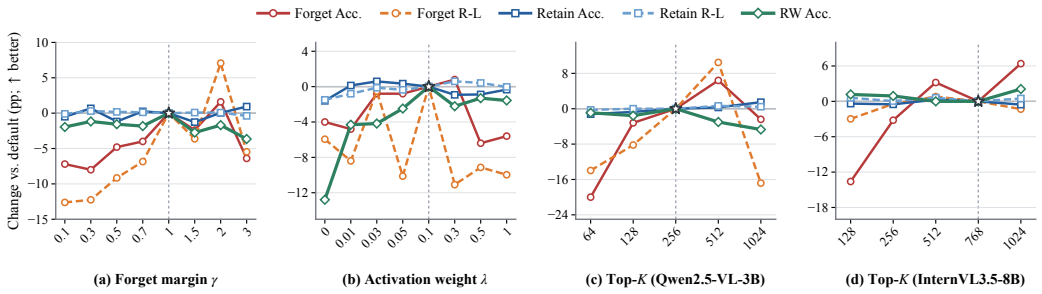


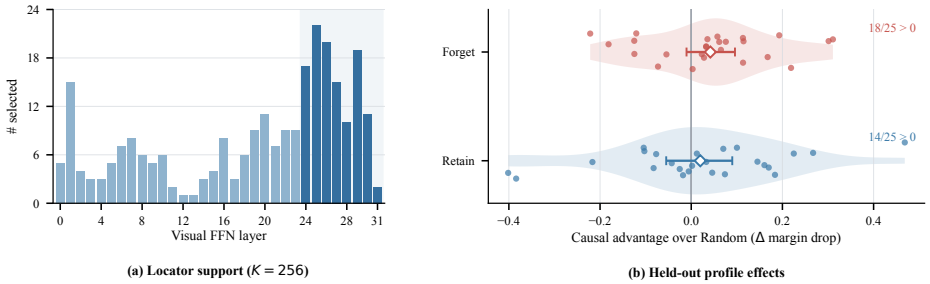
Figure 3: Hyperparameter sensitivity on MLLMU-Bench with a 5% forget ratio. Each curve reports the signed change, in percentage points (pp), relative to the default configuration; the signs of Forget metrics are reversed so that higher values consistently indicate better performance. Accuracy and ROUGE-L are denoted by Acc. and R-L, respectively.

$\mathcal{L}_{\text{act}}$ causes real-world accuracy to drop by 12.8 percentage points on both models, despite a smaller change in Retain accuracy. Together, these results show that $\mathcal{L}_{\text{retain}}$ preserves in-domain knowledge, whereas $\mathcal{L}_{\text{act}}$ is particularly important for maintaining general utility beyond the retain set.

4.4 Model Analysis

Hyperparameter sensitivity. Figure 3 shows that the forgetting margin γ and editing budget K provide controllable forgetting–utility trade-offs. On Qwen2.5-VL-3B, increasing γ or K initially improves forgetting, whereas overly large values cause degradation; the default settings therefore offer a balanced operating point. Removing activation preservation ($\lambda = 0$) decreases real-world accuracy by 12.8 percentage points, confirming its importance for general utility. InternVL3.5-8B remains comparatively stable across a broad range of K, with different budgets favoring classification- and generation-based forgetting. Overall, the method is robust around its defaults and is not dependent on a narrowly tuned hyperparameter configuration.

Localization analysis. Figure 4 evaluates channels localized on the training split. The causal probe uses unseen test images and questions from the same 25 Forget identities, totaling 125 VQA examples. Effects are averaged per identity. The selected channels concentrate in later FFN blocks, and their ablation yields a mean advantage of 0.042 over Random, with positive effects for 18 of 25 identities. Table 3 separates the effects of localization source and sparse parameterization. Random-selected samples the same number of visual channels uniformly, while Retain-selected and Unrelated-selected run the locator on Retain and unrelated images, respectively. These variants use the same editor, loss, parameter budget, and schedule as VASE. Matched Visual LoRA replaces the sparse editor with visual low-rank adapters of nearly equal size (0.658M vs. 0.655M). VASE reduces F-Acc from 39.2% to 17.6% and F-RL from 0.538 to 0.369 compared with Random, and it outperforms both alternative localization sources. It also improves F-Acc from 19.2% to 17.6% and F-RL from 0.482 to 0.369 over matched Visual LoRA, while retaining comparable utility. These results show that both Forget-conditioned localization and sparse targeted editing contribute to the final performance.



