EDGEEDIT: AN EDGE-ORIENTED MULTIMODAL KNOWLEDGE EDITING FRAMEWORK FOR ON-DEVICE MLLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

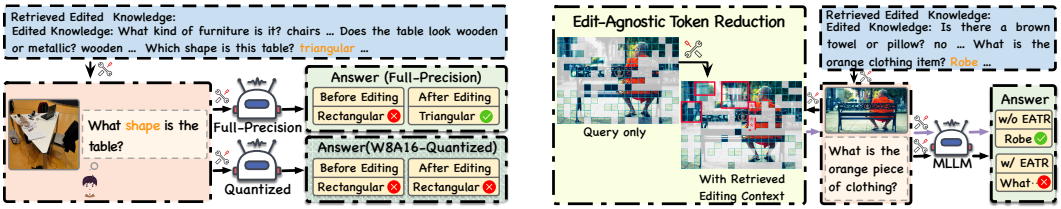
On-device multimodal large language models (MLLMs) are increasingly enabling multimodal applications, yet updating their static encoded knowledge efficiently remains a critical challenge. In-context editing (ICE) provides an efficient and lightweight approach for fine-grained knowledge updates. However, on-device deployment typically demands joint model and input compression, introducing severe obstacles for ICE: model quantization impairs the model's ability to utilize contextualized edited knowledge, while edit-agnostic token reduction inadvertently discards edit-critical visual evidence. To address these challenges, we propose `EdgeEdit`, an edge-oriented multimodal knowledge editing framework that coordinates offline model deployment with online inference. Specifically, `EdgeEdit` introduces Edit-Aware Mixed-Precision Quantization for offline model deployment, which conducts zeroth-order sensitivity profiling and budget-constrained dynamic programming to preserve contextualized edited knowledge utilization. For online inference, `EdgeEdit` employs Evidence-Saliency Guided Token Reduction, combining cross-modal saliency attribution with semantic relevance to progressively remove redundant visual tokens while retaining edit-critical visual evidence. Extensive experiments demonstrate that `EdgeEdit` maintains competitive editing performance while substantially reducing system costs. Compared with the strongest competing baseline for each efficiency metric, `EdgeEdit` achieves average reduction factors of $2.9\times$/$1.5\times$/$2.5\times$ in memory usage, editing latency, and energy consumption on the server, and $2.0\times$/$2.2\times$/$2.0\times$ on mobile devices, respectively.

1 INTRODUCTION

On-device multimodal large language models (MLLMs) are increasingly enabling multimodal applications (Lu et al., 2025a;b). Existing studies have primarily focused on enabling efficient on-device deployment of MLLMs through model compression (Feng et al., 2025; Cai et al., 2025) and hardware-aware optimization (Lu et al., 2025a;b), while generally treating the knowledge encoded in deployed models as *static*. This leaves a substantial gap between the static knowledge of deployed models and users' *dynamic and personalized* needs (Du et al., 2025; Ma et al., 2025). As Figure 3 shows, static deployed MLLMs may fail to keep pace with evolving user-specific knowledge, limiting their ability to provide personalized responses.

To make on-device MLLMs keep up with users' demands, existing studies have explored cloud-device collaborative continual adaptation (Wang et al., 2024b) and reinforcement learning fine-tuning (Wang et al., 2025) to process updated knowledge. However, both approaches rely on centralized model optimization for task or domain level adaptation, thereby requiring continuous task-specific data collection and substantial computation. These issues essentially contradict the efficient and lightweight features of on-device deployment. Therefore, this paper takes the first step toward multimodal knowledge editing (MKE) (Cheng et al., 2023) for on-device MLLMs, where the updates of MLLMs are performed on provided samples, without demanding requirements on data scale or computational resources.

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) Quantization-Induced Failure of ICE (b) Edit-Critical Evidence Loss under EATR

Figure 1: In-context editing degradation under conventional compression. w/o EATR and w/ EATR denote without and with Edit-Agnostic Token Reduction, respectively. ⋆ and □ denote removed edit-critical tokens and retained edit-irrelevant tokens under edit-agnostic token reduction, respectively.

Existing MKE approaches broadly fall into two paradigms: (1) knowledge-internalizing methods encode edited knowledge into parameters of the designed models (Pan et al., 2024; Chen et al., 2025a; Zhang et al., 2026), and (2) knowledge-externalizing methods externally store edited knowledge to support retrieval during inference (Ma et al., 2025; Hao et al., 2025; Zhang et al., 2025a). The former is less suitable for on-device MLLMs, as updating model parameters introduces additional optimization overhead and often requires backpropagation, which is poorly supported by inference-oriented mobile NPUs (Lu et al., 2026). Within the knowledge-externalizing paradigm, *in-context editing* (ICE) directly integrates retrieved edited knowledge into the inference context to generate responses in a forward-only manner. Thus, we focus on efficient ICE for resource-constrained on-device MLLMs.

Notably, on-device deployment typically requires compression of both the model and input (Qu et al., 2025; Lu et al., 2025a), which creates non-trivial obstacles for ICE. Specifically, (1) The ability to exploit contextualized edited knowledge might be impaired. On-device NPUs are particularly optimized for low-bit integer computation, making quantization a widely adopted technique (Lin et al., 2024; Qu et al., 2025). When the retrieved edited knowledge conflicts with the knowledge encoded within the model, low-bit quantization makes the model prefer to conservatively select the original knowledge, vividly shown in Figure 1(a). (2) Edit-critical visual evidence tends to be erroneously discarded during inference. Multimodal input compression generally relies on edit-agnostic token reduction, which removes visual tokens according to their estimated importance (Yuan et al., 2026). However, the retrieved context usually contains redundant and irrelevant contextual knowledge, and the reduction strategy often incorrectly treats them as valuable. This leads to filtering out edit-critical evidence and preserving the noise, as illustrated in Figure 1(b).

These compression-induced obstacles disrupt not only MLLMs' fundamental ability of ICE but also the execution of ICE itself. In response, we first propose an edge-oriented multimodal knowledge editing framework, namely `EdgeEdit`, for resource-constrained on-device MLLMs. `EdgeEdit` is designed to seamlessly integrate ICE into MLLMs throughout both the model maintenance and inference stages.

Our Solution. `EdgeEdit` enables efficient ICE through an offline–online co-compression design comprising two complementary modules: (1) Edit-Aware Mixed-Precision Quantization module for offline model compression, which identifies layer-wise sensitivity in contextualized edited knowledge utilization and allocates precision accordingly; and (2) Evidence-Saliency Guided Token Reduction module for online multimodal input compression, which dynamically reassesses visual-token importance under the current query and editing context to retain edit-critical evidence while progressively removing redundant tokens. Specifically, to preserve contextualized edited knowledge utilization under model quantization, the edit-aware mixed-precision quantization module performs zeroth-order, context-conditioned sensitivity profiling to quantify the effect of different precisions across transformer layers, and employs budget-constrained dynamic programming to derive the mixed-precision configuration. To preserve edit-critical visual evidence during input compression, the evidence-saliency guided token reduction module combines cross-modal evidence saliency with semantic relevance to identify important visual tokens, and progressively reduces redundant tokens through stage-wise token selection and merging.

Contributions. Our contributions are threefold: (1) To the best of our knowledge, we are the first to investigate efficient multimodal knowledge editing for resource-constrained on-device MLLMs and identify two key compression-induced challenges for ICE; (2) We propose `EdgeEdit`, an edge-

oriented multimodal knowledge editing framework for on-device MLLMs comprising Edit-Aware Mixed-Precision Quantization and Evidence-Saliency Guided Token Reduction; (3) Extensive experiments demonstrate that `EdgeEdit` maintains competitive editing effectiveness while reducing memory usage, editing latency, and energy consumption by average factors of $2.9\times/1.5\times/2.5\times$ on the server and $2.0\times/2.2\times/2.0\times$ on mobile devices, respectively.

