Context-Aware Cluster Decoding: Semantic Anchor-Driven Coherence in dMLLMs

Yikai Zhao[2]*, Qiyan Zhao[3], Jiaquan Zhang[4], Xiaofeng Zhang[3]
Xiaosong Yuan[5], Pengzhou Cheng[1]†
[1]Shanghai University [2]Sun Yat-sen University [3]Shanghai Jiao Tong University
[4]University of Electronic Science and Technology of China [5]Alibaba Group
zhaoyk25@mail2.sysu.edu.cn, chengpz@shu.edu.cn

Abstract

Diffusion multimodal large language models (dMLLMs) frequently produce long-form outputs marred by semantic drift and repetition, with quality generally degrading as output length increases. We identify two structural deficiencies in existing decoding methods as primary drivers of these failures: confidence-based scoring ignores decoded-neighbor support, and block partitioning prevents access to high-readiness semantic anchors, together causing tokens to be committed before their local context is sufficiently established. We propose CACD (**C**ontext-**A**ware **C**luster **D**ecoding), a training-free decoding method that scores each masked position by a multiplicative composite of softmax confidence and neighbor proximity, promoting contextually ready tokens above isolated candidates while suppressing low-confidence positional noise, operating block-free to keep high-readiness anchors globally accessible. CACD further applies architecture-aware calibration to handle confidence heterogeneity induced by diverse visual integration strategies. Experiments on three dMLLMs across four benchmarks demonstrate consistent quality gains and hallucination reduction over Original, with larger gains in several longer generation settings, highlighting the importance of neighbor support and visual integration strategy for future dMLLM decoding method design. Our code is openly available at https://github.com/zhaoyk-sysu/CACD-dMLLM.

1 Introduction

Diffusion multimodal large language models (dMLLMs (Nie et al., 2025; Bie et al., 2025; Yang et al., 2025; Li et al., 2025b; You et al., 2025; Yu et al., 2025b; Swerdlow et al., 2025)) have emerged as a compelling alternative to autoregressive generation, offering parallel decoding under bidirectional attention as a principled departure from left-to-right token commitment. Despite this architectural advantage, dMLLMs frequently produce long-form outputs marred by semantic drift and repetitive phrasing (Figure 1), with quality gaps widening monotonically with output length (Figure 2).

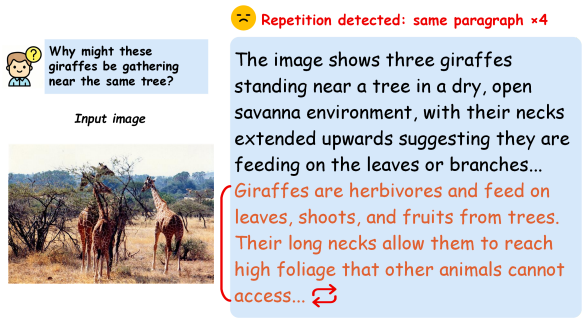


Figure 1: A representative block-diffusion failure: the same paragraph repeats four times, a symptom of incorrect commitment order leaving neighboring positions without local semantic support.

Our analysis suggests that one important source of these degradations lies in how existing decoders handle commitment order: they neither identify high-quality semantic anchors nor exploit local semantic connections between decoded tokens and their masked neighbors, which causes positions to be committed in an order that leaves subsequent tokens without sufficient local support. The key insight is that local connections to decoded neighbors substantially affect contextual readiness beyond raw confidence: on 1000 MathVista samples, positions with both neighbors decoded achieve $5.3\times$ lower posterior entropy than isolated positions, suggesting that decoded neighbors provide strong local semantic constraint (Kim et al., 2025) (Figure 3a). When a contextually ready token is wrongly deferred, a weaker candidate is committed in its place, entering surrounding context without sufficient lo-

*This work is supported by Open Research Fund of The State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS2026079).
† Corresponding author.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 47182–47200
October 24-29, 2026 ©2026 Association for Computational Linguistics

