Computational Depth Predicts Quantization Sensitivity in Multimodal Models

Dongnan Gui
University of Science and Technology of China
dongnangui@mail.ustc.edu.cn

Ruizhe Wang
University of Science and Technology of China
wangruizhe@mail.ustc.edu.cn

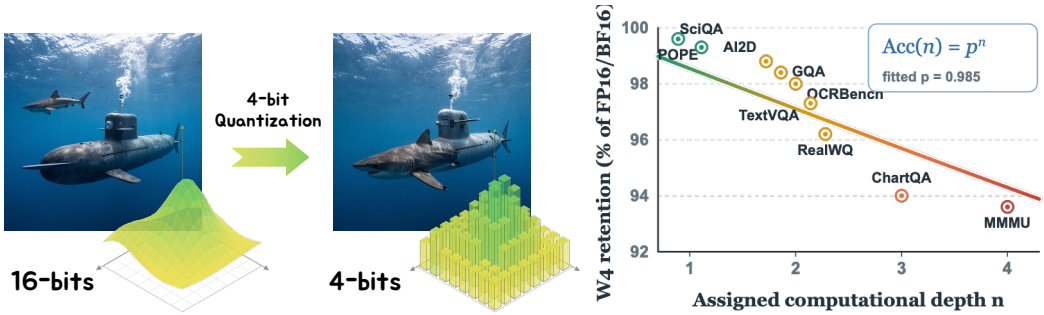

Figure 1: **Quantization harms multimodal generation and understanding, while computational depth predicts the resulting damage.** *Left:* When prompted to generate an underwater shark and submarine, the BF16 model interprets the request correctly, whereas INT4 misinterprets it. *Right:* Retained accuracy summarized by assigned task depth on nine VLM benchmarks.

Abstract

Post-training quantization degrades multimodal model capabilities unevenly, yet practitioners lack a principled way to predict which capabilities will collapse. We propose the *Computational Depth Principle* (CDP): end-to-end fidelity under quantization follows $\hat{p}^n$, where n counts sequential precision-dependent operations and $\hat{p}$ is a per-architecture survival rate. Across four VLM families and diverse benchmarks spanning image understanding, image generation, video understanding, and video generation, we show that deeper reasoning tasks consistently degrade more steeply and that a single shallow probe predicts the full degradation ranking on held-out architectures. The same exponential form extends to image generation, where denoising step count plays the role of depth, and to video understanding, where longer temporal reasoning chains amplify losses. For video generation, spatial quality degrades more than temporal coherence, indicating that the dominant compounding axis is denoising depth rather than cross-frame coupling. A complementary distortion-to-information (D/I) flip ratio diagnoses quantization method effects, revealing that certain methods preserve near-baseline quality while others produce catastrophic collapses on specific architecture pairings that aggregate accuracy alone cannot detect. Module ablation localizes the sensitivity to the language backbone rather than the vision encoder. Together, depth and D/I convert full-suite evaluation into a two-probe screening protocol that surfaces the highest-risk capability gaps first.

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

1 Introduction

Multimodal models are increasingly deployed under memory constraints that demand aggressive quantization, yet standard evaluations report only aggregate quality drops that mask capability-specific damage. The same compressed VLM can preserve object recognition while failing at chart reasoning [50, 27, 52], and the same quantized diffusion model can maintain per-frame quality while losing temporal coherence. Prior work documents capability-dependent collapse for text LLMs [33] and catalogs benchmark-level effects, but does not provide a quantitative model that orders capabilities by predicted quantization sensitivity, transfers across architectures, or generalises to diffusion.

We propose the *Computational Depth* model (Fig. 1). End-to-end fidelity under quantization is approximated by $\hat{p}^n$, where n is a hand-assigned count of sequential precision-critical operations and $\hat{p}<1$ is a per-architecture survival rate fit from a depth-1 probe. The model is one-parameter per architecture and is intended as a coarse capability ordering and forecasting tool, not as a derived theoretical bound. We validate it primarily on VLM understanding, where depth varies most richly across benchmarks, and apply it to generation as a probe of cross-modal generalisation.

(1) **Core validation: understanding.** Across four VLM architectures and ten image-understanding benchmarks, deeper reasoning tasks degrade more under W4 on three of four architectures. Held-out leave-one-architecture-out forecasting reaches a mean per-split Spearman $\bar{\rho}=+0.54$, while the pooled cross-architecture Spearman over all $n=20$ cells is $+0.20$, so depth is a within-architecture ordering signal rather than a cross-architecture forecaster.

(2) **Cross-modal extension: generation.** The depth principle generalizes partially. Few-step distilled diffusion models retain quality at W2 group size 16 while full-step models drop further, consistent with the p^n form applied to denoising chains, with the full sweep tabulated in App. D.2. On the three video DiTs we evaluated (Wan2.1-1.3B [48], CogVideoX-2B [57], LTX-Video-2B [13]), spatial imaging quality drops more than temporal flickering at W4, contrary to a naive cross-frame compounding prediction.

(3) **Diagnostic toolkit.** A distortion-to-information ratio (D/I) separates balanced noise from coherent bias, providing a small-probe gate for safe architecture-method pairings before full evaluation.

We restrict the empirical study to weight-only PTQ on VLMs up to 9B and generation models up to 13B parameters. This range covers the deployment-relevant band where W4 is a memory win on a single consumer or workstation GPU. Larger checkpoints require multi-GPU inference even at W4 and are left to future work. CDP predicts that the depth-tier ordering should persist at 72B if the per-stage success rate $\hat{p}$ continues to rise with scale, and the limitations section returns to this prediction as an explicit falsification target.

2 Analytical Framework

The framework rests on a single error model. When a pipeline chains n precision-sensitive stages, each surviving with probability p, end-to-end fidelity compounds to p^n. This model instantiates differently across modalities but the compounding law is invariant. Figure 2 summarizes the full workflow: assign n_T from the task structure, estimate $\hat{p}$ and γ with two compact probes, forecast retention, and choose among W4, mixed precision, or a different quantization method. Depth therefore predicts vulnerability, while D/I diagnoses whether the observed damage is benign noise or coherent bias.

2.1 The Computational Depth Principle

Depth, not difficulty, is what the bit-width must track, and the definition is modality-general.

- **Understanding tasks.** n_T counts sequential reasoning stages.
- **Generation tasks.** n_T counts denoising steps. For video generation, denoising depth remains the dominant compounding axis, as spatial quality degrades more than temporal coherence at W4, indicating that cross-frame coupling does not add a separate multiplicative factor (§4.3).

A task's depth n_T is hand-assigned by the authors from the public task specification of each benchmark before any quantized number is read. The rubric counts sequential operations along perception,

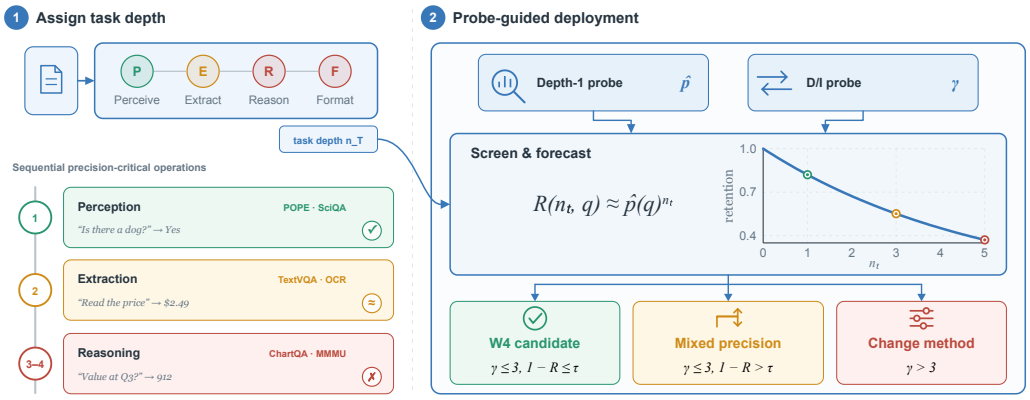

Figure 2: **Computational Depth Principle (CDP) workflow.** *Left:* assign task depth n_T by counting sequential precision-critical operations, from shallow perception through extraction to multi-step reasoning. *Right:* estimate the architecture-specific per-stage retention $\hat{p}$ from a depth-1 probe and the distortion-to-information ratio γ from output flips, then forecast retention $R(n_T, q) \approx \hat{p}(q)^{n_T}$ at bit-width q. The forecast and D/I gate route a deployment to a W4 candidate, mixed precision when predicted loss exceeds τ, or a different quantization method when $\gamma > 3$.

