# BEYOND RECONSTRUCTION LOSS IN POST-TRAINING QUANTIZATION: BALANCED FITTING FOR LARGE VISION-LANGUAGE MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Post-training quantization (PTQ) enables efficient deployment of large vision-language models (LVLMs), but is typically calibrated on a small set while expected to generalize across diverse downstream tasks. Although recent PTQ methods for LVLMs incorporate sensitivity signals, they still minimize reconstruction loss with respect to the full-precision model, potentially over-preserving FP behavior and calibration-specific bias. Rather than treating quantization solely as an error to be minimized, we observe that it can also provide beneficial regularization for certain layers and modalities. Motivated by this observation, we propose Balanced Fitting, a quantization effect-based framework that balances precision and regularization beyond reconstruction-based optimization. By measuring layer- and component-wise quantization effects for weights, vision activations, and text activations, Balanced Fitting combines fine-grained fitting for sensitive components with coarser fitting to exploit potential regularization benefits. Experiments on multiple LVLMs show that our method consistently outperforms prior PTQ approaches under both weight-only and weight-activation quantization, while lower reconstruction loss does not reliably translate into better downstream performance.

## 1 INTRODUCTION

Large Vision-Language Models (LVLMs) have achieved strong multimodal reasoning performance across diverse tasks (Liu et al., 2023a), but at the cost of high computational and memory overhead. This has motivated research on model compression, including token pruning (Rao et al., 2021; Bolya et al., 2022), knowledge distillation (Hinton et al., 2015; Li et al., 2023b), and quantization (Jacob et al., 2018; Zhou et al., 2016). Among these, post-training quantization (PTQ) is particularly practical, as it reduces memory and computation using low-precision representations and enables efficient compression with only a small calibration set (Nagel et al., 2020).

While PTQ has shown strong performance in large language models (LLMs) (Frantar et al., 2022; Lin et al., 2024; Xiao et al., 2023), extending it to LVLMs remains challenging due to heterogeneous multimodal representations. Recent LVLM-specific methods incorporate sensitivity signals, including cross-layer dependencies (Q-VLM (Wang et al., 2024a)), modality-wise gradients (MBQ (Li et al., 2025)), and token-level importance (QIG (Xiang et al., 2026)). Despite these advances, they share a common objective of optimizing quantization by minimizing reconstruction error with respect to the full-precision (FP) model on a small calibration set.

We argue that this paradigm is fundamentally limited. First, prior work (Chen et al., 2021; Liang et al., 2021; Kuzmin et al., 2023; Wang et al., 2020) has reported that compression can sometimes improve performance, suggesting that the full-precision (FP) model may contain overfitted or suboptimal components. This suggests that quantization is not always a harmful perturbation; in some layers or components, it may act as an implicit regularizer. We further observe similar phenomena in LVLMs, where different components (weights, vision tokens, and text tokens) exhibit varying sensitivity across layers, and some benefit from quantization (Fig. 2(a)). Second, the calibration set is inherently limited and may not reflect the full downstream distribution, introducing bias. Conse-



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

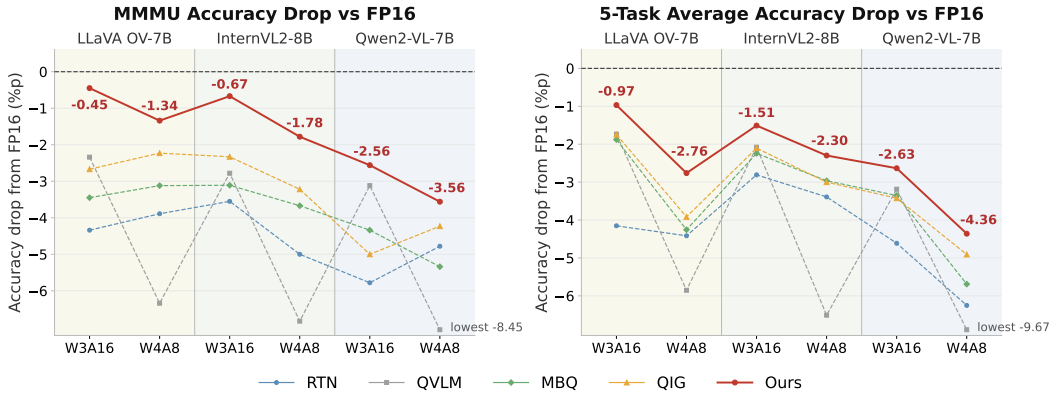



Figure 1: Accuracy drop from the full-precision (FP16) model across multiple LVLMs under low-bit settings (W3A16 and W4A8). Results are reported on the MMMU benchmark (left) and the average across five multimodal benchmarks (right). Our method consistently achieves the lowest accuracy drop across all models and configurations among state-of-the-art PTQ methods for LVLMs.

quently, minimizing reconstruction error with respect to the FP model on a limited calibration set can preserve overfitted FP behavior and calibration-set bias, leading to suboptimal generalization.

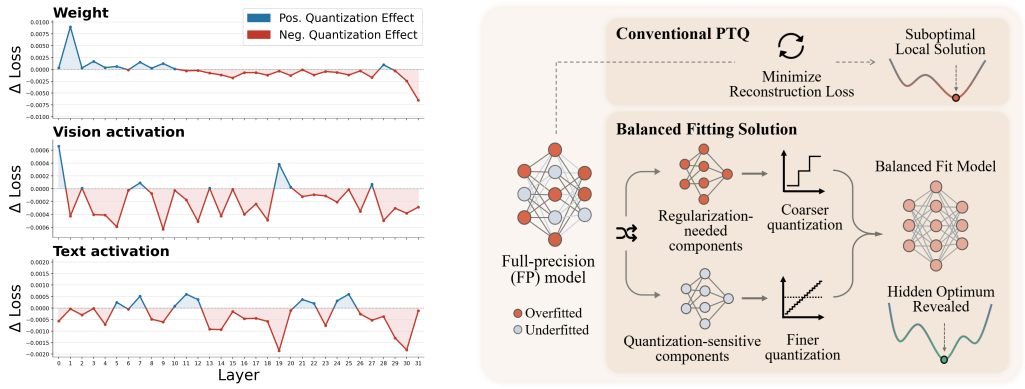
To address these limitations, we propose Balanced Fitting, a novel PTQ framework for LVLMs that goes beyond local reconstruction error minimization. As illustrated in Figure 2(b), our key idea is to adaptively control the fitting strength of quantization using component-wise effects, balancing precise fitting for sensitive components and coarser fitting where quantization serves as regularization. Specifically, we first measure the quantization effect of weights, vision activations, and text activations at each layer, indicating whether quantization disrupts important information or provides beneficial regularization on calibration samples. Based on these effects, we adapt fitting capacity using component-aware scores, encouraging finer-grained fitting for sensitive components and coarser fitting where quantization serves as regularization. This leads to a balanced distribution of quantization resources that better preserves generalization.

Extensive experiments on state-of-the-art LVLMs demonstrate the effectiveness of our approach. As shown in Figure 1, our method consistently outperforms prior PTQ methods for LVLMs under both weight-only and weight-activation settings (W3A16 and W4A8), substantially reducing the accuracy drop from the full-precision model. Notably, on MMMU, a challenging benchmark requiring broad multidisciplinary knowledge and visual reasoning, our approach achieves a 2.43× average and up to 5.2× reduction in accuracy drop relative to the strongest prior PTQ baseline across models and bit-widths, demonstrating robust generalization on complex multimodal tasks.

## 2 Related Work

### 2.1 Large Vision Language Models

Large language models (LLMs) have demonstrated strong reasoning capabilities across diverse tasks (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023). Large vision-language models (LVLMs) project features from vision encoders such as ViT (Dosovitskiy et al., 2020) and CLIP (Radford et al., 2021) into the language embedding space and combine them with text tokens for multimodal reasoning (Alayrac et al., 2022; Li et al., 2023a). Representative models include LLaVA-OneVision (Li et al., 2024), InternVL (Chen et al., 2024b), and Qwen-VL (Wang et al., 2024b). While prior work primarily focuses on improving cross-modal alignment and reasoning, we instead focus on efficient deployment, particularly post-training quantization for LVLMs.




**(a) InternVL2-8B** **(b) Conventional PTQ vs Balanced Fitting**

Figure 2: Motivation of our approach. (a) Component-wise quantization effects across layers reveal heterogeneous sensitivity across weights and modalities, where some components benefit from quantization while others are sensitive. (b) Conventional PTQ uniformly minimizes reconstruction error, leading to suboptimal solutions, whereas our Balanced Fitting adapts fitting strength using component-wise effects, improving generalization.

### 2.2 Post-Training Quantization

Post-training quantization (PTQ) is a widely used compression technique that reduces model precision without retraining (Banner et al., 2019). For vision transformers, prior work addresses challenges in token-wise representations and attention through various calibration and quantization strategies (Liu et al., 2021; Yuan et al., 2022; Li & Gu, 2023; Lin et al., 2021). For large language models (LLMs), existing approaches mitigate quantization errors caused by activation outliers and model scale (Yao et al., 2022; Wei et al., 2022; Liu et al., 2023b). PTQ has also been extended to large vision-language models (LVLMs). Representative methods incorporate sensitivity signals such as cross-layer dependencies (Q-VLM (Wang et al., 2024a)), modality-wise gradients (MBQ (Li et al., 2025)), and token-level importance (QIG (Xiang et al., 2026)), yet still rely on reconstruction-based optimization over limited calibration data. Our approach instead adapts fitting strength to component-wise quantization effects, preserving sensitive components while allowing beneficial deviations from the FP model.

## 3 Method

### 3.1 Preliminaries

Quantization maps full-precision weights and activations to a discrete set of low-bit values, reducing memory and computation. We use uniform integer quantization, with group-wise affine quantization for weights and symmetric per-token quantization for activations. The format W$x$A$y$ denotes $x$-bit weights and $y$-bit activations; for example, W4A8 uses 4-bit weights and 8-bit activations.

Let $\mathbf{W}$ and $\mathbf{X}$ denote the full-precision weight and activation matrices, respectively. PTQ commonly minimizes the reconstruction error between quantized and full-precision outputs. Existing LVLM PTQ methods (Wang et al., 2024a; Li et al., 2025; Xiang et al., 2026) use channel-wise equalization (CWE) to mitigate activation outliers by inversely rescaling $\mathbf{W}$ and $\mathbf{X}$. The channel-wise scaling vector $\mathbf{E}$ is optimized to minimize reconstruction error:

$$\mathbf{E}^* = \arg\min_{\mathbf{E}} \left\| Q(\mathbf{W} \odot \mathbf{E})\, Q(\mathbf{E}^{-1} \odot \mathbf{X}) - \mathbf{W}\mathbf{X} \right\|_2^2, \tag{1}$$

where $Q(\cdot)$ denotes quantization and $\odot$ denotes channel-wise scaling. We retain the quantization operator and reconstruction objective, changing only the number of CWE search candidates per layer.