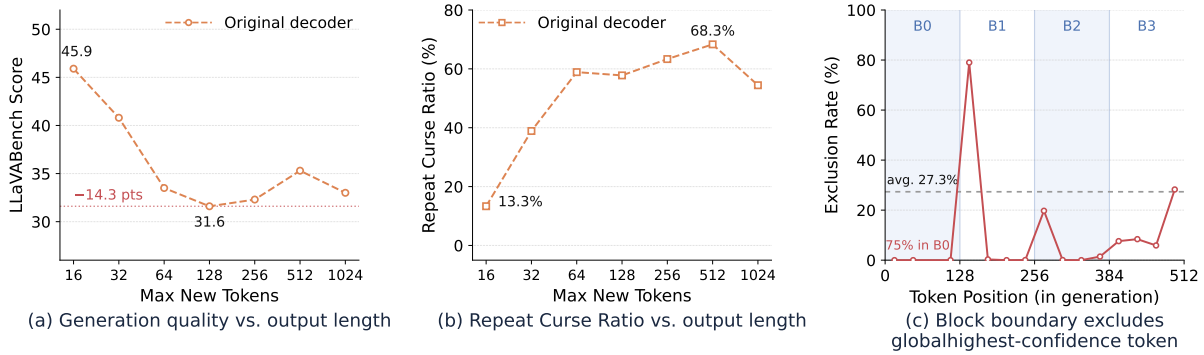


Figure 2: Two failure modes of the Original block-diffusion method on MMaDA-8B-Base. **(a)** LLaVABench-COCO score degrades monotonically with output length. **(b)** Sample Repeat Curse Ratio on LLaVABench-COCO increases with output length, measured following (Zhao et al., 2026a). **(c)** A high-readiness semantic anchor candidate lies outside the current block in **75.0%** of first-round steps and **27.3%** of all steps (block_length= 128, 90 LLaVABench-COCO samples), revealing that block partitioning systematically prevents access to highly effective anchors.

cal support and degrading subsequent predictions. This effect compounds with sequence length, ultimately producing semantic drift and hallucination.

Two structural obstacles prevent existing decoding methods from exploiting these properties.

Finding 1: Confidence ignores neighbor support. Existing scoring methods (Wu et al., 2026b; Israel et al., 2025; Chen et al., 2025) rank positions by raw confidence, entropy, or autoregressive mixtures, none of which measures the local semantic support a position's neighborhood already provides. Tokens that are contextually ready (strongly anchored by committed neighbors) are systematically ranked below isolated tokens of comparable confidence, causing the latter to be committed first despite weaker local support; conversely, low-confidence tokens at favorable positions incur no penalty under pure confidence scoring.

Finding 2: Block structure can exclude high-readiness anchor candidates. Even with accurate scoring, block-based decoding methods (Arriola et al., 2025; Lu et al., 2026; Shu et al., 2026) cannot always access the highest-readiness anchor candidate. This exclusion is most damaging during the first decoding round, when committed tokens establish the primary semantic anchors for all subsequent positions: **75.0%** of first-round steps find the globally highest-readiness anchor outside the current block (Figure 2b), and the exclusion persists in **27.3%** of all steps, meaning over one in four opportunities to commit the strongest available anchor is structurally blocked. AdaBlock (Lu et al., 2026) and DCD (Shu et al., 2026) progressively reduce this misalignment but cannot fully eliminate it without abandoning sequential block structure.

Both findings deny subsequent positions the local semantic support needed for reliable prediction. We propose CACD (**C**ontext-**A**ware **C**luster **D**ecoding), which scores every masked position by a multiplicative composite of softmax confidence and neighbor proximity, operates block-free to keep the most informative anchor always accessible, and applies architecture-aware calibration to handle confidence heterogeneity across dMLLM architectures. Extensive experiments across three dMLLMs and four benchmarks confirm that CACD provides an effective and broadly applicable commitment signal.