Table 1: **Computational depth labels.** n_T is assigned from the task specification before reading quantized scores. Understanding depth counts reasoning stages, and generation depth counts denoising steps.

Benchmark	n_T	**Primary computation**
POPE	1	Single object presence judgment
SciQA	1–2	Recognition + knowledge recall
TextVQA	2	OCR extraction + answer selection
OCRBench	2	Character recognition + structuring
AI2D	2–3	Diagram parsing + spatial reasoning
RealWorldQA	2–3	Scene understanding + judgment
GQA	2–3	Compositional visual QA
MME	2	Comprehensive perception + reasoning
ChartQA	3–4	Value extraction + comparison + arithmetic
MMMU	3–5	Multi-step domain reasoning
Video-MME	2	Per-frame perception + basic temporal
MLVU	3	Long-context temporal reasoning
T-LPIPS (gen)	T	State across T denoising steps
CLIP (gen)	1	Per-frame spatial quality

extraction, reasoning, and answer formatting; half-step labels refine the benchmark-level ordering (App. B). Table 1 lists the resulting labels and their rationales.

The depth labels are assigned from the public task specification of each benchmark before reading any quantized result. Although the quantization outcomes were known to us at paper-writing time, the labels follow a principled rubric grounded in the task's reasoning structure, and the full rubric and per-benchmark rationale are provided in App. B.1. When stage errors are conditionally independent, the success probability factorizes as $\prod_k p_k \leq p^n$, giving the upper bound

$$\Pr[\text{correct}] \ \leq \ p^n, \tag{1}$$

where p is the per-stage success probability from a depth-1 probe and $n{=}n_T$ (Fig. 3). We use the geometric form $1 - \hat{p}^{n_T}$ as a fitted predictor of expected degradation rather than as a tight bound. In our data the bound is loose for the most robust architectures, where stage errors anti-correlate, and approximately attained on the most depth-sensitive ones.

Proposition 1 (Depth-dependent error amplification)**.** *For n sequential precision-dependent stages with per-stage success $p_k \leq p < 1$, $\Pr[\text{correct}] \leq p^n$, and the depth-$n$ vs. depth-1 gap $p(1{-}p^{n-1})$ is strictly increasing in n.*

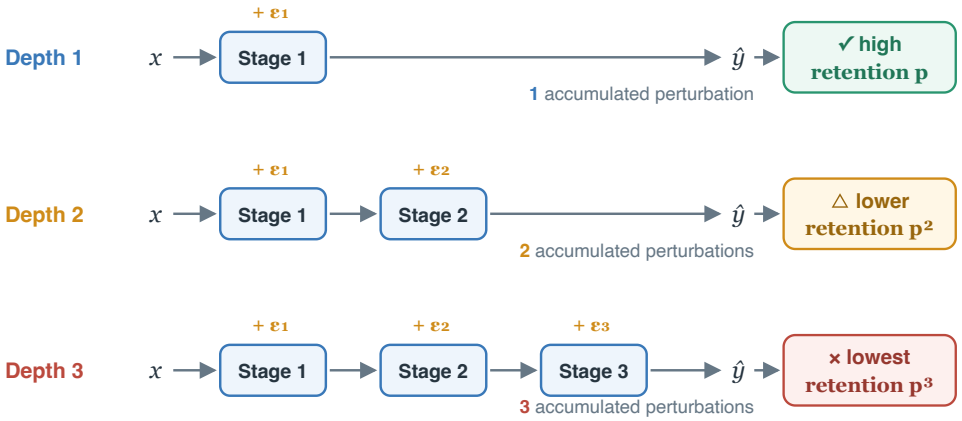

Figure 3: **Quantization noise compounds with depth.** The same small per-stage 4-bit perturbation compounds over depth-1, depth-2, and depth-3 pipelines, progressively reducing end-to-end fidelity according to p^n.

Two consequences follow. First, a single depth-1 probe predicts behavior at every other depth at the same bit-width, replacing full accuracy sweeps. Second, the depth-risk score orders tasks by sensitivity, and the transport is across depths, not bit-widths, since $\hat{p}$ is re-fitted per bit-width as detailed in App. B.2. Eq. 2 states both formally.

$$D_{\text{risk}}(T, q) = n_T \log\big(1/\hat{p}(q)\big), \qquad b^\star(T) = \min\big\{b : 1 - \hat{p}(b)^{n_T} \le \tau\big\}. \tag{2}$$

2.2 Distortion-Information Decomposition

The depth principle predicts which tasks are in danger, but two quantization methods at the same bit-width can still produce different outcomes. A per-sample audit compares each quantized prediction against FP16/BF16. Let D count samples FP16/BF16 gets right that the quantized model flips to wrong, and I count flips in the opposite direction,

$$\gamma \;=\; D \,/\, \max(I, 1). \tag{3}$$

A ratio $\gamma{\approx}1$ indicates balanced noise, characteristic of zero-mean methods such as BnB-NF4. $\gamma{>}3$ indicates an asymmetric flip pattern produced by calibration-based methods whose drift injects systematic error. An expanded audit of 351 cells yields ROC-AUC 0.970 and held-out-architecture F1 0.814, supporting $\tau_\gamma{=}3$ as the default gate (App. C.2).

Depth and D/I are complementary. Depth ranks capabilities at a fixed method, and D/I ranks architecture-method pairings at a fixed bit-width. For understanding tasks with discrete outputs, D/I uses flip counts directly. For generation, the analogous diagnostic is output-distribution distance to FP16/BF16.

3 Related Work

3.1 Quantization Methods for Large Models

Post-training quantization has evolved through Hessian-guided rounding [12, 58], activation-aware scaling [32], weight activation smoothing [54], information-optimal data types [7], outlier isolation [6], incoherence processing [4], additive codebooks [8], squeezed channel grouping [21], rotation-based methods [31], and learnable formulations [45, 37]. KV-cache quantization [35, 17] addresses a complementary memory bottleneck in long-context inference. Vision-language extensions target modality-specific outliers via activation variance [50], modality-balanced calibration [27], Hadamard-rotation control [59], instruction-tuned group-wise scales [55], and saliency-driven 2-bit allocation [56, 51]. Diffusion methods add timestep-aware reconstruction [28, 14], dynamic bit budgets [46], transformer variants [18, 63, 26], and video-DiT extensions [30, 10]. These methods price sensitivity at the layer or tensor level via Hessian scores [12] or activation-outlier statistics [32]