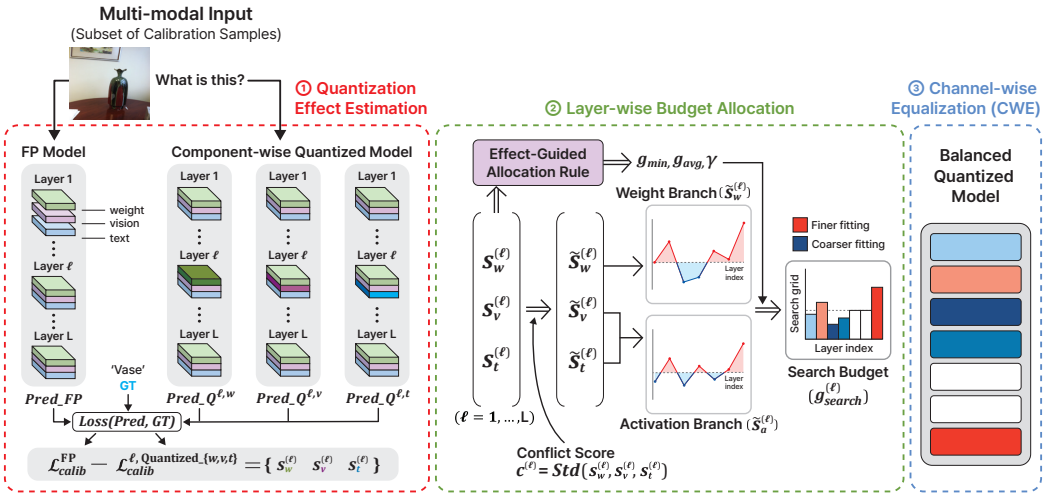



Figure 3: Overview of the proposed balanced fitting framework. We first estimate layer- and component-wise quantization effects by selectively quantizing weights and activations. These effects are then used to guide a layer-wise budget allocation policy that adaptively controls the search granularity. The resulting allocation balances precise fitting for quantization-sensitive components and coarser fitting for others, improving generalization.

### 3.2 Layer- and Component-wise Quantization Effects

As illustrated in Figure 3, we first estimate layer- and component-wise quantization effects by selectively quantizing each component and measuring its impact on the calibration loss. Instead of assigning uniform fitting strength across layers, our method characterizes how each component responds to quantization and uses this information to guide quantization budget allocation. Let $\ell \in \{1, \ldots, L\}$ denote the transformer layer index. For each layer, we consider quantizing one component at a time with the same scheme used in the target PTQ setting, where $c \in \{w, v, t\}$ indicates weights, vision activations, and text activations, respectively. Given a calibration set, we define the calibration loss on a small subset as the token-level negative log-likelihood (NLL) of ground-truth target tokens:

$$\mathcal{L}_{\text{calib}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T_i} \sum_{r=1}^{T_i} -\log p_\theta(y_{i,r} \mid x_i, y_{i,<r}) \,, \tag{2}$$

where $N$ is the calibration-subset size, $x_i$ the $i$-th multimodal input, $T_i$ the number of valid target tokens, and $y_{i,r}$ the $r$-th ground-truth token. Here, $p_\theta$ is the model output distribution under the current quantization setting. We then quantize only component $c$ at layer $\ell$ while keeping all others unchanged and define the quantization effect as

$$s_c^{(\ell)} = \mathcal{L}_{\text{calib}}^{(\text{FP})} - \mathcal{L}_{\text{calib}}^{(\ell, c\text{-quantized})}, \quad c \in \{w, v, t\}. \tag{3}$$

A positive value of $s_c^{(\ell)}$ indicates that quantization reduces the calibration loss, suggesting a potential regularization benefit for that component. Conversely, a negative value indicates that quantization increases the loss, implying that preserving full-precision information is important. Figure 2(a) illustrates the measured $s_c^{(\ell)}$ on the InternVL2-8B model (Chen et al., 2024b). The results reveal that the impact of quantization varies significantly across layers and components (weights, vision activations, and text activations), highlighting the need for quantization adapted to this heterogeneity.

Based on this observation, we transform $s_c^{(\ell)}$ into an importance score for budget allocation. Sensitive components receive positive importance scores, whereas components that benefit from quanti-

zation receive negative scores to encourage coarser fitting:

$$\hat{s}_c^{(\ell)} = \begin{cases} -\alpha\, s_c^{(\ell)}, & s_c^{(\ell)} \geq 0, \\ -s_c^{(\ell)}, & s_c^{(\ell)} < 0, \end{cases} \tag{4}$$

The transformed score $\hat{s}_c^{(\ell)}$ represents the resulting importance of component $c$ at layer $\ell$, where $\alpha \in [0, 1]$ controls how strongly positive quantization effects are down-weighted in the importance score. For activations, we aggregate the visual and textual components to obtain a unified importance score, $\hat{s}_a^{(\ell)} = \hat{s}_v^{(\ell)} + \hat{s}_t^{(\ell)}$, where $\hat{s}_a^{(\ell)}$ denotes the activation importance score at layer $\ell$.

In practice, different components within the same layer can exhibit heterogeneous responses to quantization. When such disagreement occurs, directly aggregating component-wise quantization effects may obscure vulnerable components, potentially causing sensitive components to receive insufficient search capacity. To address this issue, we introduce a layer-wise conflict score and incorporate it into the final importance scores as follows:

$$\begin{aligned} \kappa^{(\ell)} &= \mathrm{Std}\Big(s_w^{(\ell)}, s_v^{(\ell)}, s_t^{(\ell)}\Big), \\ \tilde{s}_w^{(\ell)} &= \hat{s}_w^{(\ell)} + \beta\, \kappa^{(\ell)}, \quad \tilde{s}_a^{(\ell)} = \hat{s}_a^{(\ell)} + \beta\, \kappa^{(\ell)}. \end{aligned} \tag{5}$$

where $\kappa^{(\ell)}$ measures the degree of disagreement among component-wise quantization effects in the $\ell$-th layer, and $\beta$ controls how strongly this disagreement is reflected in the final importance scores, thereby encouraging additional allocation to layers with heterogeneous sensitivity. This dispersion captures differences in magnitude as well as sign. In all experiments, we set $\alpha = 0.5$ and $\beta = 0.5$.

### 3.3 Layer-wise Budget Allocation

We allocate the search budget across layers based on the importance scores computed in the previous section. Concretely, each layer $\ell$ is assigned a discrete budget $g^{(\ell)}$, representing the number of grid points used to optimize channel-wise equalization (CWE). We define a budget policy with three quantities: the minimum per-layer budget $g_{\min}$, the target average budget $g_{\mathrm{avg}}$, and the concentration factor $\gamma$. These values are automatically determined from the distribution of quantization effects $\{s_c^{(\ell)}\}$ using our Quantization Effect-Guided Allocation Rule, with details provided in Appendix B. Each layer is first assigned the minimum budget $g_{\min}$, and the total budget is defined as

$$G_{\min} = L \cdot g_{\min}, \quad G_{\mathrm{target}} = \mathrm{round}(L \cdot g_{\mathrm{avg}}), \quad R = G_{\mathrm{target}} - G_{\min}. \tag{6}$$

Here, $G_{\min}$ is the minimum total budget obtained by assigning $g_{\min}$ to all $L$ layers, $G_{\mathrm{target}}$ is the desired total budget determined by the target average budget $g_{\mathrm{avg}}$, and $R$ denotes the remaining budget to be distributed across layers. Let $u^{(\ell)}$ denote the branch-specific importance score, where $u^{(\ell)} = \tilde{s}_w^{(\ell)}$ for the weight branch and $u^{(\ell)} = \tilde{s}_a^{(\ell)}$ for the activation branch. We then transform these scores into allocation weights and distribute the remaining budget accordingly:

$$\begin{aligned} t^{(\ell)} &= \max\Big(u^{(\ell)} - q_{\mathrm{low}}, 0\Big), \qquad p^{(\ell)} = \frac{\big(t^{(\ell)}\big)^{\gamma}}{\sum_{j=1}^{L} \big(t^{(j)}\big)^{\gamma}}, \\ g^{(\ell)} &= g_{\min} + \mathrm{round}(R \cdot p^{(\ell)}). \end{aligned} \tag{7}$$

where $q_{\mathrm{low}}$ is the 50th percentile of the branch-wise score distribution and $\gamma$ controls how strongly the remaining budget is concentrated on high-importance layers. Layers below this quantile remain at $g_{\min}$, while upper-tail layers receive additional budget according to $p^{(\ell)}$; rounding is adjusted to preserve $G_{\mathrm{target}}$. Thus, smaller budgets yield coarser CWE search that can act as regularization, whereas additional budget enables more precise fitting.

Although importance scores are computed at the component level, we allocate the search budget at the layer level because CWE parameters are jointly optimized within each layer and component effects are inherently coupled. Using a unified budget per layer enables efficient exploration of candidate configurations that account for these interactions. Therefore, we allocate a single search budget $g_{\mathrm{search}}^{(\ell)}$ per layer, determined by the quantization setting.

For W3A16, only the weight branch is used, and the search budget is $g_{\text{search}}^{(\ell)} = g_w^{(\ell)}$. For W4A8, both the weight and activation branches contribute to the final search budget. We first compute the total positive importance of each branch and derive global branch weights:

$$\begin{aligned} S_w &= \sum_{\ell} \max(\tilde{s}_w^{(\ell)}, 0), \qquad S_a = \sum_{\ell} \max(\tilde{s}_a^{(\ell)}, 0), \\ \rho_w &= \frac{S_w}{S_w + S_a}, \quad \rho_a = 1 - \rho_w, \qquad g_{\text{search}}^{(\ell)} = \text{round}\Big(\rho_w g_w^{(\ell)} + \rho_a g_a^{(\ell)}\Big). \end{aligned} \tag{8}$$

Here, $S_w$ and $S_a$ denote the total positive importance mass of the weight and activation branches, respectively. They summarize how strongly each branch demands additional search capacity across all layers. The normalized branch weights $\rho_w$ and $\rho_a$ determine each branch's contribution to the final budget. The branch-wise budgets $g_w^{(\ell)}$ and $g_a^{(\ell)}$ follow the allocation rule above; their rounded weighted combination gives the W4A8 search budget $g_{\text{search}}^{(\ell)}$.

### 3.4 CWE WITH BUDGET-CONTROLLED SEARCH

Given the allocated layer-wise search budget $g_{\text{search}}^{(\ell)}$, we apply channel-wise equalization (CWE) to optimize the scaling factors for each layer. For each linear layer, we search for an optimal scaling vector $E$ within a discrete set of candidate values determined by the allocated budget. The optimization follows the standard CWE objective:

$$E^* = \arg\min_{E} \sum_{i=1}^{M} \left\| Q_W(W \odot E)\, Q_X(E^{-1} \odot X_i) - W X_i \right\|_2^2, \tag{9}$$

where $X_i$ denotes the activation of the $i$-th calibration token; activation quantization is omitted for weight-only settings. Unlike prior approaches that incorporate sensitivity signals while still applying uniform fitting strength across layers, we preserve the original CWE formulation and instead control the granularity of the scale search via the allocated budget. Larger budgets enable finer-grained search for quantization-sensitive layers, while smaller budgets enforce coarser search, potentially reducing overfitting. This design aims to improve generalization by controlling the fitting granularity without modifying the underlying objective.

## 4 EXPERIMENTS

### 4.1 EXPERIMENTAL SETUP

**Implementation Details.** Following prior work (Li et al., 2025; Xiang et al., 2026), we use group-wise affine weight quantization and symmetric per-token activation quantization, and evaluate all methods under W3A16 and W4A8 settings. Experiments use a single NVIDIA RTX A6000 GPU (48GB), except for InternVL2-8B (Chen et al., 2024b) evaluation on MMMU (Yue et al., 2024) and InternVL2-26B evaluation, which use NVIDIA DGX Spark for its larger unified memory.