2 Background: ICE for Multimodal Large Language Models

2.1 Multimodal In-Context Editing

Given an MLLM f_θ, let $q_e = (i_e, x_e)$ denote an edit-related multimodal query, where i_e and x_e are the visual and textual inputs, respectively. MKE aims to update specific knowledge (*Reliability*), generalize the update to semantically equivalent queries (*Generality*), and preserve behavior on unrelated knowledge (*Locality*); see Appendix A for formal definitions. In this work, we focus on multimodal ICE, a knowledge-externalizing paradigm that incorporates edited knowledge into the inference context without modifying model parameters. Let $\mathcal{D}_e = \{(i_e^{(t)}, x_e^{(t)}, y_e^{(t)})\}_{t=1}^{T}$ denote the editing set. Given q_e, ICE retrieves the n most relevant examples as the editing context $\mathcal{C}_n$ and predicts $\hat{y}_e = f_\theta(\mathcal{C}_n, i_e, x_e)$, where a successful edit satisfies $\hat{y}_e = y_e$ and y_e denotes the desired edited output. The forward-only editing paradigm of ICE is well suited to resource-constrained on-device MLLMs.

2.2 Challenges of On-Device Compression for In-Context Editing

After identifying the key obstacles to integrating ICE with on-device MLLMs, analyzing the underlying root causes becomes an urgent priority. Accordingly, we further conduct empirical studies from the perspectives of ICE in model capability and execution, as shown in Figure 2, to support the systematic identification of fundamental challenges.

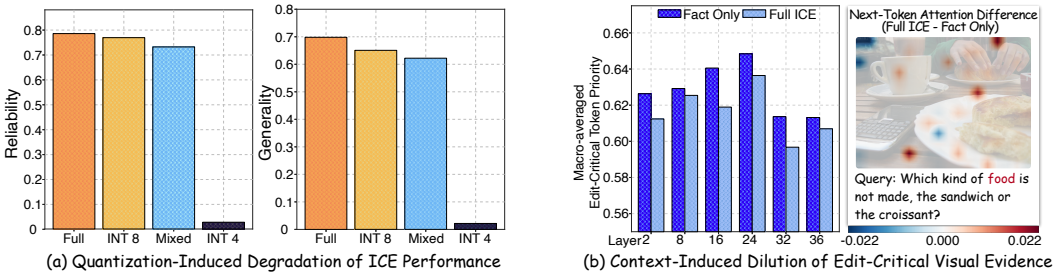


Figure 2: Edit-agnostic compression undermines in-context editing on Qwen2.5-VL-3B-Instruct with ComprehendEdit. See Appendix B for details.

Preserving Contextualized Edited Knowledge Utilization under Budget-Constrained Quantization. As shown in Figure 2(a), uniformly applying low-bit quantization substantially degrades ICE performance. Normally, typical mixed-precision quantization methods alleviate such degradation through layer-wise precision allocation, which is guided by sensitivity estimated from generic model outputs or task performance (Li et al., 2025c; Zhao et al., 2026). However, under this situation, the sensitivity fails to identify layers within MLLMs critical for contextualized edited knowledge, ultimately leading to a 6.5% average performance drop. This raises a key challenge: how to tailor the layer-wise mixed quantization sensitivity, especially for contextualized edited knowledge. Moreover, the quantization process is constrained by the resource budget. With L layers and K candidate precision levels, the assignment space grows as K^L, making exhaustive search impractical.

Preserving Edit-Critical Visual Evidence under Context-Induced Attention Shifts. Conventional token reduction methods typically rely on edit-agnostic token importance estimation (Chen et al., 2024; Alvar et al., 2025; Zhang et al., 2025c). Under ICE, however, retrieved editing contexts can alter visual-token importance and dilute the relative salience of edit-critical evidence. As shown in Figure 2(b), the relative attention priority of edit-critical visual tokens decreases across multiple

layers under the full editing context. The attention-difference map further shows that the full editing context shifts next-token attention toward non-critical visual regions, thereby diluting the relative attention assigned to edit-critical evidence. Consequently, edit-agnostic token-importance estimation may underestimate visual tokens that remain critical for applying the edited knowledge. The core challenge is therefore to reliably identify edit-critical visual evidence under context-induced attention shifts while reducing visual redundancy.

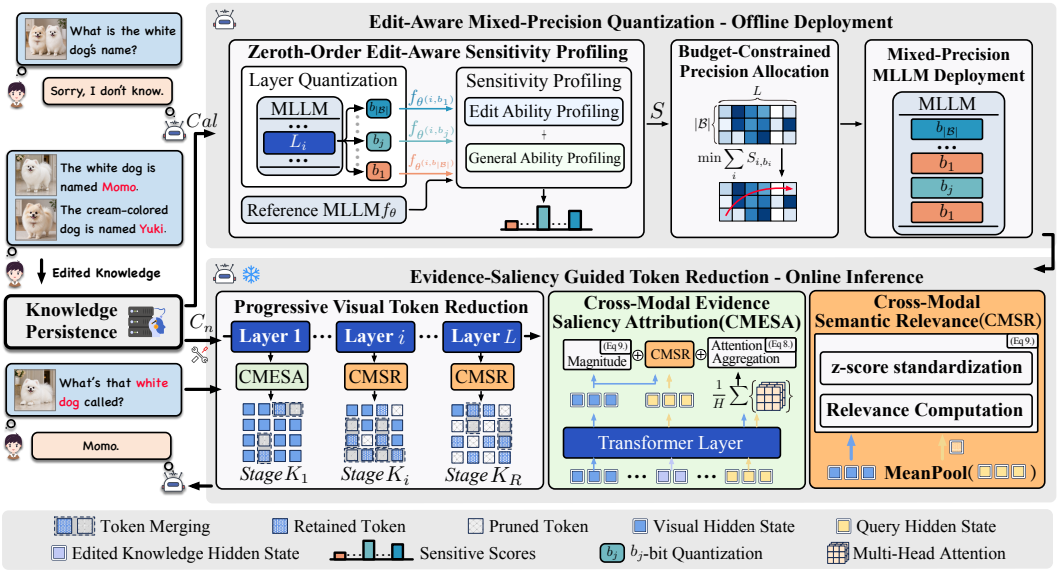


Figure 3: Overview of the EdgeEdit framework.

3 METHOD

3.1 EDGEEDIT OVERVIEW

Figure 3 provides an overview of EdgeEdit. Our focus is to jointly compress the model during offline deployment and multimodal inputs during online inference for efficient on-device ICE, as motivated in Section 2.2. Let Q denote the quantization configuration and π the input-reduction policy. Given the MKE evaluation dimensions $\mathcal{K} = \{\text{Rel}, \text{T-G}, \text{M-G}, \text{T-L}, \text{M-L}\}$ and an editing dataset $\mathcal{D}_e$, we define the compression-induced ICE degradation as

$$\Delta_{\text{ICE}}(Q, \pi) = \sum_{k \in \mathcal{K}} \lambda_k \left[\frac{M_k(f_\theta; \mathcal{D}_e) - M_k(f_{Q(\theta)} \circ \pi; \mathcal{D}_e)}{M_k(f_\theta; \mathcal{D}_e) + \epsilon} \right]_+, \tag{1}$$

where M_k denotes the corresponding metric and $[a]_+ = \max(a, 0)$.

Given model-internal and model-external budgets B_{int} and B_{ext}, we formulate the joint co-compression objective as

$$(Q^\star, \pi^\star) = \arg\min_{Q,\pi} \Delta_{\text{ICE}}(Q, \pi) \quad \text{s.t.} \quad C_{\text{int}}(Q) \le B_{\text{int}}, \quad C_{\text{ext}}(\pi) \le B_{\text{ext}}. \tag{2}$$

To make the joint optimization tractable and align with the offline–online deployment pipeline, we use a two-stage decomposition: we first optimize $Q^\star$ under B_{int} and then optimize $\pi^\star$ under B_{ext} with $Q^\star$ fixed. Accordingly, EdgeEdit comprises the Edit-Aware Mixed-Precision Quantization (EAMPQ) module for preserving contextualized edited knowledge utilization under budget-constrained quantization (§3.2) and the Evidence-Saliency Guided Token Reduction (ESGTR) module for preserving edit-critical evidence under constrained token budgets (§3.3).

3.2 EDIT-AWARE MIXED-PRECISION QUANTIZATION

Let $\mathbf{b} = (b_1, \ldots, b_L) \in \mathcal{B}^L$ denote a layer-wise precision configuration, where b_i is the precision assigned to the i-th Transformer layer and $\mathcal{B}$ is the candidate precision set. Since exhaustively

evaluating all $|\mathcal{B}|^L$ configurations is prohibitive, we introduce a layer-wise separable surrogate for the ICE degradation:

$$\widehat{\Delta}_{\mathrm{ICE}}(\mathbf{b}) = \sum_{i=1}^{L} S_{i,b_i}, \qquad S_{i,b_i} = \lambda_{\mathrm{edit}} S_{i,b_i}^{\mathrm{edit}} + \lambda_{\mathrm{pre}} S_{i,b_i}^{\mathrm{pre}}, \tag{3}$$

where $S_{i,b_i}^{\mathrm{edit}}$ measures degradation in edited knowledge utilization and S_{i,b_i}^{pre} measures changes in unrelated model behavior. EAMPQ estimates these scores through *Zeroth-Order Edit-Aware Sensitivity Profiling* and minimizes $\widehat{\Delta}_{\mathrm{ICE}}(\mathbf{b})$ through *Budget-Constrained Precision-Space Dynamic Programming*.