(a) Locator support ($K = 256$) **(b) Held-out profile effects**

Figure 4: Localization analysis on Qwen2.5-VL-3B. (a) Layer distribution of the $K = 256$ selected visual channels. (b) Profile-level margin-drop advantage over Random after ablating the selected parameter rows. Diamonds and error bars show the mean and profile-bootstrap 95% confidence interval. Positive values indicate a stronger causal effect than Random.

Table 3: Localization and parameterization controls on MLLMU-Bench using Qwen2.5-VL-3B under the 5% forget setting. All variants use the same training objective and optimization schedule. Visual LoRA is matched to VASE in trainable parameter count.

Variant	**Params**	**F-Acc ↓**	**F-RL ↓**	**R-Acc ↑**	**R-RL ↑**	**RW-Acc ↑**
Random-selected	0.655M	39.2%	0.538	53.3%	0.551	70.0%
Retain-selected	0.655M	21.6%	0.485	53.6%	0.555	66.3%
Unrelated-selected	0.655M	22.4%	0.468	53.0%	0.556	68.1%
Matched Visual LoRA	0.658M	19.2%	0.482	53.5%	**0.559**	**70.1%**
Forget-selected (VASE)	0.655M	**17.6%**	**0.369**	**53.7%**	0.554	69.7%

Table 4: Efficiency and unlearning effectiveness on MLLMU-Bench using Qwen2.5-VL-3B-Instruct under the 5% forget setting. Unlearning time includes localization and model update, including pruning for MANU, but excludes model loading and evaluation.

Method	Trainable Parameters	Intervention Parameters	Unlearning Time (min)	Forget Acc. Reduction ↑	Retain Acc. Change ↑
VASE	**0.655M (0.017%)**	**0.655M (0.017%)**	181.0	**+44.8 pp**	**+0.5 pp**
MANU	–	1.614M (0.043%)	5.0	+4.0 pp	-2.4 pp
MMUnlearner	14.967M (0.397%)	0.903M (0.024%)	151.7	+13.6 pp	-0.7 pp

Unlearning efficiency analysis. Table 4 compares parameter footprint, runtime, and unlearning effectiveness. With $K = 256$, VASE optimizes only 0.655M parameters (0.017% of the model), while MMUnlearner optimizes 22.8× as many. Its intervention footprint is 59.4% smaller than MANU and 27.4% smaller than MMUnlearner. VASE reduces Forget accuracy by 44.8 pp while improving Retain accuracy by 0.5 pp; MANU and MMUnlearner achieve smaller Forget reductions of 4.0 and 13.6 pp and decrease Retain accuracy by 2.4 and 0.7 pp, respectively. VASE requires 181.0 minutes, compared with 151.7 minutes for MMUnlearner, while MANU finishes in 5.0 minutes through direct pruning. These results show that VASE provides a stronger forgetting–retention trade-off with a compact intervention footprint.

5 Conclusion

In this paper, we introduce Visual Activation-Guided Sparse Editing (VASE) for multimodal machine unlearning. VASE locates forget-related visual FFN channels through activation analysis and selectively edits them under forgetting and retention constraints. Extensive experiments across multiple benchmarks and MLLM backbones demonstrate that VASE achieves effective forgetting while preserving retained knowledge and general utility. Our analyses also confirm the value of activation-guided localization and sparse parameter editing. Future work will extend VASE to sequential unlearning requests and broader multimodal architectures.