**Datasets and Models.** We use the improved COCO Caption dataset (Chen et al., 2015) from ShareGPT4V (Chen et al., 2024a) for calibration, sampling 64 image–caption pairs formatted with each LVLM's prompt template; as this calibration set is already small, we use all samples for both quantization effect estimation and CWE optimization. Following LMMs-Eval (Zhang et al., 2025), we evaluate on MMMU (Yue et al., 2024) and ScienceQA (Lu et al., 2022) for visual reasoning, VizWiz (Gurari et al., 2018) for real-world perception, and ChartQA (Masry et al., 2022) and AI2D (Kembhavi et al., 2017) for visual understanding. We benchmark LLaVA-OneVision-7B (Li et al., 2024), InternVL2-8B (Chen et al., 2024b), and Qwen2-VL-7B (Wang et al., 2024b). For InternVL2-26B, we additionally evaluate image captioning on NoCaps-lite (Agrawal et al., 2019).

**Baselines.** We compare with round-to-nearest (RTN), Q-VLM (Wang et al., 2024a), Modality-Balanced Quantization (MBQ) (Li et al., 2025), and Quantization-Aware Integrated Gradients (QIG) (Xiang et al., 2026). For Q-VLM, MBQ, and QIG, we follow the official open-source implementations. Since Q-VLM is designed for LLaVA-style architectures, we adapt and re-implement it to ensure compatibility with all evaluated models.

Table 1: Quantitative results across three LVLMs and five benchmarks. ΔFP: Avg. − FP Avg. (pp).

| Model | Bitwidth | Method | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. | ΔFP |
|---|---|---|---|---|---|---|---|---|---|
| LLaVA-OV-7B | FP16 | - | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 | — |
| | W3A16 | RTN | 42.22 | 57.00 | 94.55 | 68.88 | 78.98 | 68.33 | −4.15 |
| | | Q-VLM | 44.22 | 59.72 | 94.40 | 76.76 | 78.63 | 70.75 | −1.73 |
| | | MBQ | 43.11 | 59.88 | 94.74 | 77.00 | 78.27 | 70.60 | −1.88 |
| | | QIG | 43.89 | 59.54 | 94.65 | 77.12 | 78.47 | 70.73 | −1.75 |
| | | Ours | **46.11** | **60.28** | **94.94** | **77.20** | **79.02** | **71.51** | **−0.97** |
| | W4A8 | RTN | 42.67 | 54.92 | 94.10 | 70.28 | 78.34 | 68.06 | −4.42 |
| | | Q-VLM | 40.22 | 51.93 | 92.36 | 71.48 | 77.14 | 66.63 | −5.85 |
| | | MBQ | 43.44 | 55.00 | **94.25** | 70.44 | 77.98 | 68.22 | −4.26 |
| | | QIG | 44.33 | 52.48 | 93.70 | 75.00 | 77.30 | 68.56 | −3.92 |
| | | Ours | **45.22** | **55.78** | 94.00 | **75.08** | **78.50** | **69.72** | **−2.76** |
| InternVL2-8B | FP16 | - | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 | — |
| | W3A16 | RTN | 44.56 | 55.74 | **96.28** | 79.52 | **80.54** | 71.33 | −2.81 |
| | | Q-VLM | 45.33 | 59.60 | 96.18 | 79.00 | 80.18 | 72.06 | −2.08 |
| | | MBQ | 45.00 | **59.87** | 96.18 | 78.76 | 79.66 | 71.89 | −2.25 |
| | | QIG | 45.78 | 58.42 | 96.13 | 79.52 | 80.31 | 72.03 | −2.11 |
| | | Ours | **47.44** | 59.77 | 96.18 | **80.12** | 79.63 | **72.63** | **−1.51** |
| | W4A8 | RTN | 43.11 | 57.04 | 96.18 | 78.48 | 78.92 | 70.75 | −3.39 |
| | | Q-VLM | 39.89 | 56.57 | 93.26 | 73.60 | 74.81 | 67.63 | −6.51 |
| | | MBQ | 44.44 | 57.59 | 96.33 | 78.16 | 79.34 | 71.17 | −2.97 |
| | | QIG | 44.89 | 55.72 | **96.58** | **78.76** | 79.73 | 71.14 | −3.00 |
| | | Ours | **46.33** | **58.01** | **96.58** | 78.44 | **79.83** | **71.84** | **−2.30** |
| Qwen2-VL-7B | FP16 | - | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 | — |
| | W3A16 | RTN | 44.78 | 66.28 | 81.41 | 73.84 | 76.62 | 68.59 | −4.61 |
| | | Q-VLM | 47.44 | 66.73 | 79.97 | **78.36** | 77.56 | 70.01 | −3.19 |
| | | MBQ | 46.22 | 64.87 | 82.80 | 78.00 | 77.33 | 69.84 | −3.36 |
| | | QIG | 45.56 | 64.95 | **82.90** | 78.16 | 77.30 | 69.77 | −3.43 |
| | | Ours | **48.00** | **66.86** | 82.30 | 77.84 | **77.82** | **70.56** | **−2.64** |
| | W4A8 | RTN | 45.78 | 58.59 | 79.23 | 74.96 | 76.17 | 66.95 | −6.25 |
| | | Q-VLM | 42.11 | 51.30 | 80.52 | 70.72 | 72.99 | 63.53 | −9.67 |
| | | MBQ | 45.22 | 58.99 | 79.42 | 77.00 | 76.91 | 67.51 | −5.69 |
| | | QIG | 46.33 | 61.26 | 79.57 | 76.76 | **77.53** | 68.29 | −4.91 |
| | | Ours | **47.00** | **62.93** | **80.81** | **77.12** | 76.33 | **68.84** | **−4.36** |

Table 2: InternVL2-26B under W4A8. NoCaps-lite: CIDEr ×100.

| Method | MMMU | ScienceQA | AI2D-lite | NoCaps-lite | Avg. | ΔFP |
|---|---|---|---|---|---|---|
| FP | 47.11 | 97.27 | 83.00 | 87.74 | 78.78 | — |
| MBQ | 44.67 | 96.93 | 78.60 | 77.48 | 74.42 | −4.36 |
| QIG | 44.00 | 96.63 | 78.40 | **79.66** | 74.67 | −4.11 |
| Ours | **44.89** | **96.98** | **79.00** | 78.98 | **74.96** | **−3.82** |

### 4.2 Main Results

Table 1 and Figure 1 show that Balanced Fitting outperforms existing PTQ methods in five-task average across all models and both precisions, indicating consistent performance preservation across diverse tasks. On MMMU (Yue et al., 2024), which requires multidisciplinary knowledge and visual reasoning, it reduces the accuracy drop by 2.43× on average relative to the strongest competing baseline. Table 2 extends this comparison to the larger InternVL2-26B under W4A8, where our method also leads in average performance.

Figure 4 presents qualitative comparisons on MMMU and VizWiz. Balanced Fitting preserves correct FP answers while correcting erroneous predictions reproduced by MBQ and QIG, illustrating a balance between retaining useful FP behavior and allowing beneficial deviations.

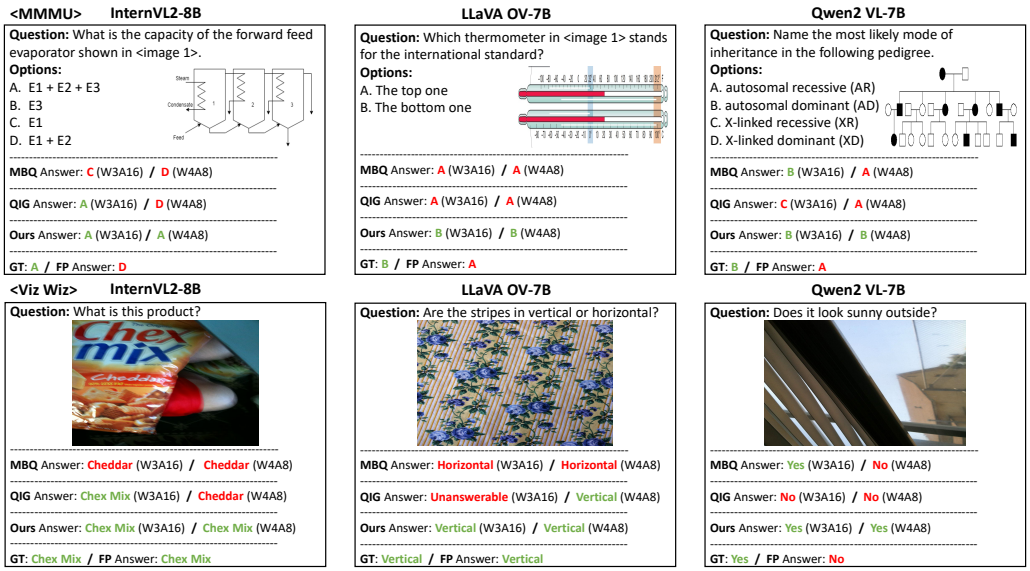



Figure 4: Qualitative comparison on MMMU and VizWiz.

Table 3: Ablation on the gain-discount coefficient ($\alpha$) and conflict coefficient ($\beta$) under the W4A8 setting. Each cell reports MMMU / 5-task Avg.

| Model | Gain-discount coefficient ($\alpha$) | | | Conflict coefficient ($\beta$) | | |
|---|---|---|---|---|---|---|
| | 0 | 0.5 | 1.0 | 0 | 0.5 | 1.0 |
| Qwen2-VL-7B | 46.67 / 68.70 | 47.00 / **68.84** | **47.33** / 68.83 | 45.67 / **68.98** | **47.00** / 68.84 | 46.22 / 68.59 |
| LLaVA-OV-7B | 43.67 / 69.06 | **45.22 / 69.72** | 41.78 / 68.69 | 43.67 / 69.37 | **45.22 / 69.72** | 44.56 / 69.18 |
| InternVL2-8B | 43.22 / 71.16 | **46.33** / 71.84 | 45.56 / **71.87** | 45.44 / 71.44 | **46.33 / 71.84** | 44.67 / 71.44 |

Table 4: Uniform vs. adaptive grid allocation. Adap. ($k$) indicates mean grid size $k$.

| Precision | Method | InternVL2-8B | | | | LLaVA-OV-7B | | | | Qwen2-VL-7B | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Grid | Loss ↓ | MMMU ↑ | Avg. ↑ | Grid | Loss ↓ | MMMU ↑ | Avg. ↑ | Grid | Loss ↓ | MMMU ↑ | Avg. ↑ |
| W3A16 | Low | 18 | 2.34 | 46.4 | 72.4 | 16 | **0.82** | 42.6 | 70.9 | 20 | 0.85 | 47.2 | 70.5 |
| | Mean | 23 | **1.85** | 46.6 | 72.3 | 27 | 0.96 | 44.8 | 70.6 | 22 | 0.85 | 47.4 | 70.4 |
| | High | 28 | 1.94 | 45.9 | 72.1 | 32 | 0.94 | 45.7 | 70.8 | 24 | **0.79** | 46.6 | 70.0 |
| | Ours | Adap. (23) | 2.34 | **47.4** | **72.6** | Adap. (27) | 0.90 | **46.1** | **71.5** | Adap. (22) | 0.87 | **48.0** | **70.6** |
| W4A8 | Low | 17 | **0.28** | 44.4 | 71.3 | 15 | 0.32 | 43.4 | 69.1 | 16 | 0.27 | 46.4 | **68.8** |
| | Mean | 21 | 0.41 | 44.6 | 71.2 | 18 | 0.38 | 45.0 | 69.3 | 19 | **0.25** | 45.3 | 67.8 |
| | High | 25 | 0.37 | 45.0 | 71.3 | 23 | **0.30** | 43.1 | 68.9 | 23 | **0.25** | 46.4 | 68.7 |
| | Ours | Adap. (21) | 0.32 | **46.3** | **71.8** | Adap. (18) | 0.33 | **45.2** | **69.7** | Adap. (19) | 0.28 | **47.0** | **68.8** |

### 4.3 Ablation Study and Analysis

To examine the effect of each allocation component, we vary the gain-discount coefficient $\alpha$ in equation 4 and the conflict coefficient $\beta$ in equation 5. These coefficients control how strongly beneficial quantization effects are discounted and conflicting component responses are emphasized, respectively. Table 3 reports the resulting performance. $\alpha = \beta = 0.5$ yields competitive performance across models and metrics.