Zeroth-Order Edit-Aware Sensitivity Profiling. Layer-wise quantization sensitivity is commonly estimated from derivative-based signals (Ding et al., 2024; Li et al., 2025a) or forward-evaluated output discrepancies (Li et al., 2025b), which do not explicitly characterize the utilization of externally provided edited knowledge. We therefore introduce an edit-aware zeroth-order profiling method based on isolated-layer forward evaluations.

Given an editing dataset $\mathcal{D}_e$, we construct a stratified calibration set $\mathcal{D}_{\mathrm{cal}} = \{z_n\}_{n=1}^{N}$. Each z_n induces an ICE context $\mathcal{C}_{z_n}$, an edit-query set $\mathcal{Q}_{\mathrm{edit}}^{(n)}$ containing its reliability and generality queries, and a preservation-query set $\mathcal{Q}_{\mathrm{pre}}^{(n)}$ containing its locality queries. For each layer i and precision $b \in \mathcal{B}$, let $\theta^{(i,b)}$ denote the parameters obtained by quantizing only layer i to precision b. Under the same context $\mathcal{C}_{z_n}$, we evaluate both the reference model f_θ and the quantization-perturbed model $f_{\theta^{(i,b)}}$ on every query-target pair $(q, y) \in \mathcal{Q}_{\mathrm{edit}}^{(n)} \cup \mathcal{Q}_{\mathrm{pre}}^{(n)}$ and obtain the target-token logits $\mathbf{Z}_{n,q}^{\mathrm{ref}} = f_\theta(\mathcal{C}_{z_n}, q)$ and $\mathbf{Z}_{n,q}^{(i,b)} = f_{\theta^{(i,b)}}(\mathcal{C}_{z_n}, q)$, respectively.

For an edit query $q \in \mathcal{Q}_{\mathrm{edit}}^{(n)}$, we characterize the impact of quantization on edited knowledge utilization by the positive increase in target-token negative log-likelihood:

$$d_{\mathrm{edit}}^{(i,b)}(q, y; \mathcal{C}_{z_n}) = \left[\mathcal{L}_{\mathrm{NLL}}\left(\mathbf{Z}_{n,q}^{(i,b)}, y\right) - \mathcal{L}_{\mathrm{NLL}}\left(\mathbf{Z}_{n,q}^{\mathrm{ref}}, y\right)\right]_+, \tag{4}$$

where $[a]_+ = \max(a, 0)$, ensuring that only quantization-induced degradation contributes to the sensitivity score.

For a preservation query $q \in \mathcal{Q}_{\mathrm{pre}}^{(n)}$, target matching alone cannot fully capture changes in predictive behavior, so we instead measure the divergence between the reference and quantized output distributions. Let $\mathcal{I}_t^K = \mathrm{TopK}(\mathbf{Z}_{n,q}^{\mathrm{ref}}[t])$ denote the top-K reference-logit indices at position t. We define the divergence as follows:

$$d_{\mathrm{pre}}^{(i,b)}(q; \mathcal{C}_{z_n}) = \frac{1}{T_q} \sum_{t=1}^{T_q} D_{\mathrm{KL}}\left(\sigma\left(\mathbf{Z}_{n,q}^{\mathrm{ref}}[t, \mathcal{I}_t^K]\right) \,\middle\|\, \sigma\left(\mathbf{Z}_{n,q}^{(i,b)}[t, \mathcal{I}_t^K]\right)\right), \tag{5}$$

where T_q is the number of evaluated target-token positions. Restricting the divergence to the reference top-K support reduces profiling memory while retaining the dominant output modes. We average the corresponding discrepancies over $\mathcal{D}_{\mathrm{cal}}$ to obtain $S_{i,b}^{\mathrm{edit}}$ and $S_{i,b}^{\mathrm{pre}}$ in Equation 3.

Budget-Constrained Precision-Space Dynamic Programming. Given the profiled sensitivity scores $\{S_{i,b}\}$, we represent the model-side budget B_{int} using a prescribed precision composition $\mathbf{m} = (m_b)_{b \in \mathcal{B}}$, where m_b denotes the number of layers assigned to precision b and $\sum_{b \in \mathcal{B}} m_b = L$. The resulting constrained allocation problem is

$$\mathbf{b}^\star = \arg\min_{\mathbf{b} \in \mathcal{B}^L} \sum_{i=1}^{L} S_{i,b_i} \quad \text{s.t.} \quad \sum_{i=1}^{L} \mathbb{I}(b_i = b) = m_b, \quad \forall b \in \mathcal{B}. \tag{6}$$

We solve Equation 6 exactly using dynamic programming. Let $V_i(\mathbf{c})$ denote the minimum cumulative sensitivity after assigning precisions to the first i layers, where $\mathbf{c} = (c_b)_{b \in \mathcal{B}}$ records the current precision composition. The recurrence is

$$V_i(\mathbf{c}) = \min_{\substack{b \in \mathcal{B} \\ c_b > 0}} \left\{V_{i-1}(\mathbf{c} - \mathbf{e}_b) + S_{i,b}\right\}, \tag{7}$$

where $\mathbf{e}_b$ is the unit vector for precision b. With $V_0(\mathbf{0}) = 0$, backtracking from $V_L(\mathbf{m})$ yields the globally optimal layer-wise precision configuration under the prescribed composition.

3.3 EVIDENCE-SALIENCY GUIDED TOKEN REDUCTION

Visual tokens constitute a substantial fraction of MLLM inputs, making their efficient reduction important for lowering input-side computation. Under ICE, retrieved editing contexts alter cross-modal interactions and redistribute visual-token importance, making conventional token-importance estimates less reliable. ESGTR therefore progressively removes redundant visual tokens while preserving edit-critical evidence.

Cross-Modal Evidence Saliency Attribution. Given an ICE context, let $\mathbf{H}^{(\ell)} \in \mathbb{R}^{N_\ell \times d}$ denote the hidden states at layer ℓ, and $\mathcal{I}_{\mathrm{q}}^{(\ell)}$ and $\mathcal{I}_{\mathrm{v}}^{(\ell)}$ index the current edit-related query and visual tokens, respectively. Let $\mathcal{L}_{\mathrm{red}} = \{\ell_1, \ldots, \ell_R\}$ denote the ordered reduction layers. At the first stage, we extract the query-to-visual attention matrix at ℓ_1, identify visually grounded query tokens through sparse weighting, and aggregate their attention to estimate visual-token saliency:

$$\begin{aligned} \mathbf{A}^{(\ell_1)} &= \frac{1}{H}\sum_{h=1}^{H} \mathrm{Attn}_h^{(\ell_1)}\left[\mathcal{I}_{\mathrm{q}}^{(\ell_1)}, \mathcal{I}_{\mathrm{v}}^{(\ell_1)}\right], \\ \boldsymbol{\omega}^{(\ell_1)} &= \mathrm{Sparsemax}\left(\frac{\mathrm{Z}\left(\mathbf{A}^{(\ell_1)}\mathbf{1}\right)}{\tau}\right), \\ \mathbf{s}_{\mathrm{attn}}^{(\ell_1)} &= \left(\boldsymbol{\omega}^{(\ell_1)}\right)^{\top} \mathbf{A}^{(\ell_1)}. \end{aligned} \tag{8}$$

Here, H is the number of attention heads, $\mathbf{1} \in \mathbb{R}^{N_v}$ is an all-ones vector, and $A^{(\ell_1)} \in \mathbb{R}^{N_q \times N_v}$ is the query-to-visual attention matrix between N_q query tokens and N_v visual tokens. $\mathrm{Z}(\cdot)$ denotes z-score standardization, and τ is a temperature parameter. Sparsemax produces sparse query-token weights $\boldsymbol{\omega}^{(\ell_1)}$, which aggregate $\mathbf{A}^{(\ell_1)}$ into the visual-token saliency $\mathbf{s}_{\mathrm{attn}}^{(\ell_1)}$.