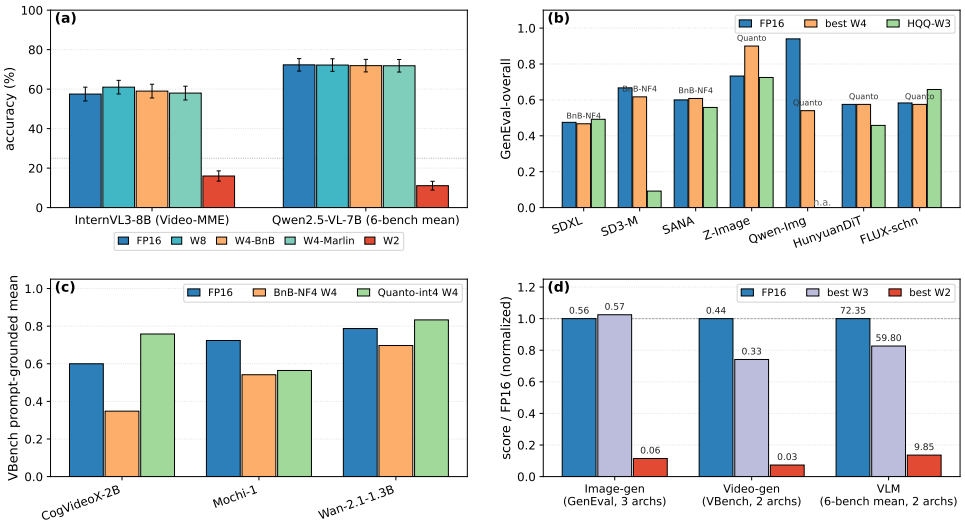

Figure 4: **Quantization sensitivity across four scenarios.** (a) VLM bit-depth ladder: W8 and W4 are within noise of FP16/BF16 while W2 collapses. (b) Image-generation GenEval-overall across seven architectures using the best-performing W4 quantizer per architecture, with the chosen quantizer (BnB-NF4 or Quanto) annotated, and HQQ-W3 where available. (c) Video-generation VBench prompt-grounded mean. (d) W2 floor across three modalities, full sweep in App. D.2.

extracted after quantization. No method forecasts which downstream capabilities will collapse once deployed.

3.2 Capability-Aware Evaluation

Modern evaluation suites such as MMBench [34], MMMU [60], SEED-Bench [24], and lmms-eval [61] organize tasks by capability, with video coverage extending to temporal reasoning [38], perception [25], and clue-grounded inference [5]. Yet quantization studies crush per-capability damage into aggregate scores such as mean accuracy, FID [15], or CLIP score [43], so single-step POPE and multi-step ChartQA [39] share one bar despite opposite reasoning depths. "Quantization Hurts Reasoning?" [33] documented capability-dependent collapse in text LLMs but left it as an empirical observation. Scaling laws [19] compress depth-versus-width trade-offs into scalar quality scores. Difficulty grading in MMMU [60] ranks items by author-assigned expertise rather than structural depth. Token pruning [62], distillation [16], neural architecture search [9], joint pruning-plus-quantization [20], and lottery-ticket sparsity [11] trade capabilities implicitly without per-capability tracking. Classical error-propagation theory [49, 44, 40] and the statistical foundations of quantization noise [53] bound layer-wise perturbation but do not capture sequential inference, where errors compound across reasoning steps.

No prior work provides a quantitative model that orders capabilities by predicted quantization sensitivity, transfers across architectures, or generalises to diffusion. We recast quantization sensitivity as a forecasting problem: given a task's structural depth, predict its degradation on an unseen architecture before running a quantized evaluation. CDP provides the predictor and the D/I diagnostic provides the method-level gate, while generation scenarios serve as cross-modal tests of whether the compounding form holds outside the VLM regime.

4 Experiments

We validate CDP primarily on VLM understanding, where reasoning depth varies richly across benchmarks, and extend to generation as supporting evidence. Four architecture families and five quantization methods span the modality task space (Fig. 4).

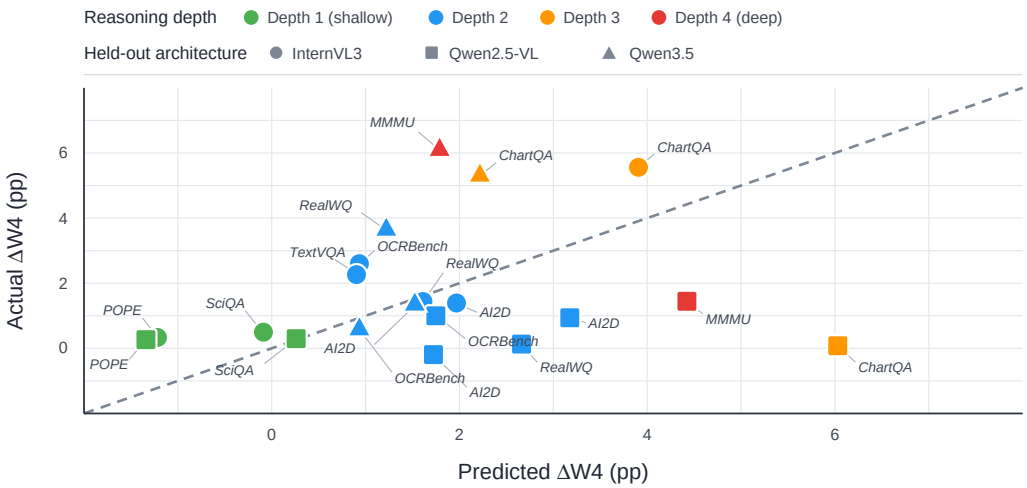

Figure 5: **Leave-one-architecture-out validation.** X-axis: W4 degradation predicted by the depth-only model trained on the remaining architectures. Y-axis: actual observed W4-NF4 accuracy drop. Points near the diagonal indicate good prediction. Color encodes reasoning depth and marker shape encodes the held-out architecture.

4.1 Protocol

The primary testbed covers seven VLMs spanning 1 to 9B parameters, including InternVL3 [64], Qwen2.5-VL [2], Qwen3.5 [42], and Phi-3.5-Vision [1], evaluated on ten image-understanding benchmarks spanning depth levels 1 to 4, from single-step perception (POPE [29], SciQA [36]) through extraction to multi-step reasoning (ChartQA [39], MMMU [60]), plus four video benchmarks. The full-precision baseline uses the model's native format, either FP16 or BF16 depending on architecture, denoted **FP16/BF16** throughout. Each VLM is compressed via W8-INT8, W4-NF4 [7], AWQ-W4 [32], and GPTQ-W4 [12]. Generation coverage adds image-diffusion models (SDXL [41], FLUX [3], SANA, HunyuanDiT [22]) and video-diffusion models (CogVideoX [57], LTX-Video [13], Wan2.1 [48]) to extend CDP to denoising-step and temporal-coupling depth axes. Full model checkpoints, benchmark subsets, and evaluation scripts are in App. A.

4.2 Image Understanding and Generation

Image understanding. Sensitivity rises monotonically with task depth on Qwen3.5, InternVL3, and Phi-3.5 (Tab. 2, Fig. 4a). Shallow perception benchmarks absorb W4 with negligible loss, while deep multi-step reasoning tasks amplify per-stage noise into measurable accuracy gaps. Qwen2.5-VL is an exception: its W4 drops on ChartQA (depth-3) and several depth-2 cells are within ± 0.5pp, consistent with a per-stage rate $\hat{p}$ close enough to unity that the predicted multiplicative gap falls within the measurement noise.

Reducing precision from W8 to W4 increases the absolute degradation magnitude but preserves the depth-tier ordering on the three depth-sensitive architectures. Aggregating across these three families, the mean W4 drop for depth-1 benchmarks is -0.3%, for depth-2 it is -1.7%, and for depth-3/4 it is -4.5%, following a roughly geometric progression consistent with the p^n functional form. Full-scale validation over 23 cells gives shallow, mid, and deep tier losses of 0.41, 1.48, and 4.15 pp, respectively (App. B). All four architectures retain most FP16/BF16 performance under the best-matched W4 recipe, and the depth-tier ordering is stable across model scales, as Phi-3.5-Vision at 4B exhibits the same pattern as the 7 to 9B models.