We next examine whether adaptive allocation improves on uniform budgets, comparing reconstruction loss and downstream performance in Table 4. Adaptive allocation consistently achieves superior performance across all models and settings. Notably, uniformly allocating high budgets can underperform configurations with lower budgets, indicating that excessive fitting can be detrimental. Moreover, we observe that models with lower reconstruction loss do not necessarily achieve better downstream performance, highlighting the limitation of reconstruction-based optimization.

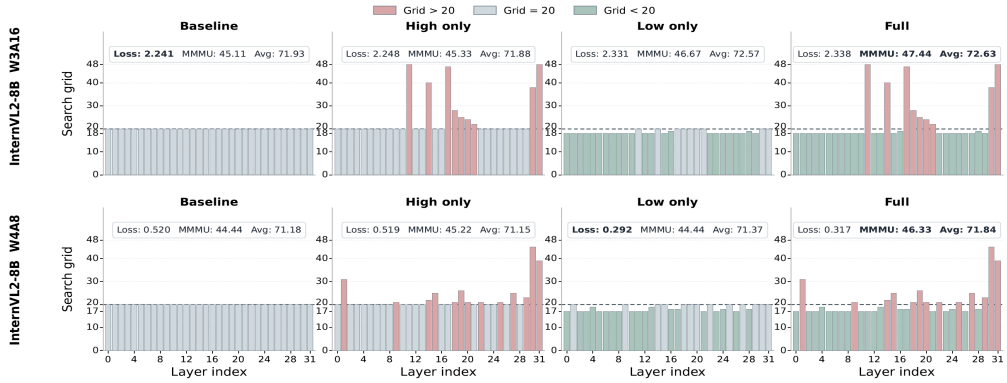



Figure 5: Layer-wise budget allocation and downstream performance on InternVL2-8B.

Table 5: Quantization effect (QE) stability across disjoint calibration subsets and an independent held-out subset under W3A16 and W4A8. Parentheses give agreement counts.

| QE-ranked subset | Count | QE mass | 3-subset sign agreement | Held-out agreement | Beneficial held-out agreement |
|---|---|---|---|---|---|
| Top 50% | 176 | 89.90% | 87.5% (154/176) | 93.2% (164/176) | 89.8% (44/49) |
| Top 25% | 88 | 73.90% | 94.3% (83/88) | 100% (88/88) | 100% (23/23) |
| Top 10% | 38 | 57.00% | 100% (38/38) | 100% (38/38) | 100% (10/10) |

Table 6: Calibration-domain shift from COCO Caption to Flickr30k: W4A8 five-task average.

| Model | FP | MBQ (ΔFP) | QIG (ΔFP) | Ours (ΔFP) |
|---|---|---|---|---|
| InternVL2 | 74.14 | 71.60 (−2.54) | 70.87 (−3.27) | **71.78** (**−2.36**) |
| LLaVA-OV | 72.48 | 68.06 (−4.42) | 68.92 (−3.56) | **69.43** (**−3.05**) |
| Qwen2-VL | 73.20 | 67.57 (−5.63) | 67.27 (−5.93) | **67.68** (**−5.52**) |

Figure 5 compares refinement (High-only), coarsening (Low-only), and Balanced Fitting (Full), which combines both. Reconstruction loss fails to track downstream performance across these variants, making it an unreliable performance proxy. Full achieves the best average; Low-only also outperforms the uniform baseline, consistent with a regularization effect from constrained fitting.

To verify that quantization effects remain stable across samples, we compare their signs across three disjoint COCO-64 subsets and an independent held-out-64 subset, ranking components using calibration data only. Table 5 shows strong held-out sign agreement for high-magnitude effects, including beneficial effects. This suggests that the signals guiding refinement and coarsening are not specific to a single calibration subset, supporting their use for budget allocation. Appendix Figure 8 further shows consistent layer-wise estimates across sample sizes.

To test robustness to a calibration-domain shift from COCO Caption to Flickr30k, we re-estimate quantization effects on 64 Flickr30k training pairs and apply the unchanged W4A8 allocation rule without retuning. Table 6 shows that Ours retains the highest five-task average across all three models, supporting robustness to a change in calibration data. Additional controls, analyses, and costs are provided in the appendix.

## 5 Conclusion

In this work, we revisit post-training quantization (PTQ) for large vision-language models (LVLMs) and show that minimizing reconstruction loss on small calibration sets is insufficient for generalization. We propose Balanced Fitting, a quantization effect-based framework that allocates layer-wise budgets to control fitting strength, balancing precision and regularization across components. Extensive experiments demonstrate consistent improvements over prior PTQ methods and reveal the limitation of reconstruction loss as a proxy for downstream performance.

## AI USE STATEMENT

Generative AI was used to assist with manuscript restructuring and revision, English phrasing, literature identification and reference formatting, and to provide feedback on the presentation and interpretation of experimental results. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final text, claims, results, and artifacts.

## REPRODUCIBILITY STATEMENT

We provide the model backbones, benchmark datasets, calibration data, quantization settings, evaluation metrics, and baseline adaptation details used in our experiments. Balanced Fitting requires no model retraining. Details of quantization effect estimation, the allocation rule and its parameters, budget-controlled CWE search, evaluation protocols, and calibration-time measurements are provided in the main text and Appendix. The supplementary material further includes the implementation and evaluation code needed to reproduce the reported results.

## REFERENCES

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8947–8956. IEEE, 2019.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. *Advances in neural information processing systems*, 32, 2019.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024a.

Wentao Chen, Hailong Qiu, Jian Zhuang, Chutong Zhang, Yu Hu, Qing Lu, Tianchen Wang, Yiyu Shi, Meiping Huang, and Xiaowe Xu. Quantization of deep neural networks for accurate edge computing. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 17(4):1–11, 2021.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024b.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of machine learning research*, 24(240):1–113, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3608–3617, 2018.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2704–2713, 2018.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, pp. 4999–5007, 2017.

Andrey Kuzmin, Markus Nagel, Mart Van Baalen, Arash Behboodi, and Tijmen Blankevoort. Pruning vs quantization: Which is better? *Advances in neural information processing systems*, 36: 62414–62427, 2023.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

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

Xuanlin Li, Yunhao Fang, Minghua Liu, Zhan Ling, Zhuowen Tu, and Hao Su. Distilling large vision-language model with out-of-distribution generalizability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2492–2503, 2023b.

Zhikai Li and Qingyi Gu. I-vit: Integer-only quantization for efficient vision transformer inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17065–17075, 2023.

Tailin Liang, John Glossner, Lei Wang, Shaobo Shi, and Xiaotong Zhang. Pruning and quantization for deep neural network acceleration: A survey. *Neurocomputing*, 461:370–403, 2021.

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. *Proceedings of machine learning and systems*, 6:87–100, 2024.

Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Post-training quantization for fully quantized vision transformer. *arXiv preprint arXiv:2111.13824*, 2021.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023a.

Yijiang Liu, Huanrui Yang, Zhen Dong, Kurt Keutzer, Li Du, and Shanghang Zhang. Noisyquant: Noisy bias-enhanced post-training activation quantization for vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20321–20330, 2023b.

Zhenhua Liu, Yunhe Wang, Kai Han, Wei Zhang, Siwei Ma, and Wen Gao. Post-training quantization for vision transformer. *Advances in Neural Information Processing Systems*, 34:28092–28103, 2021.

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

Ahmed Masry, Do Long, Jiaqi Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, 2022.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International conference on machine learning*, pp. 7197–7206. PMLR, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Q-vlm: Post-training quantization for large vision-language models. *Advances in Neural Information Processing Systems*, 37:114553–114573, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Ying Wang, Yadong Lu, and Tijmen Blankevoort. Differentiable joint pruning and quantization for hardware efficiency. In *European Conference on Computer Vision*, pp. 259–277. Springer, 2020.

Xiuying Wei, Yunchen Zhang, Xiangguo Zhang, Ruihao Gong, Shanghang Zhang, Qi Zhang, Fengwei Yu, and Xianglong Liu. Outlier suppression: Pushing the limit of low-bit transformer language models. *Advances in Neural Information Processing Systems*, 35:17402–17414, 2022.

Ziwei Xiang, Fanhu Zeng, Hongjian Fang, Rui-Qi Wang, Renxing Chen, Yanan Zhu, Yi Chen, Peipei Yang, and Xu-Yao Zhang. Fine-grained post-training quantization for large vision language models with quantization-aware integrated gradients. *arXiv preprint arXiv:2603.17809*, 2026.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International conference on machine learning*, pp. 38087–38099. PMLR, 2023.

Zhewei Yao, Reza Yazdani Aminabadi, Minjia Zhang, Xiaoxia Wu, Conglong Li, and Yuxiong He. Zeroquant: Efficient and affordable post-training quantization for large-scale transformers. *Advances in neural information processing systems*, 35:27168–27183, 2022.

Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. Ptq4vit: Post-training quantization for vision transformers with twin uniform quantization. In *European conference on computer vision*, pp. 191–207. Springer, 2022.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 881–916, 2025.

Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.

# A Additional Analysis

**Component-wise Quantization Effects Across Models and Settings.** Figure 6 extends Figure 2(a) to InternVL2-8B (Chen et al., 2024b), LLaVA-OV-7B (Li et al., 2024), and Qwen2-VL-7B (Wang et al., 2024b) under W3A16 and W4A8.

The sign and magnitude of quantization effects vary across layers, components, models, and bitwidths. Negative effects indicate increased calibration loss, whereas positive effects indicate reduced loss and suggest potential regularization benefits. The locations of sensitive and beneficial components differ across models and settings, supporting allocation based on the measured effects rather than a fixed layer-wise policy.

**Effect of Balanced Budget Allocation on Downstream Performance.** Figure 7 extends the allocation ablation to LLaVA-OV-7B and Qwen2-VL-7B. Under W4A8, it compares uniform allocation, refinement alone (High-only), coarsening alone (Low-only), and Balanced Fitting (Full), allowing the contributions of refinement and coarsening to be examined separately. The results support combining both operations rather than relying on either alone. Under W3A16, the minimum grid size is 20, so High-only coincides with Full and Low-only coincides with the uniform baseline; only the distinct configurations are shown. This setting therefore evaluates refinement above the baseline budget rather than a separate coarsening effect.

To examine how calibration-set size affects quantization effect estimation, Figure 8 compares layer-wise profiles using 16, 32, and 64 samples across three LVLMs under W3A16 and W4A8. The profiles remain highly correlated across sample sizes, supporting reliable estimation with limited calibration data. This complements Table 5, which tests sign agreement across disjoint calibration subsets and an independent held-out subset.