Since attention weights are normalized over the entire context, $\mathbf{s}_{\mathrm{attn}}^{(\ell_1)}$ only partially characterizes visual-token relevance to the current query. We therefore complement it with cross-modal semantic relevance, measured by $\cos(\mathbf{h}_j^{(\ell_1)}, \bar{\mathbf{h}}_{\mathrm{q}}^{(\ell_1)})$, and activation magnitude $\|\mathbf{h}_j^{(\ell_1)}\|_2$. The stage-wise visual saliency is defined as

$$s_j^{(r)} = \begin{cases} \alpha\ \mathrm{Z}\left(s_{\mathrm{attn},j}^{(\ell_1)}\right) + \beta\ \mathrm{Z}\left(\cos\left(\mathbf{h}_j^{(\ell_1)}, \bar{\mathbf{h}}_{\mathrm{q}}^{(\ell_1)}\right)\right) + \gamma\ \mathrm{Z}\left(\left\|\mathbf{h}_j^{(\ell_1)}\right\|_2\right), & r = 1, \\ \mathrm{Z}\left(\cos\left(\mathbf{h}_j^{(\ell_r)}, \bar{\mathbf{h}}_{\mathrm{q}}^{(\ell_r)}\right)\right), & r > 1, \end{cases} \tag{9}$$

where $\mathbf{h}_j^{(\ell_r)}$ is the hidden state of visual token j, and $\bar{\mathbf{h}}_{\mathrm{q}}^{(\ell_r)}$ is the mean-pooled query representation at layer ℓ_r. The nonnegative coefficients α, β, and γ weight the three saliency signals.

Progressive Visual Token Reduction. At stage r, we set visual-token budget $K_r = \lceil \rho_r N_{\mathrm{v}}^{(0)} \rceil$, where $\rho_1 > \cdots > \rho_R$ are prescribed retention ratios and $N_{\mathrm{v}}^{(0)}$ is the initial number of visual tokens. To avoid over-concentrating retained tokens in a limited region, ESGTR reserves a small uniformly sampled anchor set $\mathcal{A}^{(r)} \subseteq \mathcal{I}_{\mathrm{v}}^{(\ell_r)}$ and allocates the remaining budget according to saliency:

$$\mathcal{K}^{(r)} = \mathcal{A}^{(r)} \cup \mathrm{TopK}_{K_r - |\mathcal{A}^{(r)}|}\left(\left\{ s_j^{(r)} \,\middle|\, j \in \mathcal{I}_{\mathrm{v}}^{(\ell_r)} \setminus \mathcal{A}^{(r)} \right\}\right), \tag{10}$$

where TopK_k returns the indices of the k largest scores, and $\mathcal{K}^{(r)}$ is the retained visual-token set.

Rather than discarding unselected tokens, ESGTR merges each into its most similar retained token. Let $\mathcal{U}^{(r)} = \mathcal{I}_{\mathrm{v}}^{(\ell_r)} \setminus \mathcal{K}^{(r)}$. The assignment and merging operations are defined as

$$\begin{aligned} \phi_r(j) &= \arg\max_{k \in \mathcal{K}^{(r)}} \cos\left(\mathbf{h}_j^{(\ell_r)}, \mathbf{h}_k^{(\ell_r)}\right), \qquad j \in \mathcal{U}^{(r)}, \\ \widetilde{\mathbf{h}}_k^{(\ell_r)} &= \frac{\mathbf{h}_k^{(\ell_r)} + \sum_{j \in \mathcal{U}^{(r)}: \phi_r(j) = k} \mathbf{h}_j^{(\ell_r)}}{1 + \left|\left\{ j \in \mathcal{U}^{(r)} : \phi_r(j) = k \right\}\right|}, \qquad k \in \mathcal{K}^{(r)}. \end{aligned} \tag{11}$$

After merging, only the K_r retained tokens are propagated to subsequent layers.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Baselines. We compare EdgeEdit with four representative MKE baselines: the knowledge-externalizing methods IKE (Cheng et al., 2023) and HICE (Ma et al., 2025), and the knowledge-internalizing methods VisEdit (Chen et al., 2025b) and MSCKE (Zeng et al., 2025). IKE and HICE externalize edited knowledge via retrieved demonstrations, while VisEdit and MSCKE internalize edited knowledge through a visual intervention module and a counterfactual model, respectively.

Datasets and models. We evaluate EdgeEdit on two widely used MKE benchmarks: ComprehendEdit (Ma et al., 2025) and E-VQA (Cheng et al., 2023), covering diverse visual reasoning tasks and VQA with counterfactual answers, respectively. We report Reliability (Rel.), textual/multimodal Generality (T-G./M-G.), and textual/multimodal Locality (T-L./M-L.). We use Qwen2.5-VL-3B-Instruct (Bai et al., 2025) as the primary model and MobileVLM-V2-3B (Chu et al., 2024) to evaluate generalizability across model architectures and deployment settings.

Implementation details. We evaluate EdgeEdit on the server and COTS mobile devices listed in Table 1. On mobile devices, methods run on ARM64 CPUs using an ARM64-compatible PyTorch runtime within a chroot Linux userspace on rooted Android devices, with Accelerate-based disk offloading for limited memory and EdgeEdit leveraging the Qualcomm Hexagon NPU for supported operations. In Figure 4, system efficiency metrics are independently min–max normalized and reverse-scaled to [10, 100]. Due to costly on-device evaluation, mobile system costs are measured on a stratified sample of 100 ComprehendEdit instances.

Table 1: Devices used in experiments.

Type	Device	SoC/CPU	Accelerator	Memory
Server	Server	Intel Xeon Gold 6336Y	NVIDIA A800-80GB	80GB HBM2e
Mobile	Redmi K60 Pro	Snapdragon 8 Gen 2	Hexagon NPU V73	16GB LPDDR5
Mobile	Redmi K70 Pro	Snapdragon 8 Gen 3	Hexagon NPU V75	16GB LPDDR5
Mobile	Realme GT8 Pro	Snapdragon 8 Elite	Hexagon NPU V81	16GB LPDDR5X

4.2 MAIN EXPERIMENTAL RESULTS

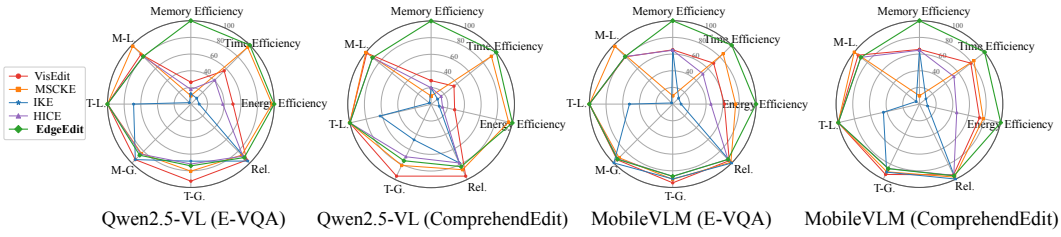


Figure 4: Editing quality and server-side resource-efficiency trade-offs of EdgeEdit and competing multimodal knowledge editing methods on E-VQA and ComprehendEdit.

Overall results. Figure 4 compares editing performance and server-side system efficiency across E-VQA and ComprehendEdit with Qwen2.5-VL-3B-Instruct and MobileVLM-V2-3B, while Figure 5 reports mobile system costs on ComprehendEdit across three COTS mobile devices. Overall, **EdgeEdit offers a favorable trade-off between editing performance and resource efficiency across the evaluated settings.** We summarize the main observations as follows:

EdgeEdit achieves the best system efficiency across server and mobile settings. Across both backbones and datasets, EdgeEdit consistently achieves the highest time, memory, and energy efficiency on the server, requiring only 0.094–0.163 seconds per edit, 3.50–6.02 GB of peak memory, and 9.81–21.81 J of energy. Compared with the best competing baseline for each system metric, EdgeEdit is 1.12×–1.91× faster, 2.38×–3.32× more memory-efficient, and 1.21×–4.43× more energy-efficient across the four server-side settings. These gains further translate to mobile deployment, where EdgeEdit requires only 6.47–7.35 GB of memory, 27.00–43.07 s of editing time, and

23.18–39.16 J of energy across three COTS mobile devices, while reducing memory usage, editing latency, and energy consumption by averages of $2.04\times$, $2.22\times$, and $1.97\times$, respectively.