Our contributions are:

- We identify that existing decoding methods systematically fail to satisfy two critical properties: semantic anchor accessibility and contextual readiness ordering. We show this failure significantly limits long-form generation quality in dMLLMs, providing empirical evidence that block partitioning cuts off high-readiness anchors in **75.0%** of first-round steps and **27.3%** of all steps.
- We introduce CACD, a plug-and-play decoding framework for dMLLMs. We demonstrate that a multiplicative composite score of softmax confidence and local semantic context simultaneously rescues contextually ready tokens and suppresses positional noise. This joint formulation provides a more effective commitment signal than confidence alone while maintaining global anchor accessibility.
- We show that distinct visual integration strate-

gies shape the confidence landscape in ways existing text-only decoding methods do not account for, a dimension that serves as an essential consideration for future dMLLM decoding method design.

2 Related Work

The masked diffusion decoding process (Nie et al., 2025) starts from a fully masked sequence of tokens and iteratively unmasks positions, where at each step the model predicts all masked tokens in parallel and a sampler selects a subset to commit based on confidence or other criteria.

Commitment Order and Prediction Difficulty. Kim et al. (Kim et al., 2025) establish that commitment order is a primary determinant of generation quality, and that committing high-certainty positions first substantially outperforms fixed-order decoding. Zhou et al. proposed HDLM (Zhou et al., 2025) (NeurIPS 2025), which reinforces this from the training perspective via a coarse-to-fine semantic hierarchy. Neither work provides an inference-time mechanism that maintains correct commitment order under complex visual context or long-form outputs, leaving the resulting quality cost unaddressed; Section 3.1 quantifies it directly.

Decoding Order and Block Structure. Arriola et al. proposed Block Diffusion (Arriola et al., 2025), partitioning sequences into fixed blocks. AdaBlock (Lu et al., 2026) (ICLR 2026), DCD (Shu et al., 2026) and others (Zhang et al., 2024a, 2025a, 2024b, 2025b, 2026a; Zhao et al., 2026b; Zhang et al., 2026b; Chang et al., 2025; Zuo et al., 2026) refine block boundaries in a more flexible manner but retain strict sequential structure, structurally preventing access to highest-readiness anchor candidates. WavefrontDiffusion (Yang et al., 2026) (ICLR 2026) expands a frontier from finalized positions and explicitly models spatial neighbor distance, yet its commitment signal remains confidence-based without incorporating bilateral neighbor support into scoring. AHD (Zou et al., 2026) enables early cross-block decoding via historical convergence trends, requiring a multi-step history buffer; CACD achieves neighbor-aware commitment within a single forward pass. Kang et al. (Kang et al., 2026) confirm that token dependency strength is the primary determinant of decoding quality, directly supporting neighborhood support as the key commitment signal.

Confidence-Based Scoring. Fast-dLLM (Wu et al., 2026b), APD (Israel et al., 2025), and Slow-Fast Sampling (Wei et al., 2026) (ICLR 2026) improve commitment through confidence-threshold and dynamic scoring strategies (Cai and Li, 2026), but treat confidence as the sole signal without measuring neighborhood support. Chen et al. proposed Coherent Contextual Decoding (Chen et al., 2025), which identifies semantically pivotal tokens via cross-step consistency in a sliding history buffer and prioritizes their commitment. By contrast, our context score operates purely in the spatial domain, measuring proximity to currently decoded bilateral neighbors within a single forward pass; the two approaches thus capture fundamentally different notions of context, temporal consistency versus spatial neighborhood support, and are complementary. Prophet (Li et al., 2025a) identifies a two-phase commitment risk pattern that motivates our dynamic threshold scheduling; further work addresses scoring pathologies (Huang et al., 2026a,b) and repetition caused by caching (Zhao et al., 2026a). Hong et al. (Hong et al., 2026) approach the same symptoms from the attention level, proposing training-free interventions to address repetition and visual grounding degradation; by contrast, CACD operates at the commitment-order level without modifying attention mechanisms.