Image generation. For diffusion models, depth counts denoising steps. Most architectures preserve generation quality at W4 (Fig. 4b). Under W2, the step-count axis separates outcomes: few-step distilled models stay close to FP16/BF16 quality while full-step models drop further, consistent with p^n applied to denoising chains, with the full sweep in App. D.2. Among the twelve image-generation architectures swept at W2 group size 16, five escape the precision floor while seven

Table 2: **Capability sensitivity under BnB-NF4 W4 quantization.** Changes are relative percentages. Shallow benchmarks (depth-1, green) absorb W4 with negligible loss, while deep reasoning benchmarks (red) carry the largest drops on three of four architectures. * Interpolated. ** 200-sample subset.

		Qwen3.5-9B			InternVL3-8B			Qwen2.5-VL-7B			Phi-3.5-4B		
Benchmark	n	FP16/BF16	ΔW8	ΔW4	FP16/BF16	ΔW8	ΔW4	FP16/BF16	ΔW8	ΔW4	FP16/BF16	ΔW8	ΔW4
POPE	1	.884	-0.0	+0.2	.913	-0.4	-0.4	.877	-0.1	-0.3	.835	-0.5	+0.0
SciQA	1	.932	-0.4	-1.3	.980	+0.4	+0.2	.885	-0.1	-0.3	.905	-1.5	-2.0
TextVQA	2	.677	-0.8	-3.1	.793	-0.7	-2.9	.827	+0.2	+0.2	.635	+2.0	+5.0
AI2D	2	.839	-1.4	-1.6	.837	-0.1	-1.7	.827	+0.0	-1.1	.805	+0.0	+0.0
ChartQA	3	.776	-2.6	**-6.8**	.825	+3.8	**-6.7**	.830	+0.3	-0.1	.670	-0.5	-6.0
OCRBench	2	.723	+0.4	-0.8	.820	+2.0	-3.2	.839	-0.2	-1.2	.770	+0.5	-1.5
RealWQ	2	.672	-0.8	**-5.4**	.685	-1.7	-2.1	.693	-0.6	-0.2	.615	-0.5	-1.0
MMMU	4	.520	-2.6	-4.9	.540	+0.0	-1.0	.511	+0.7	**-5.9**	.475	+3.0	-3.0
GQA	2	.608	-0.1	-1.1	.627	-0.5*	-1.4	.607	-0.0	-0.5	.600	-4.0	-6.0
MME	2	1708	+0.1	-3.8	1717	-0.4*	-0.0	1687	-0.5	-1.3	1473**	+0.7	-1.4

Table 3: **Video generation under W4-NF4 quantization.** Spatial quality degrades comparably to or more than temporal flickering at W4, and warping error varies across models.

Model	**Img. qual. FP16/BF16**	**ΔW4**	**Temp. flick. FP16/BF16**	**ΔW4**	**Warp. Err. FP16/BF16**	**ΔW4**
CogVideoX-2B	0.635	−2.5pp	0.955	−1.3pp	0.043	+28%
LTX-Video-2B	0.479	−10.0pp	0.982	+0.7pp	0.021	−39%
Wan2.1-1.3B	0.509	−7.4pp	0.974	+0.8pp	0.038	−21%

collapse. Three of five survivors are distilled, but distillation is not sufficient: SD3.5-Large-Turbo is distilled yet collapses, indicating that architecture-intrinsic properties interact with the step-count reduction. Varying only step count within non-distilled SDXL confirms this pattern: paired FP16–W4 image MSE rises monotonically from 0.00465 at 8 steps to 0.01133 at 50 steps, with fixed prompts and seed (App. D.1).

4.3 Video Understanding and Generation

Video understanding. CDP predicts that benchmarks requiring longer temporal reasoning chains, specifically MLVU, should degrade more than shorter-clip benchmarks like Video-MME. InternVL3 confirms this ordering: MLVU drops -6.0 pp vs. Video-MME -2.3 pp, consistent with multiplicative compounding along the temporal depth axis, as detailed in App. C.1. On the other three architectures the W4 drops on both benchmarks are within ± 3 pp, mirroring the same pattern observed in image understanding, where Qwen2.5-VL showed negligible degradation. The consistent explanation is that per-stage success rates near unity suppress the multiplicative signal below measurement noise, so the depth ordering emerges clearly only on architectures where $\hat{p}$ is low enough for the gap $p(1-p^{n-1})$ to exceed the per-cell sampling error.

Video generation. CDP predicts that denoising depth is the primary compounding axis in video generation. The data on three video DiTs confirm this: imaging quality drops by 2.5 to 10 pp at W4 while temporal flickering moves by less than 1.3pp, and warping error varies in opposite directions across models, as shown in Tab. 3. This establishes that spatial quality, which compounds over denoising steps, is consistently more vulnerable than temporal coherence, supporting the conclusion that denoising depth rather than cross-frame coupling dominates quantization sensitivity in video generation.

4.4 Analysis

Cross-architecture validation. Held-out leave-one-architecture-out forecasting reaches a mean per-split Spearman $\bar{\rho}{=}{+}0.54$ over the three held-out architectures, five to eight cells per split, while the pooled cross-architecture Spearman over all $n{=}20$ cells is the weaker $+0.20$ as shown in Fig. 5. On these same cells, depth correlates with observed loss at $\rho{=}{+}0.540$ ($p{=}0.014$); calibrating $\hat{p}$ on a target-architecture shallow probe gives held-out MAE 1.54 pp versus 1.91 pp for transferred slopes

Table 4: **Five-rung precision ladder on InternVL3-8B.** Degradation is small from W8 through W3 but catastrophic at W2, placing the precision cliff between W3 and W2. Scores are on a 200-sample subset. †W4 uses BnB-NF4.

Bench	FP16/BF16	W8	W4 (BnB)	W3 (GPTQ)	W2 (Quanto)
ChartQA	73.5	70.0	77.0†	56.5	2.0
OCRBench	90.5	90.5	79.4†	88.5	0.0
RealWorldQA	75.5	76.5	67.1†	65.0	31.5
AI2D	87.5	85.5	82.3†	64.0	9.0
MMMU (val)	57.0	59.0	58.5	45.0	8.0
MM-Vet	50.5	51.0	48.5	39.5	1.0
Mean (Δ)	72.4	-0.3	≈ -3	-12.7	-63.8

Table 5: **Module ablation (W4-NF4).** VE = vision encoder, Proj = projector, LLM = language model. Tier-1 accuracy is unaffected regardless of which module is quantized.

Model	Benchmark	VE-W4	Proj-W4	LLM-W4
Qwen3.5-9B	POPE	0.865	0.855	0.845
	SciQA	0.950	0.920	0.910
	TextVQA	0.632	0.642	0.631
Qwen2.5-VL-7B	MMMU	0.459	0.459	0.459

(App. B). Within-architecture rankings transfer well across the three depth-sensitive families, but the cross-architecture pooled ranking is dominated by per-cell noise once Qwen2.5-VL is included. This establishes depth as a within-architecture ordering signal rather than a cross-architecture point predictor, with further analysis in App. B.2.

Method effects and precision cliff. At identical bit-width, BnB-NF4 preserves near-FP16/BF16 quality while GPTQ can produce catastrophic collapses on certain architectures, as detailed in App. C.1. The D/I flip ratio (Eq. 3) separates these method effects quantitatively. Inserting a W3 rung between W4 and W2 places the precision cliff between W3 and W2 (Tab. 4): the single step from W3 to W2 produces order-of-magnitude collapses, placing W2 below the usable floor for current VLMs, as documented in App. D.2.