Figure 5: System costs on ComprehendEdit across three COTS mobile devices, with `EdgeEdit` leveraging NPU acceleration and baseline knowledge editing methods running on CPUs. Memory denotes the total memory required under disk offloading and is therefore reported once per method.

`EdgeEdit` achieves competitive and well-balanced editing performance. As shown in Figure 4, `EdgeEdit` achieves strong reliability and generality across both datasets and backbones. While some baselines achieve higher individual scores, VisEdit and MSCKE require costly offline training (Table 2), whereas IKE suffers from poor locality. Overall, `EdgeEdit` preserves the ability of the model to utilize contextualized edited knowledge while maintaining effective ICE under aggressive compression.

Table 2: Training costs on ComprehendEdit with Qwen2.5-VL-3B-Instruct.

Method	Memory (GiB)	Time (h)
VisEdit	40.13	12.27
MSCKE	22.17	2.64

`EdgeEdit` preserves editing locality despite aggressive compression. Compared with its uncompressed HICE counterpart, `EdgeEdit` maintains nearly unchanged locality across both backbones and datasets. These results indicate that `EdgeEdit` preserves contextualized edited knowledge utilization while introducing minimal interference with unrelated knowledge.

4.3 Ablation Studies

In this section, we evaluate the contributions of EAMPQ and ESGTR. We summarize the main observations as follows.

Table 3: Ablation study of EAMPQ and ESGTR on E-VQA and ComprehendEdit using Qwen2.5-VL-3B-Instruct.

Component		E-VQA					ComprehendEdit			
EAMPQ	ESGTR	Rel.	T-G.	M-G.	T-L.	M-L.	Rel.	T-G.	T-L.	M-L.
–	–	88.29	71.86	84.85	100.00	80.80	78.09	69.81	99.89	89.92
✓	–	89.87	73.15	85.71	100.00	80.80	80.57	71.31	99.89	89.92
–	✓	90.25	74.15	86.77	100.00	80.80	80.41	73.38	99.89	89.92
✓	✓	**90.68**	**74.27**	**86.77**	**100.00**	**80.80**	**82.95**	**75.12**	**99.89**	**89.92**

Both EAMPQ and ESGTR contribute to editing performance. As shown in Table 3, both components consistently improve editing performance across E-VQA and ComprehendEdit, with their combination achieving the best results. Specifically, `EdgeEdit` improves Rel./T-G./M-G. from 88.29/71.86/84.85 to 90.68/74.27/86.77 on E-VQA, and Rel./T-G. from 78.09/69.81 to 82.95/75.12 on ComprehendEdit. Locality remains nearly unchanged across all variants, demonstrating complementary gains from EAMPQ and ESGTR without sacrificing editing locality.

EAMPQ achieves stronger contextualized edited knowledge utilization than generic mixed-precision quantization. From Tables 3 and 4, EAMPQ consistently improves performance on ComprehendEdit. Under the same mixed-precision budget, it outperforms Generic Mixed-Precision by 6.18 and 3.46 points in Rel. and T-G., respectively, while surpassing the unquantized setting. These results validate the importance of edit-aware layer-wise sensitivity profiling for preserving contextualized edited knowledge utilization.

Table 4: ICE performance of Qwen2.5-VL-3B-Instruct on ComprehendEdit under different precision settings. w/o Quan. denotes without quantization; GMP denotes Generic Mixed-Precision (Appendix B.1).

Method	Rel.	T-G.
w/o Quan.	78.09	69.81
GMP	74.39	67.85
EAMPQ (Ours)	**80.57**	**71.31**

ESGTR outperforms edit-agnostic token reduction in preserving edit-critical visual evidence. From Table 5, ESGTR outperforms EATR by 0.58/0.44 points in Rel./T-G. on ComprehendEdit. Both strategies outperform the variant without token reduction, indicating that removing visual redundancy itself can benefit ICE. These results highlight the importance of preserving edit-critical visual evidence for effective token reduction under ICE.

Table 5: Visual-token reduction on ComprehendEdit with Qwen2.5-VL-3B-Instruct. w/o TR denotes without token reduction; EATR denotes edit-agnostic token reduction (Appendix B.3).

Method	Retention Ratio (%)	Rel.	T-G.
w/o TR	100	78.09	69.81
EATR	50	79.83	72.94
ESGTR (Ours)	50	**80.41**	**73.38**

5 Related Work

Multimodal Knowledge Editing. Existing MKE methods broadly follow two paradigms. Knowledge-internalizing methods encode edited knowledge into model parameters, neurons, or lightweight trainable components (Pan et al., 2024; Chen et al., 2025a; Zhang et al., 2026; Hu et al., 2026), whereas knowledge-externalizing methods retain edited knowledge externally and retrieve it to condition model outputs (Ma et al., 2025; Hao et al., 2025; Zhang et al., 2025a). Despite their effectiveness, these methods are typically designed and evaluated without considering the stringent resource constraints of on-device MLLMs. Motivated by this gap, we investigate efficient multimodal knowledge editing for resource-constrained on-device MLLMs, aiming to maintain editing effectiveness under aggressive compression.

Efficient Multimodal Large Models. Multimodal large language models have demonstrated strong cross-modal understanding and reasoning capabilities (Dong et al., 2025; Sun et al., 2025). To democratize MLLMs, recent studies have facilitated their deployment on resource-constrained devices through compact model design (Lu et al., 2025a;b), quantization (Li et al., 2025b; Wang et al., 2024a), and token-efficient inference (Zhang et al., 2025b;c; Alvar et al., 2025). To accommodate evolving knowledge after deployment, existing studies have explored parameter-efficient fine-tuning (Bi et al., 2025), continual learning (Guo et al., 2025), and cloud-device collaborative adaptation (Wang et al., 2024b). However, these approaches primarily target broader task or domain level adaptation and often rely on iterative model optimization, limiting their suitability for rapid and fine-grained knowledge updates. Accordingly, we extend efficient on-device MLLMs to support rapid and fine-grained knowledge updates through multimodal knowledge editing.

6 Conclusion

We present `EdgeEdit`, the first edge-oriented framework for efficient multimodal knowledge editing on resource-constrained on-device MLLMs. It coordinates Edit-Aware Mixed-Precision Quantization during offline deployment with Evidence-Saliency Guided Token Reduction during online inference, preserving contextualized edited knowledge utilization and edit-critical visual evidence under compression. Extensive experiments demonstrate that `EdgeEdit` maintains competitive editing performance while substantially reducing system costs, validating its effectiveness for resource-constrained on-device MLLMs.

AI USE STATEMENT

In this work, generative AI tools were used solely for language editing, including improvements to grammar, clarity, readability, and presentation. All AI-assisted revisions were carefully reviewed by the authors to ensure that they preserved the intended technical meaning and did not introduce unsupported claims, factual inaccuracies, or other substantive changes.

ETHICS STATEMENT

This work uses publicly available benchmark datasets and pretrained models. We do not collect new human-subject data or interact with participants. Some benchmark images may contain personal or sensitive information. These data are used solely for aggregate model evaluation, and we do not attempt to identify individuals or recover personal information. Any qualitative examples included in the paper or appendix are selected and, where necessary, redacted to avoid exposing direct identifiers.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the proposed method, model configurations, datasets, evaluation metrics, and experimental settings in the main paper and appendix. Upon acceptance, we plan to release the implementation and experiment scripts necessary to reproduce the reported results.

REFERENCES

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 9392–9401, June 2025.

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.

Jinhe Bi, Yujun Wang, Haokun Chen, Xun Xiao, Artur Hecker, Volker Tresp, and Yunpu Ma. LLaVA steering: Visual instruction tuning with 500x fewer parameters through modality linear representation-steering. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15230–15250, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.739. URL https://aclanthology.org/2025.acl-long.739/.

Yuxuan Cai, Jiangning Zhang, Haoyang He, Xinwei He, Ao Tong, Zhenye Gan, Chengjie Wang, Zhucun Xue, Yong Liu, and Xiang Bai. Llava-kd: A framework of distilling multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 239–249, October 2025.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models, 2024.

Qizhou Chen, Chengyu Wang, Dakan Wang, Taolin Zhang, Wangyue Li, and Xiaofeng He. Lifelong knowledge editing for vision language models with low-rank mixture-of-experts. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 9455–9466, June 2025a.