# B Quantization Effect-Guided Allocation Rule

To reduce the manual effort in configuring Balanced Fitting, we provide a simple statistics-driven rule that maps quantization effect distributions to allocation settings. This rule selects allocation settings automatically and achieves performance close to empirically selected configurations. The same formulas and normalization constants are used across models within each quantization mode. The rule operates directly on summary statistics of quantization effects and is included as a practical supplementary tool.

**Quantization effect statistics.** For each layer $\ell$ and component $c$, let $s_c^{(\ell)}$ denote the quantization effect defined in equation 3. We decompose it into positive and negative parts:

$$s_{\ell,c}^{\text{loss}} = \max(-s_c^{(\ell)}, 0), \qquad s_{\ell,c}^{\text{gain}} = \max(s_c^{(\ell)}, 0). \tag{10}$$

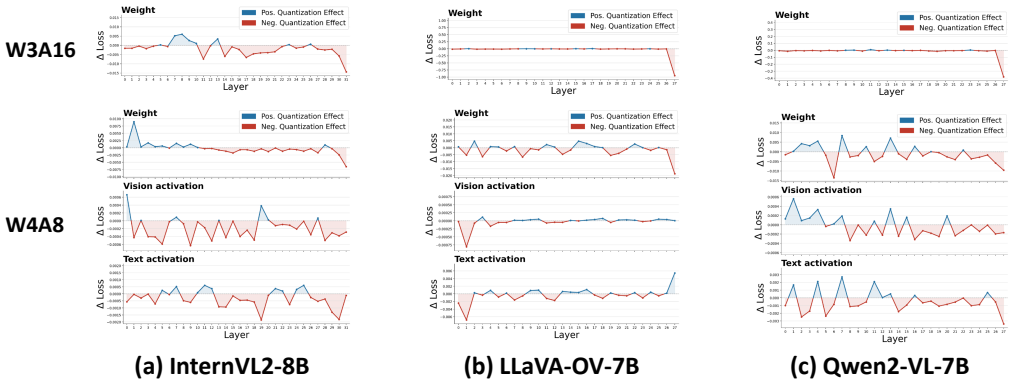



Figure 6: Component-wise quantization effects across layers for three vision-language models (InternVL2-8B, LLaVA-OV-7B, and Qwen2-VL-7B) under two quantization settings (W3A16 and W4A8). The results reveal heterogeneous sensitivity across weights and modalities, where some components benefit from quantization while others are more sensitive.

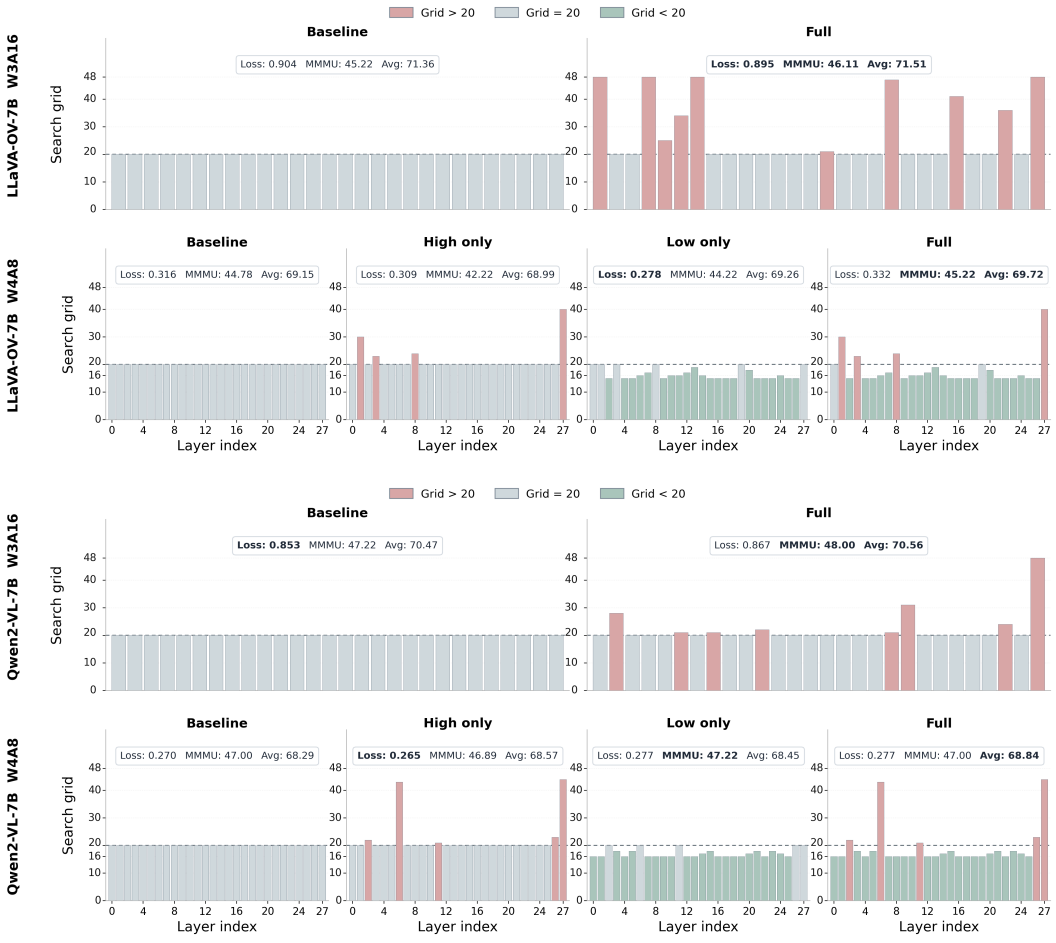



Figure 7: Layer-wise budget allocation and downstream performance on LLaVA-OV-7B and Qwen2-VL-7B. For the W3A16 setting, the minimum grid size is set to 20, making the high-only configuration equivalent to the full setting and the low-only configuration equivalent to the baseline; thus, they are excluded from comparison.

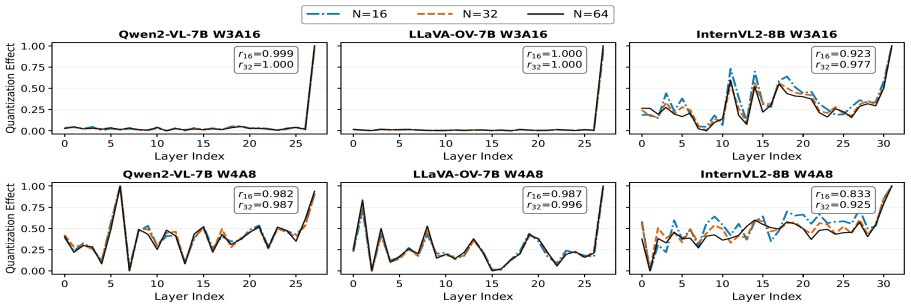



Figure 8: Stability of layer-wise quantization effects estimated from 16, 32, and 64 calibration samples on Qwen2-VL-7B, LLaVA-OV-7B, and InternVL2-8B under W3A16 (top) and W4A8 (bottom). W4A8 aggregates weight, visual-token, and textual-token effects. The reported Pearson correlations compare the 16- and 32-sample profiles with the 64-sample reference.

For W3A16, we use only the weight component:

$$H_\ell = s^{\text{loss}}_{\ell,w}, \qquad G_\ell = s^{\text{gain}}_{\ell,w}, \qquad H = \sum_\ell H_\ell, \qquad G = \sum_\ell G_\ell. \tag{11}$$

We then define the gain ratio

$$r = \frac{G}{H+G}, \tag{12}$$

and the loss severity ratio

$$M = \frac{H}{\mathcal{L}^{(\text{FP})}_{\text{task}}}, \tag{13}$$

where $\mathcal{L}^{(\text{FP})}_{\text{task}}$ is the FP model's calibration-set task loss.

For W4A8, we aggregate all components:

$$H_\ell = \sum_{c\in\{w,v,t\}} s^{\text{loss}}_{\ell,c}, \qquad H = \sum_\ell H_\ell. \tag{14}$$

We additionally define the activation-side loss share

$$A = \frac{\sum_\ell \left(s^{\text{loss}}_{\ell,v} + s^{\text{loss}}_{\ell,t}\right)}{H}, \tag{15}$$

and the support ratio

$$S = \frac{k_{0.9}}{L}, \tag{16}$$

where $k_{0.9}$ is the minimum number of layers needed to explain $90\%$ of the total loss mass and $L$ is the number of layers.

For both quantization modes, we measure concentration using the top $10\%$ of layers with positive loss mass. Let $\mathcal{I}_{0.1}$ denote the set of layer indices corresponding to the largest $\lceil 0.1L_+ \rceil$ values of $H_\ell$ among layers with $H_\ell > 0$, where $L_+$ is the number of layers satisfying $H_\ell > 0$. We define

$$C_{0.1} = \frac{\sum_{\ell\in\mathcal{I}_{0.1}} H_\ell}{H}. \tag{17}$$

**Normalization.** To place the raw statistics on a comparable scale, we normalize each statistic $z$ into the unit interval:

$$\bar{z} = \text{clip}\left(\frac{z - z_{\min}}{z_{\max} - z_{\min}}, 0, 1\right), \tag{18}$$

where $z_{\min}$ and $z_{\max}$ are the normalization bounds. W3A16 uses $r \in [0.017, 0.207]$, $C_{0.1} \in [0.375, 0.915]$, and $M \in [0.065, 1.371]$ (bounds shown rounded to three decimal places). For W4A8, we retain the zero origin of the ratios and round their upper bounds upward to one decimal: $A \in [0, 0.8]$, $S \in [0, 0.7]$, and $C_{0.1} \in [0, 0.6]$. Values outside these ranges are clipped. These bounds and all coefficients below are shared across models within a quantization mode.

**W3A16 rule.** For W3A16, after normalizing $r$, $M$, and $C_{0.1}$ into $\bar{r}$, $\bar{M}$, and $\bar{C}$, we compute

$$g_{\min}^{(\mathrm{w3a16})} = \mathrm{round}\big(g_{\mathrm{ref}} - (2 + 1.6(1-\bar{r}))\,\bar{r}(1-\bar{C})\big)\,, \tag{19}$$

$$g_{\mathrm{avg}}^{(\mathrm{w3a16})} = g_{\mathrm{ref}} + 2 + \mathrm{round}\big(5\,\bar{C}\,\bar{M}^3 + \bar{r}(1-\bar{C})\big)\,, \tag{20}$$

$$\gamma^{(\mathrm{w3a16})} = \big[\,2.12 + 0.45\bar{C}\,\big]_{0.05}\,, \tag{21}$$

where $[\cdot]_{0.05}$ denotes rounding to the nearest multiple of $0.05$.