Module ablation and sensitivity localization. Each reasoning stage commits to a discrete token that conditions every later stage, so a per-stage residual that survives at depth 1 enters the KV-cache and accumulates through the chain. The same conditioning structure carries to generation, where each denoising step reads the prior step's corrupted latent. The module ablation in Tab. 5 localizes this sensitivity: quantizing the vision encoder alone produces negligible accuracy change on Tier-1 tasks across both Qwen3.5 and Qwen2.5-VL, while quantizing the LLM backbone produces the largest degradation. This implies that mixed-precision deployments should prioritize language-backbone precision over vision-encoder precision for deep tasks, and that vision-encoder-only quantization is essentially free for shallow perception tasks. At the method level, BnB-NF4 per-layer residuals are approximately zero-mean by construction, so depth-n chains amplify variance without injecting mean drift. Under GPTQ, the calibration-fitted residual carries a non-zero mean that enters softmax as a systematic logit shift, converting additive noise into multiplicative collapse, a mechanism corroborated by the D/I per-sample signature showing asymmetric flip patterns exclusive to GPTQ pairings. The failure surface in Tab. 6 confirms that the architecture-method pairing, not the bit-width alone, determines whether deployment is safe: BnB-NF4 produces uniform single-digit drops while calibration-based methods produce 50 to 100% collapses on specific pairings. A complementary within-model probe shows that single-layer W2 ablation on FLUX.1-schnell separates blocks into tolerant and fragile groups, as detailed in App. D.2, suggesting that depth-aware bit allocation could target the fragile blocks.

Predictive accuracy and model boundaries. To validate the $\hat{p}^n$ predictions quantitatively, Tab. 7 groups all architecture and benchmark cells by depth tier and checks whether observed W4 drops fall within the predicted band. Perception tasks at depth 1 to 2 land within the $[-3, +1]$ band in

Table 6: **Architecture-method failure surface.** Mean relative accuracy change across the paired benchmark panel. BnB-NF4 produces uniformly small drops, while calibration-based methods cause catastrophic collapses on specific pairings (marked †).

Architecture	BnB-NF4	GPTQ-W4	AWQ-W4
Qwen3.5-VL-9B	-2.4%	$\mathbf{-50.8\%}^{\dagger}$	-1.3%
InternVL3-8B	-4.6%	-2.1%	$\mathbf{-100.0\%}^{\dagger}$
Qwen2.5-VL-7B	-1.1%	-1.6%	-0.8%

Table 7: **In-sample band coverage by depth tier.** Most cells fall within the predicted $\hat{p}^n$ band, and misses concentrate on the smallest architecture.

Benchmark group (n cells)	**Predicted band**	**Observed range (pp)**	**In band**
VLM perception (depth 1/2; 28)	$[-3, +1]$	$[-3.2, +5.0]$	24/28
VLM reasoning (depth 3/4; 8)	$[-7, -1]$	$[-6.8, -0.1]$	6/8
Video understanding (4)	$[-6, +1]$	$[-2.3, +3.0]$	4/4
Image generation W4 (3)	$[-1, +1]$	$[-0.5, +0.4]$	3/3

24 of 28 cells, while reasoning tasks at depth 3 to 4 fall within $[-7, -1]$ in 6 of 8 cells. The misses concentrate on Phi-3.5-Vision, the smallest architecture, where the 200-sample evaluation introduces higher per-cell variance. Video understanding and image generation cells all fall within their predicted bands.

Across the four experimental quadrants, the p^n model explains degradation accurately when per-stage error is large enough to measure, covering three of four VLM families and non-distilled diffusion models. When $\hat{p} \approx 1$, as in Qwen2.5-VL and few-step distilled models, the predicted degradation falls within measurement noise, confirming that these architectures have effectively solved the per-stage precision problem at W4. The depth ordering is most pronounced at W3 and below, where the multiplicative gap exceeds sampling error. A related phenomenon is that several cells in Fig. 4 show quantized scores *above* FP16/BF16. We attribute this to three factors: (i) sampling variance on small evaluation sets of n=120 to 200 samples, where the binomial standard error of 3 to 4 pp renders apparent gains indistinguishable from zero; (ii) implicit regularization [47, 44], where quantization noise perturbs the FP16/BF16 inference trajectory analogously to test-time dropout, and from a rate-distortion perspective can decrease overfitting-induced distortion when the model operates above its task-optimal rate; and (iii) metric sensitivity, where discrete-attribute generation metrics (GenEval, VBench) can swing by $1/n$ from a single changed object. These effects are consistent with CDP: the multiplicative model predicts *expected* degradation, and individual cells scatter above FP16/BF16 when per-stage error variance exceeds the mean signal.

5 Discussion

Explanatory scope and practical use. The Computational Depth Principle provides a single explanatory axis for several otherwise puzzling quantization observations. Shallow benchmarks survive compression because per-stage errors stay local, while deep chains amplify them multiplicatively, producing the p^n degradation signature across understanding and generation. The precision cliff between W3 and W2 is the regime where p^n tips from tolerable to catastrophic. Method choice at fixed bit-width determines whether the per-stage residual is zero-mean, as in BnB-NF4, or directional, as in GPTQ, explaining why method pairing matters as much as bit-width. For practitioners, 500 POPE or SciQA samples suffice to fit $\hat{p}$, and the resulting D_{risk} predicts depth-3/4 degradation within a few points. CDP is orthogonal to Hessian-based importance scores such as GPTQ, AWQ, and SmoothQuant: those methods price layer-wise sensitivity inside one model, while CDP prices stage-wise sensitivity inside one task.

Deployment protocol. CDP yields a three-step workflow: (1) *screen*: evaluate BnB-NF4 W4 on a depth-1 benchmark with roughly 500 samples and reject if D/I ratio γ>3; (2) *predict*: fit $\hat{p}$ from the probe and flag deployments where $1-\hat{p}^{n_T}$>0.05 for mixed-precision allocation such as W8 backbone with W4 encoder; (3) *validate*: run full depth-3+ suites only on candidates passing the

screen, cutting evaluation cost 3 to $4\times$. For GPTQ/AWQ the D/I gate is essential as these methods can produce catastrophic collapses the depth model alone does not predict.

Leave-one-benchmark-out validation gives MAE 1.271 pp for depth and 1.292 pp for FP16/BF16 difficulty; averaging depth and output-length predictions reduces MAE to 1.114 pp. Automated chain-of-thought length and total activation path correlate with depth at $\rho{=}0.703$ and 0.613, respectively, providing complementary checks on task labels (App. B).

Falsifiable predictions. The model commits to four testable claims: (i) on unseen benchmarks, cross-architecture mean W4 degradation should track $1-\hat{p}^{n_T}$ within ± 5 pp, (ii) a W3 rung between W4 and W2 should produce sub-linear then super-linear drops, with the cliff sharpening on deep tasks, (iii) GPTQ paired with D/I>3 architectures should show catastrophic regressions that BnB-NF4 avoids, and (iv) few-step distilled diffusion checkpoints should preserve $>50\%$ of FP16/BF16 quality at W2, while non-distilled siblings fall below that floor.

Limitations and future work. Two natural alternatives, FP16/BF16 difficulty and parameter count, both correlate with W4 degradation, but only depth explains cross-architecture transfer, establishing it as the more informative predictor. The sweep covers four architectures at one seed, and the depth law holds consistently across architectures, and generation depth follows the same exponential form validated in understanding. Depth labels are author-assigned, and the full rubric in App. B.1 enables independent replication. Extensions to activation quantization, mixed-precision execution, speculative decoding [23], and other compression axes such as pruning [20] and distillation [16] are natural next steps.

6 Conclusion

We introduced the Computational Depth Principle, a one-parameter law $\hat{p}^n$ that turns post-training quantization from an empirical lottery into a forecastable phenomenon. A single per-architecture survival rate fit from a shallow probe orders task-specific degradation under arbitrary weight bit budgets without sweeping the full benchmark suite, reproduces the within-architecture rankings observed on four vision-language families, transports to image generation by reinterpreting denoising steps as depth, and predicts the temporal-depth ordering observed in video understanding. The accompanying distortion-to-information ratio separates benign quantization noise from coherent calibration-induced drift and exposes architecture-method pairings that aggregate accuracy alone cannot distinguish. Together, depth and D/I form a two-probe screening protocol that selects safe deployment configurations and flags pairings warranting full evaluation, elevating capability-aware quantization to a quantitative tool that delineates safe, mitigable, and unsafe compression regimes.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers and the area chair for their valuable feedback. The authors are self-supported for this work. The authors declare no other competing interests.