Qizhou Chen, Taolin Zhang, Chengyu Wang, Xiaofeng He, Dakan Wang, and Tingting Liu. Attribution analysis meets model editing: Advancing knowledge correction in vision language models with visedit. volume 39, pp. 2168–2176, 2025b.

Siyuan Cheng, Bozhong Tian, Qingbin Liu, Xi Chen, Yongheng Wang, Huajun Chen, and Ningyu Zhang. Can we edit multimodal large language models? In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13877–13888, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.856. URL https://aclanthology.org/2023.emnlp-main.856/.

Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming Hu, Xinyang Lin, Bo Zhang, and Chunhua Shen. Mobilevlm v2: Faster and stronger baseline for vision language model, 2024. URL https://arxiv.org/abs/2402.03766.

Li Ding, Wen Fei, Yuyang Huang, Shuangrui Ding, Wenrui Dai, Chenglin Li, Junni Zou, and Hongkai Xiong. AMPA: Adaptive mixed precision allocation for low-bit integer training. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10960–10977. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/ding24b.html.

Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 9062–9072, June 2025.

Yuntao Du, Kailin Jiang, Zhi Gao, Chenrui Shi, Zilong Zheng, Siyuan Qi, and Qing Li. MMKE-bench: A multimodal editing benchmark for diverse visual knowledge. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=v8qABSeeKO.

Mark Endo, Xiaohan Wang, and Serena Yeung-Levy. Feather the throttle: Revisiting visual token pruning for vision-language model acceleration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22826–22835, October 2025.

Qianhan Feng, Wenshuo Li, Tong Lin, and Xinghao Chen. Align-kd: Distilling cross-modal alignment knowledge for mobile vision-language large model enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4178–4188, June 2025.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. In *International Conference on Learning Representations*, 2023.

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 Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13572–13586, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.666. URL https://aclanthology.org/2025.acl-long.666/.

Haoran Hao, Jiaming Han, Changsheng Li, Yu-Feng Li, and Xiangyu Yue. Rap: Retrieval-augmented personalization for multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 14538–14548, June 2025.

Jiayu Hu, Beibei Li, Jiangwei Xia, Yanjun Qin, Bing Ji, and Zhongshi He. Look closer! an adversarial parametric editing framework for hallucination mitigation in vlms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(26):21840–21848, Mar. 2026. doi: 10.1609/aaai.v40i26.39336. URL https://ojs.aaai.org/index.php/AAAI/article/view/39336.

Nan Li, Yonghui Su, and Lianbo Ma. Efficient and generalizable mixed-precision quantization via topological entropy. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=9Hweh7Wx8R.

Shiyao Li, Yingchun Hu, Xuefei Ning, Xihui Liu, Ke Hong, Xiaotao Jia, Xiuhong Li, Yaqi Yan, Pei Ran, Guohao Dai, Shengen Yan, Huazhong Yang, and Yu Wang. Mbq: Modality-balanced quantization for large vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 4167–4177, June 2025b.

Xing Li, Zeyu XING, Yiming Li, Linping Qu, Hui-Ling Zhen, Yiwu Yao, Wulong Liu, Sinno Jialin Pan, and Mingxuan Yuan. KVTuner: Sensitivity-aware layer-wise mixed-precision KV cache quantization for efficient and nearly lossless LLM inference. In *Forty-second International Conference on Machine Learning*, 2025c. URL https://openreview.net/forum?id=zDwipF6h06.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. In P. Gibbons, G. Pekhimenko, and C. De Sa (eds.), *Proceedings of Machine Learning and Systems*, volume 6, pp. 87–100, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/file/42a452cbafa9dd64e9ba4aa95cc1ef21-Paper-Conference.pdf.

Xudong Lu, Yinghao Chen, Cheng Chen, Hui Tan, Boheng Chen, Yina Xie, Rui Hu, Guanxin Tan, Renshou Wu, Yan Hu, Yi Zeng, Lei Wu, Liuyang Bian, Zhaoxiong Wang, Long Liu, Yanzhou Yang, Han Xiao, Aojun Zhou, Yafei Wen, Xiaoxin Chen, Shuai Ren, and Hongsheng Li. Bluelm-v-3b: Algorithm and system co-design for multimodal large language models on mobile devices. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 4145–4155, June 2025a.

Xudong Lu, Yinghao Chen, Renshou Wu, Haohao Gao, Xi Chen, Xue Yang, Xiangyu Zhao, Aojun Zhou, Fangyuan Li, Yafei Wen, Xiaoxin Chen, Shuai Ren, and Hongsheng Li. Genieblue: Integrating both linguistic and multimodal capabilities for large language models on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4198–4210, October 2025b.

Zhenyan Lu, Daliang Xu, Dongqi Cai, Zexi Li, Wei Liu, Jian Luan, Fangming Liu, Shangguang Wang, and Mengwei Xu. Mobiedit: Resource-efficient knowledge editing for personalized on-device LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=fb7yTBOV3p.

Yaohui Ma, Xiaopeng Hong, Shizhou Zhang, Huiyun Li, Zhilin Zhu, Wei Luo, and Zhiheng Ma. Comprehendedit: A comprehensive dataset and evaluation framework for multimodal knowledge editing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(18):19323–19331, 2025.

Haowen Pan, Yixin Cao, Xiaozhi Wang, Xun Yang, and Meng Wang. Finding and editing multimodal neurons in pre-trained transformers. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics*, pp. 1012–1037, Bangkok, Thailand, 2024. Association for Computational Linguistics.

Guanqiao Qu, Qiyuan Chen, Wei Wei, Zheng Lin, Xianhao Chen, and Kaibin Huang. Mobile edge intelligence for large language models: A contemporary survey. *IEEE Communications Surveys Tutorials*, 27(6):3820–3860, 2025. doi: 10.1109/COMST.2025.3527641.

Linzhuang Sun, Hao Liang, Jingxuan Wei, Bihui Yu, Tianpeng Li, Fan Yang, Zenan Zhou, and Wentao Zhang. MM-verify: Enhancing multimodal reasoning with chain-of-thought verification. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14100–14115, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.689. URL https://aclanthology.org/2025.acl-long.689/.

Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Q-VLM: Post-training quantization for large vision-language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=gxMfNArldP.

Guanqun Wang, Jiaming Liu, Chenxuan Li, Yuan Zhang, Junpeng Ma, Xinyu Wei, Kevin Zhang, Maurice Chong, Renrui Zhang, Yijiang Liu, and Shanghang Zhang. Cloud-device collaborative learning for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12646–12655, June 2024b.

Taiyi Wang, Zhihao Wu, Jianheng Liu, Jianye HAO, Jun Wang, and Kun Shao. DistRL: An asynchronous distributed reinforcement learning framework for on-device control agent. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LPG8pPSfQD.

Hanzhang Yuan, Mengxuan Hu, Wenhao Zhang, Tianlong Wang, Zhongliang Zhou, Jiasen Lu, and Sheng Li. Reducing token redundancy in LVLMs: A systematic review of token pruning methods. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7231–7251, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.328. URL https://aclanthology.org/2026.acl-long.328/.

Zhen Zeng, Leijiang Gu, Xun Yang, Zhangling Duan, Zenglin Shi, and Meng Wang. Visual-oriented fine-grained knowledge editing for multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2491–2500, October 2025.

Qiang Zhang, Fanrui Zhang, Jiawei Liu, Ming Hu, Junjun He, and Zheng-Jun Zha. Reliable lifelong multimodal editing: Conflict-aware retrieval meets multi-level guidance. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=hdJXzKZjY9.

Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-mini: Efficient image and video large multimodal models with one vision token. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=UQJ7CDW8nb.

Tianyu Zhang, Junfeng Fang, Houcheng Jiang, Xingyu Zhu, Xiang Wang, and Xiangnan He. Lpedit: Locality-preserving knowledge editing for multimodal large language models. In *Proceedings of the ACM Web Conference 2026*, WWW '26, pp. 8425–8428, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400723070. doi: 10.1145/3774904.3792876. URL https://doi.org/10.1145/3774904.3792876.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In *International Conference on Machine Learning*, 2025c.

Junchen Zhao, Ali Derakhshan, Jayden Hyman, Junhao Dong, Sangeetha Abdu Jyothi, and Ian Harris. CoopQ: Cooperative game inspired layerwise mixed precision quantization for LLMs. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 7566–7578, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.373. URL https://aclanthology.org/2026.findings-acl.373/.