**W4A8 rule.** For W4A8, after normalizing $A$, $S$, and $C_{0.1}$ into $\bar{A}$, $\bar{S}$, and $\bar{C}$, we compute

$$g_{\min}^{(\mathrm{w4a8})} = g_{\mathrm{ref}} - 8 + \mathrm{round}\big(2\bar{A}\bar{S} + 9.5(1-\bar{C})\big)\,, \tag{22}$$

$$g_{\mathrm{avg}}^{(\mathrm{w4a8})} = g_{\mathrm{ref}} - 4 + \mathrm{round}\big(9\bar{A}\bar{S} + \bar{S}(1-\bar{C})\big)\,, \tag{23}$$

$$\gamma^{(\mathrm{w4a8})} = \big[\,1 + 0.9\bar{C} + 0.6(1-\bar{A})(1-\bar{C})\,\big]_{0.05}\,. \tag{24}$$

Here, $g_{\mathrm{ref}}$ denotes the common reference grid budget. In our experiments, we follow prior work and set $g_{\mathrm{ref}} = 20$. After rounding, we enforce $2 \le g_{\min} \le g_{\mathrm{avg}} \le 48$; no-effect profiles use uniform 20-grid search. Intuitively, for W4A8, $g_{\min}$ decreases when the loss profile is highly concentrated and increases when the loss mass is more diffuse or activation-heavy. The target budget $g_{\mathrm{avg}}$ increases when activation-side loss is broader and stronger, while $\gamma$ is governed by concentration with an adjustment for diffuse low-activation cases.

Figure 9 examines key allocation dimensions, including base grid, target mean, and $\gamma$. The selected settings are close to the best observed configurations, even when they do not always maximize MMMU.

**Interpretation.** The rule remains interpretable by linking allocation decisions to quantization effect statistics. For W3A16, the gain ratio and loss concentration jointly determine the minimum budget and allocation sharpness. For W4A8, the allocation is further influenced by the spread and magnitude of activation-side loss, leading to increased budgets when activation sensitivity is broader and stronger.

# C Additional Comparative Evaluation

## C.1 Evaluation settings

InternVL2-26B uses COCO-64 on one NVIDIA DGX Spark (GB10, 128 GB unified memory). MMMU and ScienceQA use at most five image tiles instead of six for all methods, including FP. Its QE is measured directly, and the common rule reproduces $g_{\min} = 14$, $g_{\mathrm{avg}} = 21$, and $\gamma = 1.8$. AI2D-lite and NoCaps-lite use the official fixed 500-example splits. For Flickr30k, we re-estimate quantization effects on 64 training pairs and apply the same allocation rule, normalization bounds, and coefficients without retuning, yielding $(g_{\min}, g_{\mathrm{avg}}, \gamma) = (16, 25, 1.65)$, $(16, 18, 1.90)$, and $(16, 30, 1.70)$ for InternVL2, LLaVA-OV, and Qwen2-VL, respectively. Calibration pairs do not overlap with the 1,000-image Karpathy test split.

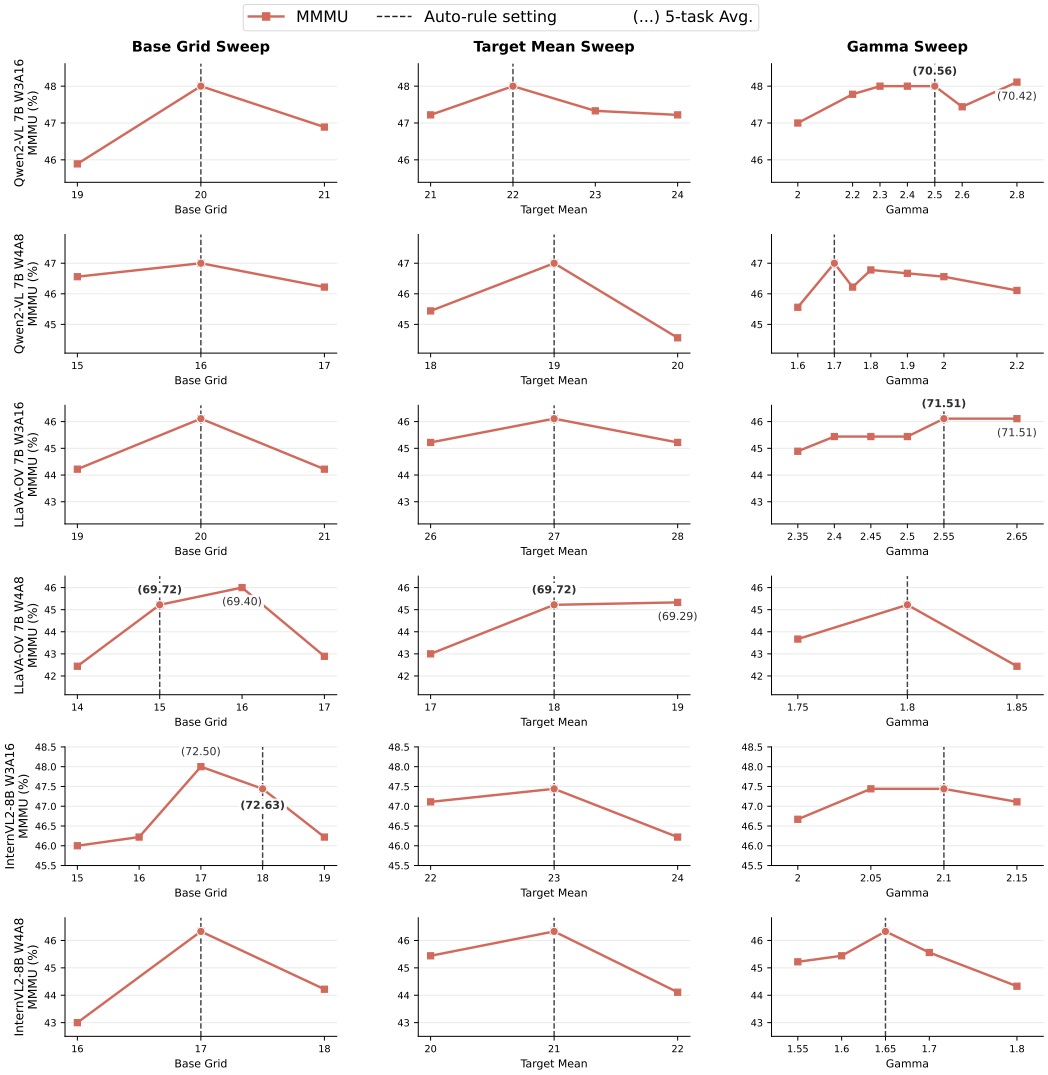



Figure 9: Analysis of the proposed Quantization Effect-Guided Allocation Rule. Each row corresponds to a model and setting, and each column sweeps one allocation dimension (base grid, target mean, or $\gamma$) while keeping the others fixed. Numbers in parentheses indicate the 5-task average performance, and curves report MMMU performance. The dashed vertical line indicates the reference allocation in the sweep.

Table 7: W3A16 per-task results with LLM-backbone AWQ and GPTQ; FP16 is the reference. Bold denotes the best quantized result.

| Model | Method | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. |
|---|---|---|---|---|---|---|---|
| InternVL2 | FP16 | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 |
| | AWQ | 45.22 | 58.06 | 95.84 | 79.36 | **79.99** | 71.69 |
| | GPTQ | 41.33 | **60.52** | 93.90 | 75.68 | 75.58 | 69.40 |
| | Ours | **47.44** | 59.77 | **96.18** | **80.12** | 79.63 | **72.63** |
| LLaVA-OV | FP16 | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 |
| | AWQ | 41.78 | 58.59 | **95.09** | 77.04 | 78.59 | 70.22 |
| | GPTQ | 37.22 | 51.79 | 90.43 | 72.68 | 75.19 | 65.46 |
| | Ours | **46.11** | **60.28** | 94.94 | **77.20** | **79.02** | **71.51** |
| Qwen2-VL | FP16 | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 |
| | AWQ | 45.56 | 63.18 | 82.10 | **78.44** | 77.46 | 69.35 |
| | GPTQ | 41.89 | **67.13** | 77.79 | 72.56 | 72.22 | 66.32 |
| | Ours | **48.00** | 66.86 | **82.30** | 77.84 | **77.82** | **70.56** |

Table 8: Per-task comparison of sensitivity-matched mixed precision, uniform CWE-20, and Ours under W3A16 and W4A8 target budgets. Bold denotes the best quantized result within each target setting. *Collapse* denotes severe performance breakdown with repetitive, invalid MMMU outputs; the five-task average is therefore omitted.

| Model | Target | Method | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. |
|---|---|---|---|---|---|---|---|---|
| InternVL2 | FP16 | FP | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 |
| | W3A16 | Uniform CWE-20 | 45.11 | 59.27 | 95.93 | 79.48 | **79.86** | 71.93 |
| | | Mixed precision | 35.11 | 54.21 | 80.07 | 55.76 | 64.67 | 57.96 |
| | | Ours | **47.44** | **59.77** | **96.18** | **80.12** | 79.63 | **72.63** |
| | W4A8 | Uniform CWE-20 | 44.44 | 57.25 | 96.53 | **78.88** | 78.79 | 71.18 |
| | | Mixed precision | Collapse | 8.50 | 30.44 | 2.80 | 21.15 | – |
| | | Ours | **46.33** | **58.01** | **96.58** | 78.44 | **79.83** | **71.84** |
| LLaVA-OV | FP16 | FP | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 |
| | W3A16 | Uniform CWE-20 | 45.22 | **60.35** | **95.09** | **77.36** | 78.76 | 71.36 |
| | | Mixed precision | 36.00 | 51.87 | 87.46 | 71.12 | 72.51 | 63.79 |
| | | Ours | **46.11** | 60.28 | 94.94 | 77.20 | **79.02** | **71.51** |
| | W4A8 | Uniform CWE-20 | 44.78 | 55.04 | 93.41 | 74.52 | 78.01 | 69.15 |
| | | Mixed precision | 27.33 | 41.56 | 48.34 | 38.52 | 37.76 | 38.70 |
| | | Ours | **45.22** | **55.78** | **94.00** | **75.08** | **78.50** | **69.72** |
| Qwen2-VL | FP16 | FP | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 |
| | W3A16 | Uniform CWE-20 | 47.22 | **67.06** | 81.76 | **78.24** | **78.08** | 70.47 |
| | | Mixed precision | 40.00 | 55.63 | 76.05 | 71.76 | 66.35 | 61.96 |
| | | Ours | **48.00** | 66.86 | **82.30** | 77.84 | 77.82 | **70.56** |
| | W4A8 | Uniform CWE-20 | **47.00** | 58.87 | **80.96** | **77.28** | **77.36** | 68.29 |
| | | Mixed precision | 27.78 | 36.55 | 53.20 | 47.32 | 44.07 | 41.78 |
| | | Ours | **47.00** | **62.93** | 80.81 | 77.12 | 76.33 | **68.84** |

### C.2 LLM-BACKBONE QUANTIZATION BASELINES

To compare Balanced Fitting with established language-model quantization methods, we apply AWQ (Lin et al., 2024) and GPTQ (Frantar et al., 2022) to the LLM backbone under W3A16 while keeping the visual encoder in FP.

Table 7 shows that Ours achieves a higher five-task average on all three models. This comparison extends the evaluation beyond LVLM-specific baselines and supports the value of allocating fitting capacity using quantization effects measured on multimodal calibration data.

### C.3 SENSITIVITY-MATCHED MIXED PRECISION

To distinguish allocating search capacity from allocating numerical precision, we construct a mixed-precision control using the same quantization effect scores. The control assigns W2/W3/W4 at an exact parameter-weighted 3-bit average for W3A16, or fixes W4 and assigns A4/A8/A16 at an exact input-width-weighted 8-bit average for W4A8, monotonically following the same QE score. Every layer uses 20-candidate CWE.