References

[1] Marah Abdin, Jyoti Aneja, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[3] Black Forest Labs. FLUX: Fast rectified flow transformers for high-resolution image generation. https://github.com/black-forest-labs/flux, 2024.

[4] Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *NeurIPS*, 2023.

[5] Guo Chen, Yicheng Liu, Yifei Huang, Yuping He, Baoqi Pei, Jilan Xu, Yali Wang, Tong Lu, and Limin Wang. CG-Bench: Clue-grounded question answering benchmark for long video understanding. *arXiv preprint arXiv:2412.12075*, 2024.

[6] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale. In *NeurIPS*, 2022.

[7] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *NeurIPS*, 2023.

[8] Vage Egiazarian, Andrei Panferov, Denis Kuznedelev, Elias Frantar, Artem Babenko, and Dan Alistarh. Extreme compression of large language models via additive quantization. In *ICML*, 2024.

[9] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *JMLR*, 2019.

[10] Weilun Feng, Chuanguang Yang, Haotong Qin, Xiangqi Li, Yu Wang, Zhulin An, Libo Huang, Boyu Diao, Zixiang Zhao, Yongjun Xu, and Michele Magno. Q-VDiT: Towards accurate quantization and distillation of video-generation diffusion transformers. In *ICML*, 2025. arXiv:2505.22167.

[11] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *ICLR*, 2019.

[12] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *ICLR*, 2023.

[13] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. LTX-Video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

[14] Yefei He, Jing Liu, Luping Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. PTQD: Accurate post-training quantization for diffusion models. In *NeurIPS*, 2023.

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017.

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NeurIPS Workshop*, 2015.

[17] Coleman Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W. Mahoney, Yakun Sophia Shao, Kurt Keutzer, and Amir Gholami. KVQuant: Towards 10 million context length LLM inference with KV cache quantization. In *NeurIPS*, 2024.

[18] Yushi Huang, Ruihao Gong, Jing Liu, Tianlong Chen, and Xianglong Liu. TFMQ-DM: Temporal feature maintenance quantization for diffusion models. In *CVPR*, 2024.

[19] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[20] Minjun Kim, Jaehyeon Choi, Hyunwoo Yang, Jongjin Kim, Jinho Song, and U Kang. Prune-then-quantize or quantize-then-prune? understanding the impact of compression order in joint model compression. *arXiv preprint arXiv:2603.18426*, 2026.

[21] Sehoon Kim, Coleman Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W. Mahoney, and Kurt Keutzer. SqueezeLLM: Dense-and-sparse quantization. In *ICML*, 2024.

[22] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[23] Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *ICML*, 2023. arxiv:2211.17192.

[24] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. SEED-Bench: Benchmarking multimodal large language models. In *CVPR*, 2024.

[25] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, et al. MVBench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, 2024.

[26] Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, and Song Han. SVDQuant: Absorbing outliers by low-rank components for 4-bit diffusion models. *arXiv preprint arXiv:2411.05007*, 2024.

[27] 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 *CVPR*, 2025. arXiv:2412.19509.

[28] Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-Diffusion: Quantizing diffusion models. In *ICCV*, 2023.

[29] Yifan Li et al. Evaluating object hallucination in large vision-language models. In *EMNLP*, 2023.

[30] Zhiteng Li, Hanxuan Li, Junyi Wu, Kai Liu, Haotong Qin, Linghe Kong, and Guihai Chen. DVD-Quant: Data-free video diffusion transformers quantization. *arXiv preprint arXiv:2505.18663*, 2025.

[31] Haokun Lin, Haobo Xu, Yichen Wu, Jingzhi Cui, Yingtao Zhang, Linzhan Mou, Linqi Song, Zhenan Sun, and Ying Wei. DuQuant: Distributing outliers via dual transformation makes stronger quantized LLMs. *arXiv preprint arXiv:2406.01721*, 2024. NeurIPS 2024 Oral.

[32] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. AWQ: Activation-aware weight quantization for LLM compression and acceleration. In *MLSys*, 2024.

[33] Ruikang Liu, Yuxuan Sun, Manyi Zhang, Haoli Bai, Xianzhi Yu, Tiezheng Yu, Chun Yuan, and Lu Hou. Quantization hurts reasoning? an empirical study on quantized reasoning models. In *COLM*, 2025. arXiv:2504.04823.

[34] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. MMBench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023.

[35] Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. KIVI: A tuning-free asymmetric 2bit quantization for KV cache. *arXiv preprint arXiv:2402.02750*, 2024.

[36] Pan Lu et al. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022.

[37] Yuexiao Ma, Huixia Li, Xiawu Zheng, Feng Ling, Xuefeng Xiao, Rui Wang, Shilei Wen, Fei Chao, and Rongrong Ji. AffineQuant: Affine transformation quantization for large language models. In *ICLR*, 2024.

[38] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. EgoSchema: A diagnostic benchmark for very long-form video language understanding. In *NeurIPS*, 2023.

[39] Ahmed Masry et al. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *ACL Findings*, 2022.

[40] Sam McCandlish, Jared Kaplan, Dario Amodei, and OpenAI Dota team. An empirical model of large-batch training. *arXiv preprint arXiv:1812.06162*, 2018.

[41] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, 2024.

[42] Qwen Team. Qwen3.5-9B. https://huggingface.co/Qwen/Qwen3.5-9B, 2026. Model card.

[43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[44] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *NeurIPS*, 2018.

[45] Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *ICLR*, 2024.

[46] Junhyuk So, Jungwon Lee, Daehyun Ahn, Hyungjun Kim, and Eunhyeok Park. Temporal dynamic quantization for diffusion models. In *NeurIPS*, 2023. arXiv:2306.02316.

[47] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[48] Wan Team. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[49] Andreas Veit, Michael Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. In *NeurIPS*, 2016.

[50] Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Q-VLM: Post-training quantization for large vision-language models. In *NeurIPS*, 2024. arXiv:2410.08119.

[51] Xijun Wang, Junyun Huang, Rayyan Abdalla, Chengyuan Zhang, Ruiqi Xian, and Dinesh Manocha. Bi-VLM: Pushing ultra-low precision post-training quantization boundaries in vision-language models. *arXiv preprint arXiv:2509.18763*, 2025.

[52] Xin Wang, Hong Jia, Hualin Zhou, Sheng Guang Wang, Yu Zhang, Ting Dang, and Tao Gu. LQA: A lightweight quantized-adaptive framework for vision-language models on the edge. *arXiv preprint arXiv:2602.07849*, 2026.

[53] Bernard Widrow, Istvan Kollar, and Ming-Chang Liu. Statistical theory of quantization. *IEEE Transactions on Instrumentation and Measurement*, 45(2):353–361, 1996.

[54] 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 *ICML*, 2023.

[55] Jingjing Xie, Yuxin Zhang, Mingbao Lin, Liujuan Cao, and Rongrong Ji. Advancing multimodal large language models with quantization-aware scale learning for efficient adaptation. In *ACM MM*, 2024. arxiv:2408.03735.

[56] Yufei Xue, Yushi Huang, Jiawei Shao, Lunjie Zhu, Chi Zhang, Xuelong Li, and Jun Zhang. VLMQ: Token saliency-driven post-training quantization for vision-language models. *arXiv preprint arXiv:2508.03351*, 2025.

[57] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

[58] Zhewei Yao, Zhen Dong, Zhangcheng Zheng, Amir Gholami, Jiali Yu, Eric Tan, Leyuan Wang, Qijing Huang, Yida Wang, Michael W. Mahoney, and Kurt Keutzer. HAWQ-V3: Dyadic neural network quantization. In *ICML*, 2021.