A Task Formulation

Multimodal knowledge editing aims to update a specific piece of knowledge encoded or utilized by an MLLM while preserving its behavior on unrelated knowledge. Formally, we denote an MLLM parameterized by θ as $\mathcal{M}_\theta$. An editing dataset is defined as $\mathcal{D}_e = \left\{\left(i_e^{(t)}, x_e^{(t)}, y_e^{(t)}\right)\right\}_{t=1}^{T}$, where i_e and x_e denote the visual and textual inputs associated with the edited knowledge, respectively, and y_e denotes the target output. For each editing sample $(i_e^{(t)}, x_e^{(t)}, y_e^{(t)}) \in \mathcal{D}_e$, the original model produces $y_o^{(t)} = \mathcal{M}_\theta(i_e^{(t)}, x_e^{(t)})$. Multimodal knowledge editing aims to obtain an edited model $\mathcal{M}_{\theta_e}$

such that $\mathcal{M}_{\theta_e}(i_e^{(t)}, x_e^{(t)}) = y_e^{(t)}$, where $y_e^{(t)} \neq y_o^{(t)}$. Following standard multimodal knowledge editing evaluation protocols, an effective editing method is expected to satisfy three complementary properties: **reliability**, **locality**, and **generality**.

Reliability. Reliability measures whether the edited model successfully produces the target output for the editing instance. It is defined as

$$\mathrm{Rel} = \mathbb{E}_{(i_e, x_e, y_e)\sim\mathcal{D}_e}\left[\mathbb{I}\left(\mathcal{M}_{\theta_e}(i_e, x_e) = y_e\right)\right], \tag{12}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function.

Locality. Locality measures whether the predictions on inputs unrelated to the edited knowledge remain unchanged after editing. Locality is evaluated from two perspectives: textual locality and multimodal locality. Let $\mathcal{D}_{tl}$ and $\mathcal{D}_{vl}$ denote the text-only and multimodal locality evaluation sets, respectively. Textual locality (T-L) and multimodal locality (M-L) are formulated as

$$\text{T-L} = \mathbb{E}_{(x_{tl}, y_{tl})\sim\mathcal{D}_{tl}}\left[\mathbb{I}\left(\mathcal{M}_{\theta_e}(\emptyset, x_{tl}) = \mathcal{M}_{\theta}(\emptyset, x_{tl})\right)\right], \tag{13}$$

and

$$\text{M-L} = \mathbb{E}_{(i_{vl}, x_{vl}, y_{vl})\sim\mathcal{D}_{vl}}\left[\mathbb{I}\left(\mathcal{M}_{\theta_e}(i_{vl}, x_{vl}) = \mathcal{M}_{\theta}(i_{vl}, x_{vl})\right)\right], \tag{14}$$

where $\emptyset$ denotes the absence of visual input.

Generality. Generality measures whether the editing effect can generalize from the original editing instance to semantically equivalent inputs. Generality is evaluated from two perspectives: textual generality and multimodal generality. Let $\mathcal{P}_x(x_e)$ and $\mathcal{P}_i(i_e)$ denote the sets of semantically equivalent textual and visual variants of x_e and i_e, respectively. Textual generality (T-G) and multimodal generality (M-G) are defined as

$$\text{T-G} = \mathbb{E}_{\substack{(i_e, x_e, y_e)\sim\mathcal{D}_e \\ x_{re}\sim\mathcal{P}_x(x_e)}}\left[\mathbb{I}\left(\mathcal{M}_{\theta_e}(i_e, x_{re}) = y_e\right)\right], \tag{15}$$

and

$$\text{M-G} = \mathbb{E}_{\substack{(i_e, x_e, y_e)\sim\mathcal{D}_e \\ i_{re}\sim\mathcal{P}_i(i_e)}}\left[\mathbb{I}\left(\mathcal{M}_{\theta_e}(i_{re}, x_e) = y_e\right)\right]. \tag{16}$$

These metrics characterize the three fundamental objectives of MKE: accurately incorporating the target knowledge, preserving behavior on unrelated knowledge, and generalizing the editing effect across equivalent expressions.

B Additional Experimental Details

B.1 Quantization-Induced Degradation of ICE Performance

To investigate the impact of model quantization on in-context editing, we evaluate four quantization strategies for Qwen2.5-VL-3B-Instruct on the ComprehendEdit benchmark, as shown in Figure 2(a). Specifically, Full Precision (Full) serves as the unquantized baseline. Uniform INT8 (INT 8) employs the BitsAndBytes LLM.int8 backend with `load_in_8bit=True` and an outlier threshold of 6.0. Uniform INT4 (INT 4) applies uniform weight-only INT4 quantization. Generic Mixed-Precision (Mixed) follows the same layer-wise profiling and precision-allocation paradigm as `EdgeEdit`, but replaces the edit-aware profiling objective with a general VQA-oriented quantization loss. This loss combines target-output NLL increase, top-K KL divergence, and prediction-flip penalty, while allocating FP16, INT8, and INT4 precision under an average 8-bit weight budget. We implement in-context editing using HICE (Ma et al., 2025) and keep the HICE editing protocol fixed across all quantization configurations. We measure ICE performance using two complementary metrics: reliability, which evaluates whether the model correctly applies the edited knowledge, and generality, which evaluates whether the edited knowledge generalizes to rephrased queries.

B.2 Context-Induced Dilution of Edit-Critical Visual Evidence

To investigate whether textual editing demonstrations dilute the visual evidence required for applying edited knowledge, we analyze the attention assigned to edit-critical visual tokens under two in-context editing conditions. We use Qwen2.5-VL-3B-Instruct on the ComprehendEdit benchmark and focus on examples for which the model correctly answers the edited query under the Fact-Only

condition. This criterion focuses the analysis on cases where the edited knowledge is successfully applied without additional demonstrations, enabling a direct comparison of the attention assigned to edit-critical visual evidence before and after adding textual editing demonstrations. We compare two input conditions. In the Full ICE condition, we follow the standard HICE (Ma et al., 2025) protocol, which includes the retrieved editing demonstrations and the target multimodal query. In the Fact-Only condition, we use an ablated HICE input that contains only the edited knowledge and the target multimodal query. We use these two conditions to quantify how the presence of retrieved editing demonstrations changes the model's attention to visual evidence that is critical for applying the edited knowledge. To ensure a fair comparison, we first identify a shared set of edit-critical visual groups and then measure their attention priority under both conditions.

Edit-critical visual group identification. Let $\mathbf{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ denote the visual-token representations of an image, and let $\mathbf{p}$ denote the model's next-token probability distribution at the last prompt position under the Fact-Only condition. We divide the visual tokens, following their flattened spatial order, into non-overlapping groups of at most four consecutive tokens:

$$\mathcal{G} = \{G_1, \ldots, G_M\}, \qquad G_m = \{\mathbf{v}_{4m-3}, \ldots, \mathbf{v}_{\min(4m,N)}\}.$$

For each visual group G_m, we compute a mean visual representation from the tokens outside the group:

$$\bar{\mathbf{v}}_{-m} = \frac{1}{N - |G_m|} \sum_{j:\mathbf{v}_j \notin G_m} \mathbf{v}_j.$$

We construct a perturbed input by replacing the tokens in G_m with this group-excluded mean representation:

$$\tilde{\mathbf{v}}_i^{(m)} = \begin{cases} \bar{\mathbf{v}}_{-m}, & \mathbf{v}_i \in G_m, \\ \mathbf{v}_i, & \mathbf{v}_i \notin G_m. \end{cases}$$

This replacement suppresses the group-specific visual information while preserving the number and positions of visual tokens. Moreover, because $\bar{\mathbf{v}}_{-m}$ excludes all tokens in G_m, the replacement representation does not retain information contributed by the perturbed group itself.

Let $\mathbf{p}^{(m)}$ denote the next-token probability distribution produced from the perturbed visual input. We measure the importance of group G_m using the Jensen–Shannon divergence between the original and perturbed distributions:

$$s_m = \mathrm{JS}\Big(\mathbf{p} \,\Big\|\, \mathbf{p}^{(m)}\Big),$$

where

$$\mathrm{JS}(\mathbf{p} \,\|\, \mathbf{q}) = \frac{1}{2}\,\mathrm{KL}(\mathbf{p} \,\|\, \mathbf{r}) + \frac{1}{2}\,\mathrm{KL}(\mathbf{q} \,\|\, \mathbf{r}), \qquad \mathbf{r} = \frac{\mathbf{p} + \mathbf{q}}{2}.$$

A larger s_m indicates a greater change in the next-token distribution after suppressing the group-specific information in G_m, suggesting that G_m provides more visual evidence relevant to next-token prediction. We define the edit-critical token set as the union of the top 10% most influential visual groups:

$$\mathcal{C} = \bigcup_{G_m \in \mathrm{TopK}_{G_m \in \mathcal{G}}(s_m; K)} \{i : \mathbf{v}_i \in G_m\}, \qquad K = \max(1, \lceil 0.1M \rceil).$$

We identify $\mathcal{C}$ under Fact-Only and keep it fixed when evaluating Fact-Only and Full ICE.