Summary. Existing anchor-seeking methods either rely on the multi-step temporal history buffers of AHD (Zou et al., 2026) for contextual consistency, or restrict the decoding frontier via architectural spatial distance constraints as in Wavefront-Diffusion (Yang et al., 2026). Crucially, neither translates spatial proximity into a dynamic scoring signal, leaving bilateral neighbor support unmeasured during the commitment phase. Furthermore, block-based frameworks structurally isolate high-readiness candidates across boundaries, and confidence heterogeneity from multimodal integration remains unaddressed. By contrast, CACD introduces a global, single-pass bilateral neighbor proximity score, transforming structural constraints into a direct commitment signal. A broader discussion is provided in Appendix A.

3 Motivation

3.1 Local Semantic Connections Determine Prediction Quality

Let $\mathbf{x} = [x_1, \ldots, x_L]$ be the generation sequence, where position i is either a decoded token or

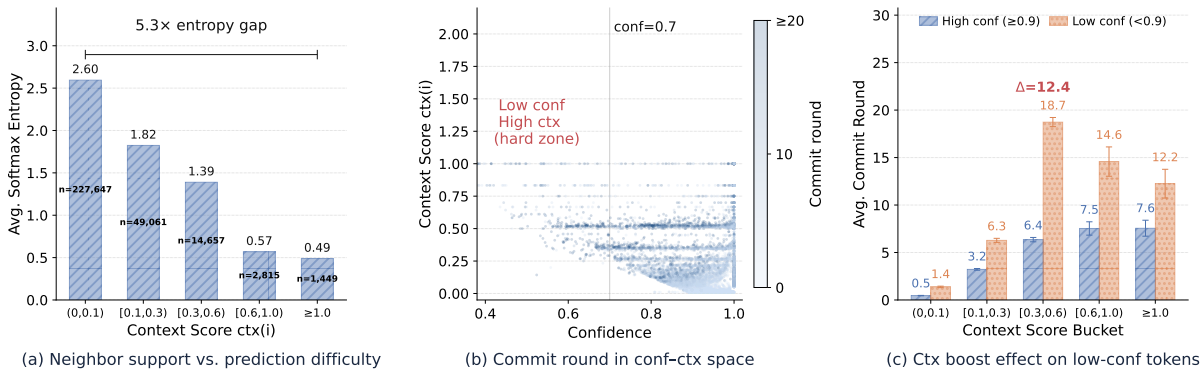


Figure 3: Motivation analysis on 1,000 MathVista samples using MMaDA. **(a)** Prediction entropy decreases by 5.3× from isolated positions to those with two decoded neighbors, showing that neighbor support provides information beyond raw confidence. **(b)** Tokens in the "hard zone" have low confidence but high context scores and are committed late under confidence-only scoring. Colors indicate the commitment round. **(c)** The commitment-round difference reaches Δ=12.4, showing how long locally supported positions remain masked.

[MASK]. In dMLLMs, visual inputs are encoded and prepended as conditioning context, with integration strategies varying across architectures. At each step the model performs a full-sequence forward pass for logits $\mathbf{l} \in \mathbb{R}^{L \times V}$; the confidence of position i is $c_i = \max_v \mathrm{softmax}(\mathbf{l}_i)_v$. A sampler selects a subset of masked positions to commit, repeating until all positions are decoded.

We define *contextual readiness* as the strength of local semantic connections a position shares with its already-decoded bilateral neighborhood. Figure 3a quantifies this over 295,629 masked positions: positions with both neighbors decoded achieve average entropy 0.49 versus 2.60 for isolated positions, a **5.3**× gap confirming that decoded neighbors provide strong additional signal beyond raw confidence (Kim et al., 2025).