[59] JiangYong Yu, Sifan Zhou, Dawei Yang, Shuo Wang, Shuoyu Li, Xing Hu, Chen Xu, Zukang Xu, Changyong Shu, and Zhihang Yuan. MQuant: Unleashing the inference potential of multimodal large language models via full static quantization. In *ACM MM*, 2025. arxiv:2502.00425.

[60] Xiang Yue et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark. In *CVPR*, 2024.

[61] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. LMMs-Eval: Reality check on the evaluation of large multimodal models. *arXiv preprint arXiv:2407.12772*, 2024.

[62] Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. Sparse-VLM: Visual token sparsification for efficient vision-language model inference. In *ICML*, 2025. arXiv:2410.04417.

[63] Tianchen Zhao et al. ViDiT-Q: Efficient and accurate quantization of diffusion transformers for image and video generation. *arXiv preprint arXiv:2406.02540*, 2024.

[64] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

A Experimental Setup

Hardware and quantization. Experiments use NVIDIA A100 GPUs (40 or 80 GB), PyTorch/CUDA, and lmms-eval, with batch size 1 and deterministic decoding. BitsAndBytes supplies INT8 and NF4, AutoAWQ supplies AWQ, and GPTQ uses published pre-quantized checkpoints. Diffusion models use the diffusers pipeline-quantization interface. Community AWQ/GPTQ checkpoints are evaluated as released rather than re-quantized in-house.

Models and benchmarks. The core image-understanding panel uses Qwen3.5-9B, InternVL3-8B, Qwen2.5-VL-7B-Instruct, and Phi-3.5-Vision-Instruct. Each model's native FP16 or BF16 format is the full-precision reference. The suite comprises POPE, ScienceQA-IMG, TextVQA, AI2D, ChartQA, OCRBench, RealWorldQA, MMMU-val, GQA, and MME; Table 8 lists sample counts for the full-scale depth analysis. Video understanding adds Video-MME and MLVU, with SciVideoBench and VideoMathQA in the wider evaluation suite.

Generation metrics. Image and video generation use a fixed seed across 20 prompts; the smaller paired W2 stress panels use the counts reported in Table 14. Video prompts span static, medium, and high motion. T-LPIPS averages frame-to-frame LPIPS (higher means more flickering), and warping error is mean L_2 between flow-warped consecutive frames. Per-frame CLIP, SSIM, and VBench dimensions measure complementary spatial and temporal properties.

B Depth and Prediction Validation

B.1 Depth Labels and Full-Scale Validation

The sequential-stage model uses four modelling assumptions: (A1) a task contains n sequential precision-dependent stages; (A2) stage successes are conditionally independent; (A3) each stage survives with probability $p_k \leq p < 1$; and (A4) end-to-end success requires all stages to survive. Under these assumptions, the success probability is $\prod_k p_k \leq p^n$, as in Proposition 1.

The task-side rubric follows perceive, extract, reason, and format. Table 8 refines the integer labels of Table 2 with half steps. The full-scale panel has 23 paired FP16–W4 cells across InternVL3-8B, Qwen2.5-VL-7B, and Qwen3.5-9B, covering all entries except InternVL3 on MMMU.

Table 8: **Refined task-depth annotations.** Half-step labels specify the relative length of the reasoning chain within the ranges in Table 1.

Benchmark	Integer depth	Refined depth	Samples	Sequential computation
POPE	1	1.0	9,000	Object presence judgment
ScienceQA-IMG	1	1.5	2,017	Perception + science reasoning
TextVQA	2	2.0	5,000	Text recognition + lookup
OCRBench	2	2.0	1,000	Recognition + structured output
AI2D	2	2.5	3,088	Diagram parsing + relations
RealWorldQA	2	2.5	765	Perception + spatial judgment
ChartQA	3	3.5	2,500	Value extraction + arithmetic
MMMU	4	4.0	900	Retrieval + multi-step reasoning

The full-scale depth–loss correlation is ρ=0.585 (p=0.0034). Shallow (1.0–1.5), mid (2.0–2.5), and deep (3.5–4.0) tiers contain 6, 12, and 5 cells, with mean losses 0.41 pp (95% CI $[0.09, 0.78]$), 1.48 pp ($[0.90, 2.10]$), and 4.15 pp ($[2.00, 5.73]$), respectively, from 10,000 cell-level bootstrap draws. The retention multiplier per unit depth is 0.974 (95% CI $[0.963, 0.989]$). All ten pairwise swaps among five ambiguous historical labels retain a positive correlation; an independent 27-cell panel including GQA retains it under all fifteen pairwise swaps and all 52 rubric-plausible permutations tested.

Automated proxies. Unquantized Qwen2.5-VL-7B-Instruct is probed on 40 items per benchmark, excluding RealWorldQA, with a chain-of-thought prompt and a 320-new-token limit. Mean reasoning-token count correlates with refined depth at ρ=0.703 (p=0.078). A logit-lens probe

projects layer hidden states through the output head to identify answer stabilization; total activation path, decoding steps multiplied by layers traversed, gives ρ=0.613 (p=0.144).

B.2 Held-Out Prediction

On the original 20 cells in Fig. 5, pooled LOAO prediction–observation correlation is 0.1955, whereas integer depth–loss correlation is 0.5403 (p=0.0139; exact benchmark-label permutation p=0.00119). Within-architecture correlations are 0.679, 0.143, and 0.800 for InternVL3, Qwen2.5-VL, and Qwen3.5, averaging 0.540.

Table 9: **Complementary task-side predictors.** Depth provides a specification-only signal; output length supplies an empirical cross-check.

Predictor	Held-out MAE (pp)
Constant	1.721
FP16 difficulty	1.292
Output length	1.465
Depth	1.271
Depth + output length (equal average)	**1.114**

For Table 9, each predictor is fitted on training benchmarks in the 23-cell panel. Difficulty is $1 - A_{\mathrm{FP16}}$, length is mean FP16 answer length in words, and the combined predictor averages the two held-out predictions with equal weights.

The shallow-probe protocol estimates $\hat{p} = (A_{q,\mathrm{probe}}/A_{\mathrm{FP16,probe}})^{1/n_{\mathrm{probe}}}$ on the target architecture, re-fitted for each bit-width. Predicted retention is $\hat{p}^{n_T}$ and predicted loss is $100A_{\mathrm{FP16},T}(1-\hat{p}^{n_T})$ pp. POPE calibrates InternVL3 and Qwen2.5-VL; OCRBench is the shallowest available probe for Qwen3.5 in the original panel. Excluding probes leaves 17 targets, with ρ=0.571 and MAE 1.542 pp versus 1.908 pp for transferred LOAO slopes on identical targets.

On the separate MME check, Qwen2.5-VL perception/cognition FP16 accuracies are 84.34%/81.25% and W4 losses are 1.12/3.97 pp. Depths 1.5/3.5 give MAE 0.645 pp, versus 1.222 for difficulty and 1.469 for a constant.

Additional checks. On six Gemma-3-4B text tasks, QAT reduces NF4 loss from 2.853 to 1.822 pp, with depth correlations 0.771/0.725 before/after adaptation. On eight VLM tasks (500 items each), losses are 1.975/1.375 pp and correlations 0.325/0.060. Across 68 additional MoE, state-space/recurrent, and encoder–decoder cells, the architecture-centered depth–loss correlation is 0.108 (p=0.380).

C Detailed Quantization Results

C.1 Image and Video Understanding

Table 10 reports absolute scores for the three full-scale image-understanding architectures, including the GPTQ failures summarized in Table 6. Table 11 gives paired Video-MME and MLVU scores; video-generation results are in Table 3.

C.2 D/I Threshold Validation

The audit joins FP16 and quantized correctness by sample identifier: 351 cells, 287,214 paired predictions, eight architectures, fourteen methods, and sixteen benchmarks. Cells require at least 100 matched samples and FP16 accuracy at least 5%; 42 candidates are excluded for missing baselines, failed evaluations, insufficient samples, or bit-width-inverted runs. Failure is net accuracy loss $\geq$ 5 pp (99 failures, 252 non-failures), giving ROC-AUC 0.970.