AI USE STATEMENT

In this work, we used generative AI tools for none of the tasks requiring disclosure. We have not used generative AI tools for the generation or fabrication of experimental data or results, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for checking grammar and typographical errors. We have reviewed all AI-assisted work. All suggested corrections were manually verified by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report, 2025. URL https://arxiv.org/abs/2502.13923.

Chengyi Cai, Zesheng Ye, Peike Li, Bo Han, Jianzhong Qi, and Feng Liu. Visual-guided key-token regularization for multimodal large language model unlearning. *arXiv preprint arXiv:2601.22020*, 2026. URL https://arxiv.org/abs/2601.22020.

Alexey Dontsov, Dmitrii Korzh, Alexey Zhavoronkin, Boris Mikheev, Denis Bobkov, Aibek Alanov, Oleg Rogov, Ivan Oseledets, and Elena Tutubalina. CLEAR: Character unlearning in textual and visual modalities. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20582–20603, 2025. doi: 10.18653/v1/2025.findings-acl.1058. URL https://aclanthology.org/2025.findings-acl.1058/.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0058. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/02443e4e008231e0af0855f3cc70ed17-Abstract-Conference.html.

Akshat Gupta, Dev Sajnani, and Gopala Anumanchipalli. A unified framework for model editing. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15403–15418. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.903. URL https://aclanthology.org/2024.findings-emnlp.903/.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Han Huang, Haitian Zhong, Tao Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. VLKEB: A large vision-language model knowledge editing benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pp. 9257–9280, 2024. doi: 10.52202/079017-0294. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/1198b53fa686831d5f0c0860d7ec4f34-Abstract-Datasets_and_Benchmarks_Track.html.

Jiahao Huo, Yibo Yan, Xu Zheng, Yuanhuiyi Lyu, Xin Zou, Zhihua Wei, and Xuming Hu. MMUnlearner: Reformulating multimodal machine unlearning in the era of multimodal large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7190–7206, 2025. doi: 10.18653/v1/2025.findings-acl.375. URL https://aclanthology.org/2025.findings-acl.375/.

Yejin Kim, Dongjun Hwang, Sungmin Cha, and Junsuk Choe. Knowledge vector weakening: Efficient training-free unlearning for large vision-language models. *arXiv preprint arXiv:2601.21794*, 2026. URL https://arxiv.org/abs/2601.21794.

Kunhao Li, Wenhao Li, Di Wu, Lei Yang, Jun Bai, Ju Jia, and Jason Xue. Cross-modal unlearning via influential neuron path editing in multimodal large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(42):35589–35597, 2026. doi: 10.1609/aaai.v40i42.40870. URL https://ojs.aaai.org/index.php/AAAI/article/view/40870.

Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents*, pp. 243–254. PMLR, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

Zheyuan Liu, Guangyao Dou, Mengzhao Jia, Zhaoxuan Tan, Qingkai Zeng, Yongle Yuan, and Meng Jiang. Protecting privacy in multimodal large language models with MLLMU-bench. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4105–4135, 2025a. doi: 10.18653/v1/2025.naacl-long.207. URL https://aclanthology.org/2025.naacl-long.207/.

Zheyuan Liu, Guangyao Dou, Xiangchi Yuan, Chunhui Zhang, Zhaoxuan Tan, and Meng Jiang. Modality-aware neuron pruning for unlearning in multimodal large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5913–5933, 2025b. doi: 10.18653/v1/2025.acl-long.295. URL https://aclanthology.org/2025.acl-long.295/.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. *arXiv preprint arXiv:2401.06121*, 2024. URL https://arxiv.org/abs/2401.06121.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *International Conference on Learning Representations*, 2022. URL https://mlanthology.org/iclr/2022/mitchell2022iclr-fast/.

Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Variational bayesian unlearning. *Advances in Neural Information Processing Systems*, 33:16025–16036, 2020.

Vaidehi Patil, Yi-Lin Sung, Peter Hase, Jie Peng, Tianlong Chen, and Mohit Bansal. Unlearning sensitive information in multimodal LLMs: Benchmark and attack-defense evaluation, 2025. URL https://arxiv.org/abs/2505.01456.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4556f5398bd2c61bd7500e306b4e560a-Abstract-Conference.html.

Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling SGD: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pp. 303–319. IEEE, 2022.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025a. URL https://arxiv.org/abs/2508.18265.

Yue Wang, Qizhou Wang, Feng Liu, Wei Huang, Yali Du, Xiaojiang Du, and Bo Han. GRU: Mitigating the trade-off between unlearning and retention for LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 64690–64710. PMLR, 2025b. URL https://proceedings.mlr.press/v267/wang25de.html.

Yuhang Wang, Zhenxing Niu, Haoxuan Ji, Guangyu He, Linlin Zhang, and Haichang Gao. Null space constrained contrastive visual forgetting for MLLM unlearning, 2026. URL https://arxiv.org/abs/2605.05909.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. *arXiv preprint arXiv:2310.10683*, 2023. URL https://arxiv.org/abs/2310.10683.

Zhen Zeng, Leijiang Gu, Zhangling Duan, Feng Li, Cees G. M. Snoek, Meng Wang, and Zenglin Shi. Towards benign memory forgetting for selective multimodal large language model unlearning, 2025. URL https://arxiv.org/abs/2511.20196.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024. URL https://arxiv.org/abs/2404.05868.

A Additional Experimental Details

Dataset statistics. MLLMU-Bench (Liu et al., 2025a) contains 500 fictitious profiles and 153 public celebrity profiles, each associated with portrait images and questions probing both visual and textual knowledge. CLEAR (Dontsov et al., 2025) extends TOFU (Maini et al., 2024) with portrait images for 200 synthetic author profiles, comprising 3,770 visual question–answer pairs and 4,000 text-only question–answer pairs. For both benchmarks, we use the official Forget, Retain, and real-world evaluation splits with a 5% forget ratio.

Model preparation. We evaluate Qwen2.5-VL-3B (Bai et al., 2025) and InternVL3.5-8B (Wang et al., 2025a), which cover the standard and gated visual feed-forward architectures considered by our method. Following the benchmark protocols, each backbone is first trained on the full training set using LoRA to obtain a vanilla checkpoint that has acquired the target profiles. All unlearning methods start from the same vanilla checkpoint for a given backbone and dataset.

Baseline methods. Gradient Ascent (GA) (Thudi et al., 2022) maximizes the training loss on the Forget set. Gradient Difference (GA Diff) (Liu et al., 2022) combines gradient ascent on Forget samples with supervised training on Retain samples. KL Minimization (KL Min) (Nguyen et al., 2020) suppresses Forget samples while regularizing the model toward its original outputs on the Retain set. Negative Preference Optimization (NPO) (Zhang et al., 2024) treats Forget samples as dispreferred responses under a preference-learning objective. MANU (Liu et al., 2025b) identifies forget-related multimodal neurons and removes them through structured pruning. MMUnlearner (Huo et al., 2025) applies saliency-guided updates to suppress target visual knowledge while protecting non-target parameters.

Evaluation protocols. We evaluate four complementary aspects of post-unlearning behavior. For *classification VQA*, we report multiple-choice accuracy on the Forget and Retain sets; lower Forget accuracy indicates stronger forgetting, whereas higher Retain accuracy indicates better preservation. For *generative VQA*, the model answers open-ended visual questions without candidate-answer cues, and its responses are evaluated using ROUGE-L. MLLMU-Bench uses image–text generation questions, whereas CLEAR asks the model to identify a person from an image. Lower Forget ROUGE-L and higher Retain ROUGE-L are preferred.

We additionally evaluate *text-only QA* to determine whether editing the vision encoder affects textual knowledge. MLLMU-Bench reports QA accuracy, while CLEAR reports QA ROUGE-L; higher values indicate better preservation. Finally, general utility is measured using VQA and QA accuracy on the real-world celebrity split of MLLMU-Bench and VQA accuracy on the real-world and real-face splits of CLEAR.