From observation to design. This entropy gap directly motivates the composite score: a token with both neighbors decoded is far more predictable than an isolated one, yet pure confidence scoring treats them identically, committing the isolated token first if its confidence is higher. The composite score corrects this by incorporating neighbor proximity as a multiplicative amplifier on confidence, reflecting the measured entropy advantage.

The consequence of ignoring neighbor support. Pure confidence scoring strands contextually ready tokens below threshold despite having both neighbors decoded. These tokens persist as [MASK] voids: adjacent positions must predict across the gap using weaker, more distant context. We term this cascading effect *void compounding*: each misdeferred token creates a void that weakens local support for its neighbors, propagating further misdeferral monotonically with sequence length. As shown in Figure 3b, tokens with low confidence but high context score are systematically stranded in the "hard zone" under pure confidence scoring. Figure 3c quantifies the cost directly: the commit-round gap peaks at Δ=12.4 rounds, confirming that voids persist long enough to substantially degrade local semantic support for neighboring positions. During this interval, surrounding positions must predict across an expanding gap, driving their context scores toward zero and effectively reducing their scoring to pure confidence, the same regime that caused the original misdeferral. A step-by-step visualization is provided in Appendix B (Figure 7).

3.2 Block Structure Can Exclude Globally Preferred Anchor Candidates

Even with accurate scoring, block-based methods may fail to access high-readiness anchor tokens, whose exclusion leaves neighboring positions without strong local support. This is most damaging during the first decoding round, when committed tokens become the primary semantic anchors for all subsequent positions, as quantified in Figure 2c. Increasing block size cannot resolve this, as any fixed sequential partition creates boundaries where high-readiness anchors may be unavailable.

The two failure modes compound: block exclusion forces weaker tokens to be committed first, and each such commit creates a void that propagates through subsequent positions via the same void-compounding mechanism described in Sec-

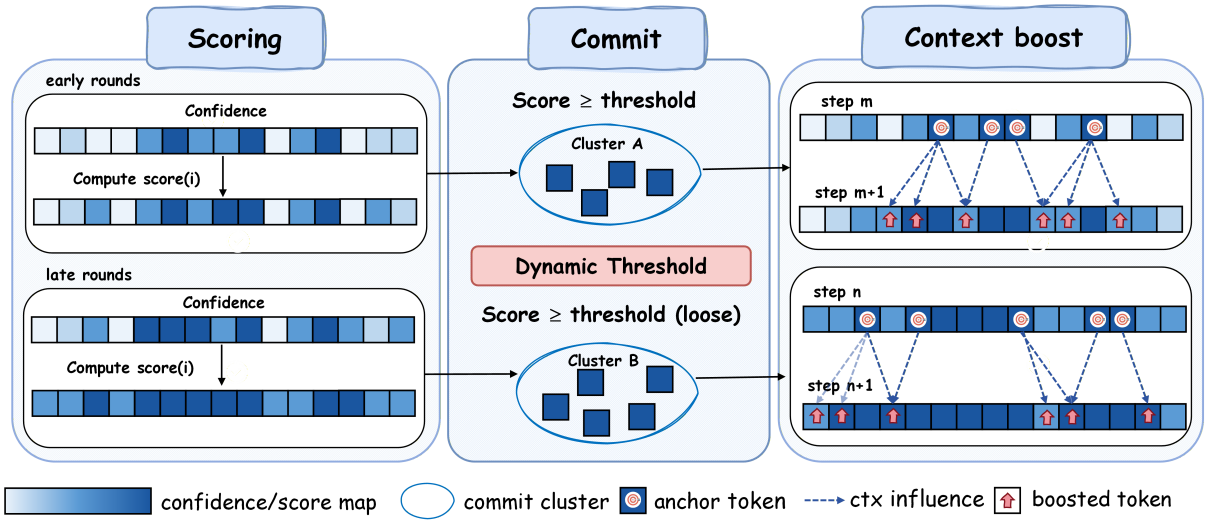


Figure 4: Overview of CACD. *Left*: the Scoring phase computes a composite score for each masked position across early and late rounds; early rounds apply a strict threshold τ, while late rounds apply a relaxed dynamic threshold triggered when $r_t \leq \delta$. *Center*: positions with composite scores exceeding the current threshold are committed simultaneously as a dense cluster. *Right*: committed anchor tokens propagate localized context influence (dashed lines) to adjacent masked positions, boosting their scores for the subsequent decoding round.