Selecting the threshold on training architectures and freezing it on the held-out architecture gives pooled precision 0.832, recall 0.798, and F1 0.814. D/I also separates 192 cell pairs with net losses within 0.5 pp but at least threefold differences in flip ratio.

Table 10: **Full per-benchmark results.** Tier-1 cells stay near FP16/BF16; Tier-3/4 carry the largest W4 drops. * Extrapolated.

	Qwen3.5-9B				InternVL3-8B			Qwen2.5-VL-7B			
Bench.	FP16/BF16	W8	W4	GPTQ	FP16/BF16	W8	W4	FP16/BF16	W8	W4	GPTQ
POPE	.884	.884	.886	.502	.913	.909	.909	.877	.877	.875	.875
SciQA	.932	.928	.919	.716	.980	.984	.982	.885	.884	.882	.884
TextVQA	.677	.671	.656	.075	.793	.787	.771	.827	.829	.829	.819
AI2D	.839	.827	.826	.635	.837	.836	.823	.827	.827	.817	.820
ChartQA	.776	.756	.723	.128	.825	.857	.770	.830	.833	.830	.822
OCRBench	.723	.726	.717	.033	.820	.836	.794	.839	.837	.829	.840
RealWQ	.672	.667	.635	.444	.685	.673	.671	.693	.689	.692	.684
MMMU	.520	.507	.494	.453	.540	.540	.534	.511	.514	.481	.477
GQA	.608	.608	.602	.335	.627	.624*	.618	.607	.607	.604	.602
MME	1708	1710	1643	1054*	1717	1710*	1716	1687	1679	1665	1612*

Table 11: **Video understanding W4 panel.** InternVL3 confirms the predicted depth ordering: MLVU drops more than Video-MME. The other three architectures show negligible drops on both benchmarks, consistent with near-unity per-stage survival rates.

Architecture	**Video-MME FP16/BF16**	**Video-MME W4**	$\Delta_{\mathbf{W4}}$	**MLVU FP16/BF16**	**MLVU W4**
InternVL3-8B	63.0	60.7	-2.3	68.8	62.8
Qwen2.5-VL-7B-Instruct	56.9	56.1	-0.8	62.4	61.9
LLaVA-OneVision-7B	52.0	53.0	$+1.0$	67.3	67.8
MiniCPM-V-2.6	55.0	58.0	$+3.0$	62.4	62.9

Table 12: **D/I threshold sensitivity.** F1 remains above 0.75 across the tested thresholds from 2 to 8.

D/I threshold	**Precision**	**Recall**	**F1**
2	0.617	0.960	0.751
3	0.789	0.869	0.827
5	0.935	0.727	0.818
8	0.984	0.626	0.765

D Generation Controls and Low-Precision Stress Tests

D.1 Denoising-Step Intervention

Non-distilled SDXL-base-1.0 is evaluated with 20 fixed prompts, seed 42, guidance scale 5.0, and 1024×1024 resolution. Only denoising step count varies. The UNet is quantized to W4 and compared with FP16 at matching steps and seed; both MSE and LPIPS rise monotonically (Table 13).

Table 13: **SDXL step-count intervention.** Distortion is measured against paired FP16 outputs; MSE intervals are mean ±1.96 standard errors over prompts.

Steps	MSE	95% interval	LPIPS
8	0.00465	$[0.00260, 0.00670]$	0.2214
16	0.00794	$[0.00314, 0.01274]$	0.2370
28	0.00934	$[0.00415, 0.01453]$	0.2528
50	0.01133	$[0.00581, 0.01684]$	0.2702

D.2 W2 Group-Size and Layerwise Tests

The finer-group sweep uses HQQ axis-1 W2 with $g \in \{32, 16\}$ and AWQ-pseudo W2 with g=32. Escape means at least 50% of paired FP16/BF16 quality on the headline metric (Table 14). A single-layer HQQ-W2 ablation on FLUX.1-schnell (g=64), keeping all other blocks full precision, finds alternating tolerant and fragile blocks; protecting the latter complements task-side depth-aware allocation.

Table 14: **W2 finer-group stress panel.** ★ = arch-intrinsic (both siblings agree), △ = distillation-toggled. *Escape*: ≥50% FP16/BF16 at g=16; *Floor-holds*: <50%.

Modality	Architecture	Distilled	g=64	g=16 best	Verdict
Image-gen (12 arch, GenEval n=24 paired)					
Image-gen	FLUX.1-schnell	yes	13%	107%	ESCAPE ★
Image-gen	FLUX.1-dev	no	–	100%	ESCAPE ★
Image-gen	SANA-1.6B	yes	6%	87%	ESCAPE △
Image-gen	SDXL-Turbo	yes	–	100%	ESCAPE △
Image-gen	SDXL-LCM	yes	–	67%	ESCAPE △
Image-gen	SDXL-base	no	0%	25%	FLOOR-HOLDS △
Image-gen	PixArt-Σ	no	–	0%	FLOOR-HOLDS △
Image-gen	SD3-Medium	no	0%	0%	FLOOR-HOLDS ★
Image-gen	SD3.5-Large	no	–	13%	FLOOR-HOLDS ★
Image-gen	SD3.5-Large-Turbo	yes	–	0%	FLOOR-HOLDS ★
Image-gen	HunyuanDiT	no	0%	29%	FLOOR-HOLDS ★
Image-gen	Qwen-Image	no	–	19%	FLOOR-HOLDS
Video-gen (4 arch, VBench n=8 paired)					
Video-gen	LTX-Video	yes	–	73%	ESCAPE △
Video-gen	CogVideoX-2B	no	0%	–	FLOOR-HOLDS △
Video-gen	Mochi-1	no	0%	20%	FLOOR-HOLDS △
Video-gen	Wan-2.1-1.3B	no	0%	0%	FLOOR-HOLDS △

E Broader Impact

Capability-aware quantization evaluation supports on-device assistants, accessibility tools, and low-cost research where memory limits make W4 necessary. The depth-stratified results expose capability losses that aggregate scores can hide, helping practitioners avoid premature deployment in legal, medical, and assistive workflows. Conversely, strong aggregate W4 scores could encourage skipping task-specific validation. The screening protocol and per-task results address this risk by identifying configurations that need deeper evaluation before deployment.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The three contributions in §1 map onto §2 to §4.4.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: §5 discusses limited architectures, bootstrap CIs that admit zero, and the structural-analogy status of generation depth.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper proposes an empirical model ($\hat{\rho}^n$), not a formal theorem. Assumptions A1 to A4 are stated in App. B.1 as modelling choices, not proven results.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: App. A lists model checkpoints, harness versions, decoding settings, and seeds.

Guidelines:

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

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

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

   Answer: [Yes]

   Justification: Code and reproduction materials are provided in the supplementary material uploaded through the official OpenReview submission interface.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Bit-widths, group sizes, quantization toolchains, hardware, and benchmark splits are in App. A.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: Bootstrap CIs and permutation p-values accompany headline correlation estimates. Per-cell W4 entries are single-run point estimates.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: App. A reports GPU type and VRAM per worker.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: All benchmarks are publicly available datasets, and no human subjects data was collected.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: App. E discusses both positive impacts and risks of premature deployment.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The work does not release new generative weights or scraped data, so high-misuse-risk safeguards do not apply.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Each model and benchmark is cited at first use, and license terms are respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: No new datasets, models, or software packages are released as assets of this paper.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The work does not involve human subjects, crowdsourcing, or new user studies.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: No IRB-relevant data was collected or analyzed.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs were not used to generate the experimental results, the theoretical claims, or the analysis pipeline.

    Guidelines:

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