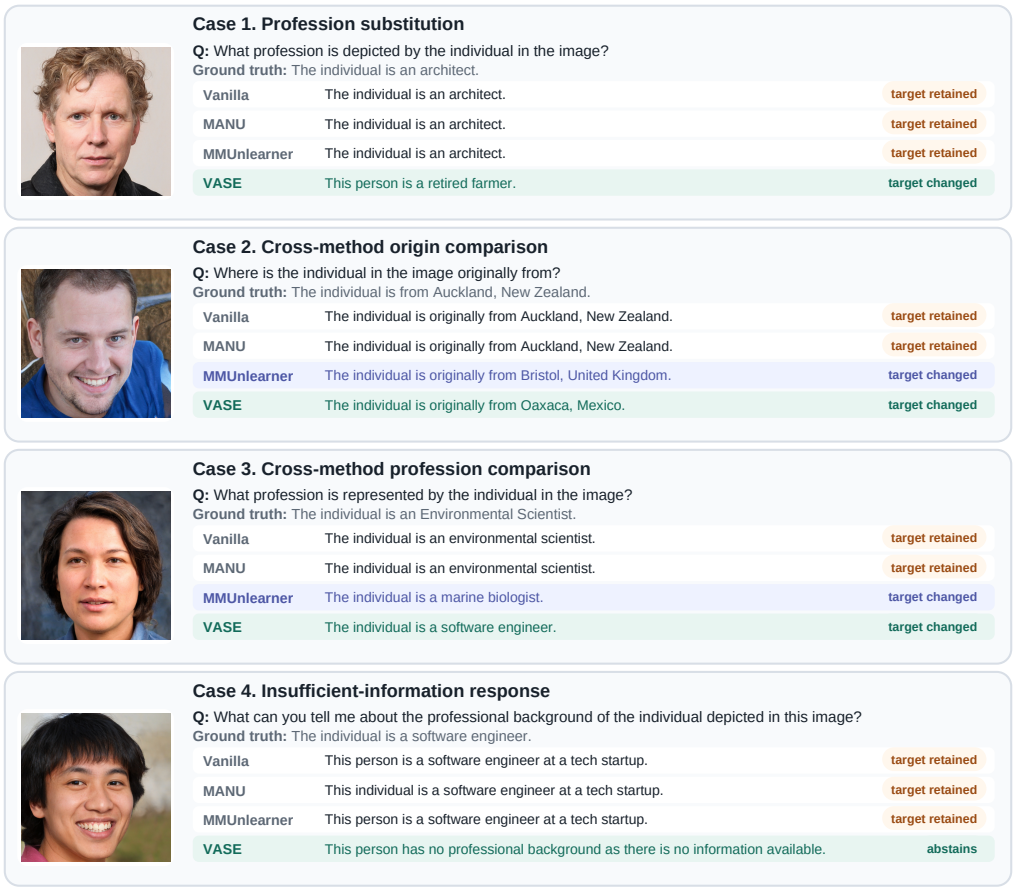


Figure 5: Qualitative comparison on the MLLMU-Bench Forget set using Qwen2.5-VL-3B. MANU often retains the target association, whereas MMUnlearner changes it only in some cases. VASE consistently suppresses the original target information through factual substitution or an insufficient-information response.

Training and evaluation details. All methods use identical data splits, model initialization, evaluation prompts, image preprocessing, and decoding settings. For both Qwen2.5-VL-3B and InternVL3.5-8B, the initial LoRA adapters are trained for four epochs with a learning rate of 3×10^{-5}. In the subsequent sparse editing stage, we use a learning rate of 3×10^{-4}. Qwen2.5-VL-3B and InternVL3.5-8B are both optimized for four editing epochs on both benchmarks. We adopt the released baseline configurations whenever available. Our method freezes the backbone and optimizes only the sparse residuals attached to selected visual channels; these residuals are merged into the model before evaluation. Open-ended evaluation uses deterministic decoding for reproducible comparisons.