tion 3.1 — meaning degradation begins from the very first decoding round rather than accumulating gradually. Appendix B further illustrates how block boundaries delay high-readiness tokens despite favorable scores. CACD addresses both with a single unified approach, described in Section 4.

4 Method

CACD is built on a single principle: generation quality in dMLLMs depends on identifying high-quality semantic anchors and exploiting local semantic connections between decoded tokens and their masked neighbors. Two components instantiate this principle: a composite score (Section 4.1) and our cluster-based commitment approach (Section 4.2). Three calibration components ensure stable operation across architectures and output lengths (Section 4.3). Figure 4 illustrates the three-phase pipeline of CACD.

4.1 Context-Aware Scoring

To ensure contextually ready tokens are committed before isolated tokens of comparable confidence, CACD incorporates neighbor proximity directly into the commitment score.

Token Confidence.

$$c_i = \max_{v \in \mathcal{V}} \operatorname{softmax}(\mathbf{l}_i)_v. \tag{1}$$

Dynamic Context Score. Let $\mathcal{D} = \{j : x_j \neq [\text{MASK}]\}$ be decoded positions. Nearest-neighbor distances to decoded left and right neighbors:

$$d_\ell(i) = i - \max\{j \in \mathcal{D} : j < i\}, \tag{2}$$

$$d_r(i) = \min\{j \in \mathcal{D} : j > i\} - i, \tag{3}$$

with $d_\ell(i) = d_r(i) = \infty$ when no neighbor exists. The context score aggregates both via harmonic decay:

$$\operatorname{ctx}(i) = \frac{1}{1 + d_\ell(i)} + \frac{1}{1 + d_r(i)}, \quad \operatorname{ctx}(i) \in [0, 2]. \tag{4}$$

Composite Score. The composite score treats token confidence as a base weight and the context score as a multiplicative amplifier:

$$\operatorname{score}(i) = c_i \cdot \big(1 + \beta \cdot \operatorname{ctx}(i)\big). \tag{5}$$

The multiplicative form ensures contextual readiness scales proportionally with confidence: a high-confidence token adjacent to decoded neighbors receives a strong boost, while a low-confidence token gains only a modest increment, suppressing noisy predictions from being promoted on position alone. Crucially, a low-confidence token with strong bilateral support is still promoted above an equally confident isolated token, rescuing contextually ready candidates from the "hard zone" (Figure 3b). When no decoded neighbor exists, $\operatorname{ctx}(i)$=0 and the amplifier collapses to unity, reducing the score to pure

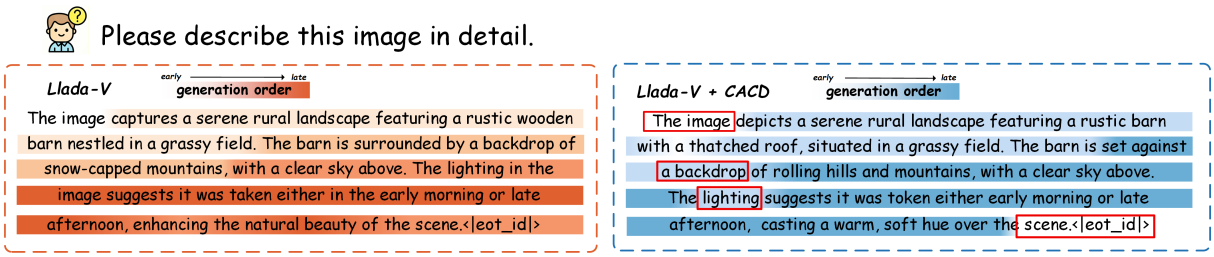


Figure 5: Decoding order on a representative captioning sample (lighter = earlier commit; red boxes = semantic anchors). Block diffusion frequently bypasses high-readiness anchors at structural boundaries; in contrast, CACD tends to first capture global anchors across the full sequence for initial decoding, and subsequently identifies critical local tokens within each sub-segment to propagate context outward.