Table 8 compares it with uniform 20-candidate search and Ours. Ours outperforms both controls across all three models and both target budgets. Thus, directly translating the same sensitivity signal into bit-width allocation does not reproduce the benefit of Balanced Fitting in these settings. Adjusting search capacity also retains uniform inference precision; this comparison concerns the tested mixed-precision control rather than mixed-precision methods in general.

### C.4 QIG WITH A LARGER CALIBRATION SET

QIG (Xiang et al., 2026) uses 128 calibration pairs in its reported setting, whereas our main comparison fixes all methods to 64 pairs for a matched calibration-data budget. To check whether this choice accounts for our advantage, Table 9 additionally compares QIG with 128 pairs against Ours with 64.

Despite using half as many calibration pairs, Ours achieves the higher five-task average in five of the six model–precision settings; Qwen2-VL under W3A16 is the exception. These results support calibration-data efficiency without implying that Ours wins on every individual task.

Table 9: Five-task comparison between QIG with 128 calibration pairs and Ours with 64 pairs. Avg. is the unweighted five-task mean.

| Model | Precision | Method | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. |
|---|---|---|---|---|---|---|---|---|
| InternVL2 | FP16 | FP | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 |
| | W3A16 | QIG ($n = 128$) | 45.44 | 59.05 | **96.33** | 79.76 | **80.05** | 72.13 |
| | | Ours ($n = 64$) | **47.44** | **59.77** | 96.18 | **80.12** | 79.63 | **72.63** |
| | W4A8 | QIG ($n = 128$) | 45.56 | 56.19 | **96.73** | **79.32** | 79.21 | 71.40 |
| | | Ours ($n = 64$) | **46.33** | **58.01** | 96.58 | 78.44 | **79.83** | **71.84** |
| LLaVA-OV | FP16 | FP | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 |
| | W3A16 | QIG ($n = 128$) | 45.00 | 60.21 | **95.09** | **77.44** | **79.05** | 71.36 |
| | | Ours ($n = 64$) | **46.11** | **60.28** | 94.94 | 77.20 | 79.02 | **71.51** |
| | W4A8 | QIG ($n = 128$) | 43.22 | 53.96 | **94.25** | **75.12** | 77.88 | 68.89 |
| | | Ours ($n = 64$) | **45.22** | **55.78** | 94.00 | 75.08 | **78.50** | **69.72** |
| Qwen2-VL | FP16 | FP | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 |
| | W3A16 | QIG ($n = 128$) | 47.44 | **67.44** | **83.19** | **78.04** | 77.27 | **70.68** |
| | | Ours ($n = 64$) | **48.00** | 66.86 | 82.30 | 77.84 | **77.82** | 70.56 |
| | W4A8 | QIG ($n = 128$) | 45.33 | 59.78 | **81.56** | 76.68 | **76.91** | 68.05 |
| | | Ours ($n = 64$) | **47.00** | **62.93** | 80.81 | **77.12** | 76.33 | **68.84** |

### C.5 BLOCK-SENSITIVITY ALLOCATION CONTROLS

We compare two block-sensitivity controls under W4A8 using the same 64 COCO calibration pairs and CWE search setup. Both rank layers by unsigned, unweighted block reconstruction error, assigning larger grids to more sensitive layers. Fixed-grid Block independently assigns grids $\{16, 18, 20, 22, 24\}$ to five ordered sensitivity groups, placing residual layers in the middle group to retain an exact mean of 20. It does not use our grid distribution or QE-to-budget rule. Ours-grid-matched Block instead reassigns the exact grid-size multiset of Ours according to block sensitivity, preserving its total search budget. Ours uses target mean grids of 21, 18, and 19 for InternVL2, LLaVA-OV, and Qwen2-VL, respectively; the independent control therefore has a comparable, but not identical, budget.

Table 10 shows that Ours achieves the highest five-task average against both controls on all three models. Relative to Fixed-grid Block, MMMU improves on every model, by up to 2.55 points; VizWiz improves on two models, by up to 3.43 points. The controls answer complementary questions: Fixed-grid Block compares the complete allocation procedure with an independent block-based policy, whereas Ours-grid-matched Block isolates layer assignment at the same grid distribution. They are not successive ablations, and inheriting our grid distribution alone does not consistently improve block-based allocation.

Table 10: W4A8 per-task comparison with independent fixed-grid Block sensitivity. FP16 is the reference; bold denotes the best quantized result within each model.

| Model | Method | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. |
|---|---|---|---|---|---|---|---|
| InternVL2 | FP16 | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 |
| | Fixed-grid Block | 43.78 | 55.93 | **96.63** | 78.48 | 78.89 | 70.74 |
| | Ours-grid-matched Block | 44.00 | 57.47 | 96.33 | **78.80** | 79.57 | 71.23 |
| | Ours | **46.33** | **58.01** | 96.58 | 78.44 | **79.83** | **71.84** |
| LLaVA-OV | FP16 | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 |
| | Fixed-grid Block | 44.22 | **56.19** | **94.79** | 74.40 | **78.50** | 69.62 |
| | Ours-grid-matched Block | **45.22** | 54.56 | 94.40 | 74.64 | 78.04 | 69.37 |
| | Ours | **45.22** | 55.78 | 94.00 | **75.08** | **78.50** | **69.72** |
| Qwen2-VL | FP16 | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 |
| | Fixed-grid Block | 46.44 | 59.50 | **80.81** | 76.76 | **77.33** | 68.17 |
| | Ours-grid-matched Block | 45.56 | 60.24 | 80.52 | 76.68 | 76.68 | 67.94 |
| | Ours | **47.00** | **62.93** | **80.81** | **77.12** | 76.33 | **68.84** |

### C.6 Computational cost

Table 11 reports W4A8 calibration wall-clock time on COCO-64, excluding model loading and evaluation. Ours runs three quantization effect analyses concurrently on three NVIDIA RTX A6000 GPUs (48GB each); each CWE search run and both baselines use one GPU. Thus, this comparison measures wall-clock time under different GPU allocations, rather than matched compute.

Ours has the shortest CWE search time on all three models, which offsets much of the additional effect-analysis time. Its total elapsed time is lower than QIG on all three models and lower than MBQ on LLaVA-OV, while MBQ remains faster on InternVL2 and Qwen2-VL. The benefit is therefore comparable calibration turnaround under the stated hardware allocation, not a claim of uniformly lower computational cost. Calibration is performed once and introduces no inference overhead.

Table 11: W4A8 calibration time on NVIDIA RTX A6000 48GB GPUs using 64 COCO samples; model loading and evaluation are excluded. Times are wall-clock times.

| Model | Method | Analysis | CWE search | Total |
|---|---|---|---|---|
| InternVL2 | MBQ | 0m 56s | 24m 12s | 25m 08s |
| | QIG | 4m 24s | 23m 30s | 27m 54s |
| | Ours | 21m 02s | 6m 28s | 27m 30s |
| LLaVA-OV | MBQ | 1m 10s | 35m 02s | 36m 12s |
| | QIG | 7m 04s | 35m 31s | 42m 35s |
| | Ours | 32m 11s | 1m 55s | 34m 06s |
| Qwen2-VL | MBQ | 1m 01s | 26m 21s | 27m 22s |
| | QIG | 4m 49s | 26m 57s | 31m 46s |
| | Ours | 18m 59s | 8m 41s | 27m 40s |

# D Quantization Effect and Prediction Analyses

## D.1 Stability metric definitions

To test whether the allocation signal depends on a particular calibration subset, Table 5 compares three disjoint calibration subsets with an independent held-out subset. Components are ranked by calibration mean relative absolute QE. QE mass is the captured fraction of absolute QE. Three-subset agreement requires matching signs across all calibration subsets; held-out agreement compares their majority sign with the held-out sign. Beneficial held-out agreement restricts this comparison to calibration-beneficial effects. Ranking uses calibration data only, so the held-out subset tests whether the selected effects retain their direction on unseen samples.

Agreement is strongest for high-magnitude effects, including beneficial ones, supporting both refinement and coarsening decisions. This analysis tests the stability of the allocation signal, rather than the variance of final downstream performance across calibration seeds.

## D.2 Controlled search granularity

To examine whether coarser fitting can help independently of bit-width allocation, we use W4A8 and COCO-64 and vary only the CWE grids that Ours coarsens below the 20-point reference, keeping precision and all remaining grids fixed. Fine replaces these counts with 25 for InternVL2 and 23 for LLaVA-OV and Qwen2-VL; Mid uses 20, and Coarse retains the original Ours counts. All counts uniformly discretize the same exponent interval $[0, 1]$, so their interior candidates need not be nested. After recalibration, we compare aligned per-channel scale vectors.

Table 12 shows that Coarse achieves the highest five-task average on all three models despite highly similar Fine–Coarse scale vectors. The intermediate setting is not consistently better than Fine, so the result does not suggest a monotonic benefit from reducing candidates. Instead, it supports selectively restricting fitting capacity according to the QE-guided allocation.

Table 12: W4A8 per-task results under controlled search granularity. Fine–Coarse scale-vector cosine similarity is 0.9895 / 0.9955 / 0.9902 for InternVL2 / LLaVA-OV / Qwen2-VL, respectively. Bold denotes the best quantized result.

| Model | Setting | MMMU | VizWiz | ScienceQA | ChartQA | AI2D | Avg. |
|---|---|---|---|---|---|---|---|
| InternVL2 | FP16 | 48.11 | 60.67 | 97.07 | 82.48 | 82.35 | 74.14 |
| | Fine | 45.11 | 57.39 | **96.63** | **78.96** | 79.15 | 71.45 |
| | Mid | 45.22 | 57.26 | **96.63** | 78.52 | 78.14 | 71.15 |
| | Coarse (Ours) | **46.33** | **58.01** | 96.58 | 78.44 | **79.83** | **71.84** |
| LLaVA-OV | FP16 | 46.56 | 58.71 | 95.84 | 80.00 | 81.28 | 72.48 |
| | Fine | 43.67 | 55.36 | **94.35** | **75.08** | 78.11 | 69.31 |
| | Mid | 42.22 | **55.93** | 93.51 | 75.04 | 78.24 | 68.99 |
| | Coarse (Ours) | **45.22** | 55.78 | 94.00 | **75.08** | **78.50** | **69.72** |
| Qwen2-VL | FP16 | 50.56 | 68.91 | 84.88 | 81.56 | 80.08 | 73.20 |
| | Fine | 46.89 | 61.58 | 80.42 | 77.00 | 76.59 | 68.50 |
| | Mid | 46.89 | 61.64 | 80.61 | 76.52 | **77.20** | 68.57 |
| | Coarse (Ours) | **47.00** | **62.93** | **80.81** | **77.12** | 76.33 | **68.84** |

## D.3 Full-set FP-answer transitions

To determine whether the gains primarily come from correcting FP mistakes or avoiding new errors, we evaluate all paired FP–PTQ predictions on MMMU (900), VizWiz (4,319), ScienceQA (2,017), ChartQA (2,500), and AI2D (3,088): 12,824 examples per model and precision. Correctness follows each benchmark's evaluator, with positive VQA credit counted as correct for VizWiz. Transition rates use their corresponding FP-wrong or FP-correct subset; fixes/new errors is a ratio of counts.

Table 13 shows that Ours consistently reduces new errors and improves retention and the fixes/new-errors ratio, while its correction rate is not uniformly highest. The aggregate gains therefore pri-

marily reflect better preservation of correct FP behavior and fewer newly introduced errors, complemented by corrections of some FP mistakes. This full-set analysis provides context for the qualitative examples rather than relying on selected corrections alone.