B Qualitative Analysis

As shown in Figure 5, Vanilla and MANU continue to recover the target facts in all four examples. MMUnlearner retains the target information in Cases 1 and 4, but changes the predicted origin and profession in Cases 2 and 3. In contrast, VASE disrupts the original image-conditioned association in every example. Its behavior includes substitutions of professions and locations, as well as an explicit insufficient-information response in Case 4. These examples complement the aggregate results by showing that VASE suppresses target visual knowledge at the response level, although forgetting may manifest as either an alternative prediction or abstention.

Table 5: Results across different forget ratios on MLLMU-Bench using Qwen2.5-VL-3B. The editing budget K is scaled with the forget ratio. Before and After denote the vanilla and unlearned models, respectively. Acc and R-L denote accuracy and ROUGE-L.

Forget Ratio	K	**Stage**	F-VQA Acc ↓	F-VQA R-L ↓	R-VQA Acc ↑	R-VQA R-L ↑	RW-VQA Acc ↑
5%	256	Before	62.4%	0.631	53.2%	**0.555**	**69.7%**
		After	**17.6%**	**0.369**	**53.7%**	0.554	**69.7%**
10%	512	Before	56.0%	0.570	54.0%	**0.556**	**69.7%**
		After	**21.2%**	**0.374**	**55.9%**	0.555	64.2%
15%	768	Before	54.1%	0.536	**53.6%**	**0.563**	**69.7%**
		After	**21.9%**	**0.393**	52.6%	0.537	67.9%

Table 6: Before-and-after results of VASE on additional backbones on MLLMU-Bench with a 5% forget ratio. F, R, and RW denote the Forget, Retain, and Real-World splits.

Stage	**Qwen2.5-VL-7B**					**LLaVA-1.5-7B**				
	F-VQA Acc ↓	F-VQA R-L ↓	R-VQA Acc ↑	R-VQA R-L ↑	RW-VQA Acc ↑	F-VQA Acc ↓	F-VQA R-L ↓	R-VQA Acc ↑	R-VQA R-L ↑	RW-VQA Acc ↑
Before Unlearning	69.6%	0.595	**65.7%**	0.542	**76.4%**	45.6%	0.411	**43.5%**	**0.359**	**48.5%**
After Unlearning	**36.8%**	**0.499**	65.2%	**0.543**	76.0%	**37.6%**	**0.251**	42.8%	0.353	48.3%

C Effect of the forget ratio

We evaluate VASE with forget ratios of 5%, 10%, and 15%, scaling the editing budget K proportionally. As shown in Table 5, VASE consistently reduces Forget VQA accuracy and ROUGE-L across all settings. The post-unlearning Forget accuracy remains between 17.6% and 21.9%, despite the growing forget set. Retain accuracy changes by at most 1.9 percentage points, while Retain ROUGE-L remains nearly unchanged at 5% and 10%. Real-world utility is preserved at 5% and 15%, with a larger fluctuation at the 10% ratio. These results show that VASE remains effective as the forget ratio increases, while utility can vary with the editing budget.

D Experiments on Additional Backbones

To examine whether VASE generalizes beyond the backbones used in the main experiments, we further evaluate it on Qwen2.5-VL-7B and LLaVA-1.5-7B using MLLMU-Bench with a 5% forget ratio. As shown in Table 6, VASE reduces Forget VQA accuracy by 32.8 percentage points on Qwen2.5-VL-7B and by 8.0 points on LLaVA-1.5-7B. Forget ROUGE-L also decreases from 0.595 to 0.499 and from 0.411 to 0.251, respectively, demonstrating consistent forgetting under both classification and open-ended evaluation.

Meanwhile, the changes in Retain accuracy are limited to 0.5 and 0.7 percentage points, and real-world accuracy decreases by only 0.4 and 0.2 points. Retain ROUGE-L also remains nearly unchanged on both models. These results show that VASE transfers across model families and scales, achieving targeted visual forgetting while largely preserving non-target and general-purpose capabilities.