Edit-critical token priority. Let $c \in \{\mathrm{F}, \mathrm{I}\}$ index the Fact-Only and Full ICE conditions, respectively. Let $A_{l,h}^{(c)}[t, i]$ denote the attention weight from the last prompt position t to visual token i at Transformer layer l and attention head h. We first average the attention weights over all H attention heads:

$$a_{l,i}^{(c)} = \frac{1}{H} \sum_{h=1}^{H} A_{l,h}^{(c)}[t, i].$$

We then normalize the attention weights over the visual-token domain:

$$\tilde{a}_{l,i}^{(c)} = \frac{a_{l,i}^{(c)}}{\sum_{j=1}^{N} a_{l,j}^{(c)}}.$$

For each condition, we rank all visual tokens in descending order according to $\tilde{a}_{l,i}^{(c)}$. Let $\mathrm{rank}_{l,c}(i) \in \{0, \ldots, N-1\}$ denote the zero-based rank of visual token i, where rank 0 corresponds to the largest attention weight. We define its normalized attention-rank percentile as

$$r_{l,i}^{(c)} = 1 - \frac{\mathrm{rank}_{l,c}(i)}{\max(1, N-1)}.$$

For each example d, the edit-critical token priority is computed as the average rank percentile of its fixed edit-critical tokens:

$$P_{l,d}^{(c)} = \frac{1}{|\mathcal{C}_d|} \sum_{i \in \mathcal{C}_d} r_{l,i}^{(c)}.$$

A larger $P_{l,d}^{(c)}$ indicates that the model assigns higher attention priority to edit-critical visual tokens than to other visual tokens.

Finally, we report the macro-average over all selected examples:

$$\bar{P}_l^{(c)} = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} P_{l,d}^{(c)},$$

where $\mathcal{D}$ contains the examples for which Fact-Only correctly predicts the edited target. We evaluate layers $l \in \{2, 8, 16, 24, 32, 36\}$ and compare the attention priority between the two conditions using

$$\Delta_l = \bar{P}_l^{(\mathrm{I})} - \bar{P}_l^{(\mathrm{F})}.$$

A negative Δ_l indicates that Full ICE reduces the relative attention-rank priority of visual tokens that are relevant to the next-token prediction, providing evidence of context-induced dilution.

B.3 Edit-Agnostic Token Reduction

Following the attention-based token pruning criterion studied in FEATHER (Endo et al., 2025), we construct an edit-agnostic baseline that ranks visual tokens according to the attention they receive from the last prompt token. Specifically, the importance of visual token i at layer ℓ is computed by averaging its attention weights across all heads:

$$s_i^{(\ell)} = \frac{1}{H} \sum_{h=1}^{H} A_{h,t,i}^{(\ell)}, \tag{17}$$

where t denotes the last prompt position before answer generation. Tokens with the highest scores are retained. All other settings, including the reduction layers, retention schedule, spatial anchors, and token-merging operation, are identical to ESGTR. Unlike ESGTR, this baseline does not employ an edit-specific importance estimator.

C Further Analysis

Table 6: Comparison of quantization strategies on Qwen2.5-VL-3B-Instruct with E-VQA.

Method	Rel.	T-G.	M-G.	T-L.	M-L.
Uniform RTN-W8A16	88.20	71.43	84.09	100.00	80.77
Uniform RTN-W4A16	2.58	2.53	2.63	100.00	80.80
Uniform GPTQ-W8A16	88.06	71.67	84.09	100.00	80.79
Uniform GPTQ-W4A16	87.72	69.66	85.69	100.00	80.80
Ours (EAMPQ only)	**89.87**	**73.15**	**85.71**	**100.00**	**80.80**

Comparison of quantization strategies. We compare EAMPQ with two representative post-training quantization baselines in Table 6: calibration-free round-to-nearest (RTN) quantization and GPTQ (Frantar et al., 2023), a widely used calibration-aware quantization method. RTN applies

Table 7: Comparison between random and edit-aware precision assignment on ComprehendEdit under the same mixed-precision budget. Random results are reported as mean and standard deviation over three independent assignments. Avg. denotes the mean of Rel. and T-G.

Precision Assignment	Rel.	T-G.	Avg.	T-L.	M-L.
Random	75.63 ± 4.34	66.67 ± 4.13	71.15 ± 4.22	99.53 ± 0.23	88.93 ± 0.41
Edit-aware (`EdgeEdit`)	**80.57**	**71.31**	**75.94**	**99.89**	**89.92**

uniform per-channel quantization without using calibration data, whereas GPTQ exploits calibration activations and approximate second-order information to reduce quantization error. These baselines allow us to assess whether the gains of EAMPQ arise merely from using a more sophisticated quantizer, or from its edit-aware layer-wise precision allocation. Direct uniform quantization is particularly harmful at very low precision: Uniform RTN-W4A16 reduces Rel., T-G., and M-G. to 2.58, 2.53, and 2.63, respectively. Although GPTQ substantially alleviates this degradation, Uniform GPTQ-W8A16 and GPTQ-W4A16 still achieve lower Rel. and T-G. than EAMPQ. In contrast, EAMPQ achieves 89.87 Rel., 73.15 T-G., and 85.71 M-G., while maintaining 100.00 T-L. and 80.80 M-L. These results validate the **necessity of edit-aware layer-wise sensitivity profiling and precision allocation, rather than applying a uniform precision configuration optimized only for generic model behavior**.

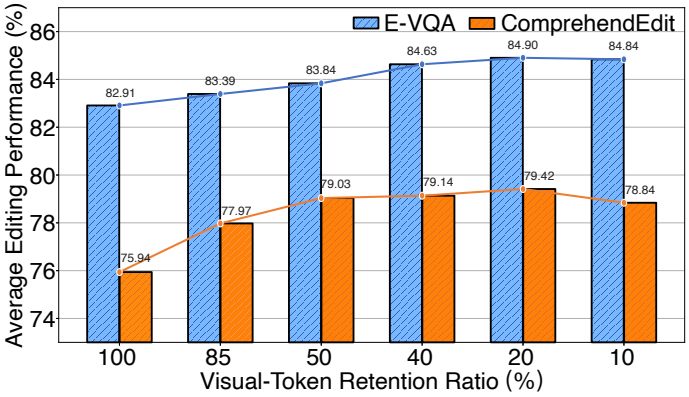


Figure 6: Effect of visual-token retention ratio on average editing performance on E-VQA and ComprehendEdit.

Effect of visual-token budget. Figure 6 evaluates the sensitivity of ESGTR to the visual-token retention ratio. Reducing the retention ratio from 100% to 20% improves the average editing performance from 82.91 to 84.90 on E-VQA and from 75.94 to 79.42 on ComprehendEdit. Performance drops only slightly at 10% retention, to 84.84 and 78.84, respectively, suggesting that ESGTR effectively removes visual redundancy while preserving edit-critical evidence until compression becomes overly aggressive. Notably, 20% retention achieves the highest editing performance on both datasets, while the performance gains become increasingly marginal below 50% retention.

Random Precision Assignment vs. Edit-Aware Mixed-Precision Quantization. To isolate the contribution of edit-aware mixed-precision quantization, we construct three random baselines by independently assigning precision to Transformer layers while maintaining the same mixed-precision budget as `EdgeEdit`. All other experimental settings, including the quantization backend, calibration data, and evaluation protocol, remain unchanged. We report the mean and standard deviation across the three random assignments in Table 7.

The edit-aware assignment improves Rel., T-G., and their average compared with random assignment. The relatively large standard deviations of the random baseline further indicate that editing performance is sensitive to how precision is assigned across layers. Moreover, `EdgeEdit` preserves nearly perfect T-L. and achieves higher M-L. than the random assignment baseline while consistently delivering stronger editing performance. These results demonstrate that the gains do

not arise merely from the mixed-precision budget, but from assigning precision according to editing sensitivity.