Table 13: Full-set FP-answer transitions under W3A16 and W4A8. Rates are conditional on the corresponding FP-correct or FP-wrong subset.

| Model | Precision | Method | FP wrong → correct (%) | FP correct → wrong (%) | FP correct → correct (%) | Fixes/ new errors |
|---|---|---|---|---|---|---|
| InternVL2 | W3A16 | MBQ | **15.92** | 6.99 | 93.01 | 0.63 |
| | | QIG | 14.27 | 6.70 | 93.30 | 0.59 |
| | | Ours | 15.38 | **6.23** | **93.77** | **0.68** |
| | W4A8 | MBQ | **19.13** | 9.00 | 91.00 | 0.59 |
| | | QIG | 16.53 | 8.98 | 91.02 | 0.51 |
| | | Ours | 19.02 | **8.32** | **91.68** | **0.63** |
| LLaVA-OV | W3A16 | MBQ | 22.52 | 8.59 | 91.41 | 0.80 |
| | | QIG | 21.78 | 8.33 | 91.67 | 0.80 |
| | | Ours | **23.05** | **7.94** | **92.06** | **0.88** |
| | W4A8 | MBQ | **23.02** | 12.81 | 87.19 | 0.55 |
| | | QIG | 20.01 | 12.26 | 87.74 | 0.50 |
| | | Ours | 22.19 | **10.70** | **89.30** | **0.63** |
| Qwen2-VL | W3A16 | MBQ | 20.28 | 9.74 | 90.26 | 0.55 |
| | | QIG | 19.76 | 9.50 | 90.50 | 0.55 |
| | | Ours | **20.73** | **8.70** | **91.30** | **0.63** |
| | W4A8 | MBQ | 19.99 | 13.44 | 86.56 | 0.40 |
| | | QIG | **20.58** | 12.22 | 87.78 | 0.45 |
| | | Ours | 19.80 | **11.02** | **88.98** | **0.48** |

### D.4 QE UNDER JOINT QUANTIZATION

Isolated component effects may change when multiple components are quantized together. To examine whether their direction remains informative in that setting, we compare isolated QE with restoration-based QE obtained by restoring one layer-component to FP in an otherwise quantized model. We define restoration-based QE as $\mathcal{L}_{\text{restore}} - \mathcal{L}_{\text{joint}}$, the calibration loss after restoration minus that of the jointly quantized model; positive values indicate that retaining the quantized component lowers loss. Both use COCO-64 with allocation and CWE search disabled. QE mass uses absolute isolated QE; beneficial agreement considers only beneficial isolated effects.

Table 14 reports sign agreement between isolated and restoration-based QE, with stronger agreement for high-magnitude effects, including beneficial effects. This supports using isolated measurements to identify influential components even in jointly quantized models. The imperfect agreement also indicates that isolated effects do not fully capture interactions among quantized components.

Table 14: Agreement between isolated QE and restoration-based QE in jointly quantized W3A16 and W4A8 models. Parentheses give agreement counts.

| QE-ranked subset | Count | QE mass | Joint sign agreement | Beneficial joint agreement |
|---|---|---|---|---|
| Top 50% | 176 | 91.50% | 68.20% (120/176) | 75.60% (34/45) |
| Top 25% | 88 | 75.80% | 81.80% (72/88) | 95.50% (21/22) |
| Top 10% | 38 | 58.20% | **89.50% (34/38)** | **100% (11/11)** |

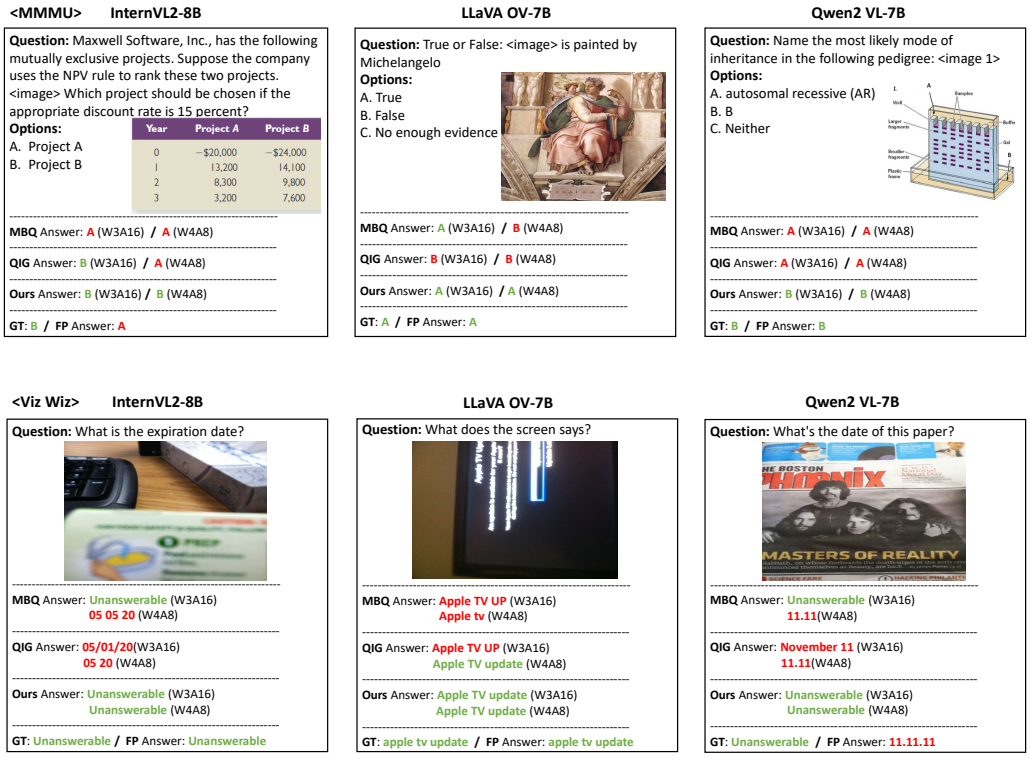



Figure 10: Additional qualitative comparison on MMMU and VizWiz.

## E Additional Qualitative Results

Figure 10 and Figure 11 provide additional qualitative comparisons on five vision-language benchmarks (Yue et al., 2024; Gurari et al., 2018; Lu et al., 2022; Masry et al., 2022; Kembhavi et al., 2017) across three LVLMs (Chen et al., 2024b; Li et al., 2024; Wang et al., 2024b).

Consistent with the full-set transition analysis, baseline PTQ methods (MBQ (Li et al., 2025) and QIG (Xiang et al., 2026)) can replicate incorrect FP predictions or introduce specific but erroneous outputs under low-bit settings, particularly in W4A8. In contrast, our method more reliably preserves correct FP predictions and introduces fewer new errors; selected examples also show corrections of FP errors. On MMMU (Yue et al., 2024), the examples illustrate that selective coarse fitting need not reproduce an incorrect FP answer and can recover the correct choice. On VizWiz (Gurari et al., 2018), where inputs are often ambiguous or low-quality, baseline methods frequently generate incorrect specific answers, while our method produces more reliable outputs, including correctly identifying unanswerable cases. Figure 11 shows similar trends on ScienceQA (Lu et al., 2022), ChartQA (Masry et al., 2022), and AI2D (Kembhavi et al., 2017). Across these structured reasoning and diagram understanding tasks, our method primarily benefits from preserving valid FP behavior while using regularization-like constrained fitting to reduce calibration-specific overfitting.

Overall, these results suggest that our approach better balances precise fitting and regularization-like capacity control, preserving valid FP behavior and improving generalization under aggressive quantization.

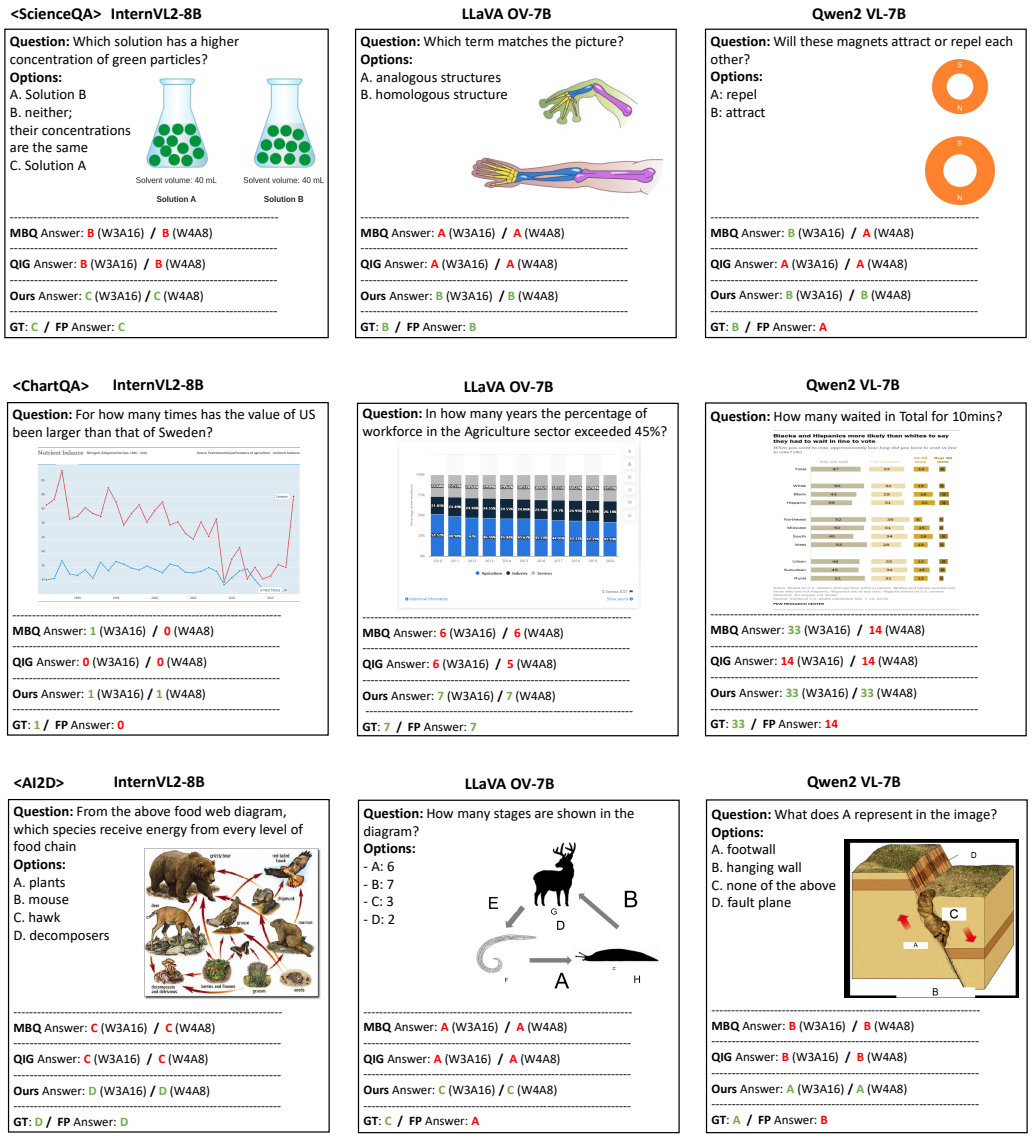



Figure 11: Additional qualitative comparison on ScienceQA, ChartQA, and AI2D.