confidence c_i and ensuring isolated positions remain rankable. β=0 recovers pure confidence scoring as a special case; Appendix D confirms that β=1.0 strikes a robust balance between confidence and contextual influence across models.

4.2 Full-Sequence Cluster Commitment

To keep high-readiness anchors globally accessible, CACD operates on the full sequence at every step: each round consumes 1 NFE, and all positions with $\text{score}(i) \geq \tau_t$ are committed as a cluster, immediately becoming anchors that reduce prediction difficulty for their neighbors. Figure 5 illustrates the resulting decoding order on a representative captioning sample. Crucially, comprehensive empirical evaluations in Appendix D show that CACD remains remarkably stable across a broad range of joint β and τ configurations, confirming that its performance gains are driven by its structural design rather than fragile hyperparameter selection.

4.3 Architecture-Aware Calibration

The evaluated models exhibit substantially different token confidence distributions. MMaDA has a mean confidence of 0.63, LaViDa has a bimodal distribution with a mean of 0.47, and LLaDA-V has a strongly left skewed distribution with a mean of 0.12, with more than 73% of tokens below c_i=0.1. Notably, LaViDa and LLaDA-V share the same SigLIP encoder but produce markedly different distributions. These observations motivate architecture specific calibration, while the complete distributions are provided in Appendix C.

Dynamic threshold scheduling. Inspired by the two-phase commitment risk pattern in Prophet (Li et al., 2025a), we employ a dynamic scheduling mechanism to shift from strict precision to completion. Early phases favor high thresholds to protect anchor quality and ensure initial accuracy, while later phases use lower thresholds to prevent deadlock:

$$\tau_t = \begin{cases} \tau & \text{if } r_t > \delta, \\ \tau - \dfrac{\delta - r_t}{\delta}(\tau - \tau_{\min}) & \text{otherwise,} \end{cases} \tag{6}$$

where $r_t = N^t_{\text{mask}}/L$ and $\tau_{\min} = 0.75\,\tau$.

Algorithm 1 CACD: Context-Aware Cluster Decoding (one round)

Require: Sequence $\mathbf{x}$; parameters $\tau, \beta, \delta, \rho$; EOS token id

$p \leftarrow N^t_{\text{dec,non-EOS}}/L$; $\epsilon \leftarrow \max(0,\, 1 - p/\rho)$
$r_t \leftarrow N^t_{\text{mask}}/L$; $\tau_t \leftarrow \textsc{Schedule}(\tau, r_t, \delta)$ *// Eq. 6*
$\mathbf{l} \leftarrow f_\theta(\mathbf{x})$ *// full-sequence forward pass, 1 NFE*
Compute c_i, $\text{ctx}(i)$, $\text{score}(i)$ for all masked i *// Eqs. 1–5*
$\text{score}_{\text{EOS}} \leftarrow \text{score}_{\text{EOS}} \cdot (1 - \epsilon)$ *// suppress premature EOS*
$\mathcal{C} \leftarrow \{i : \text{score}(i) \geq \tau_t,\ x_i = [\text{MASK}]\}$
if $\mathcal{C} = \emptyset$ **then**
  $\mathcal{C} \leftarrow \{\arg\max_i \text{score}(i)\}$ *// deadlock prevention*
end if
$x_i \leftarrow \hat{x}_i$ **for all** $i \in \mathcal{C}$

EOS soft suppression. EOS tokens accumulate spuriously high confidence before generation is complete, biasing the model toward short outputs. We suppress this by applying a progress-dependent penalty:

$$\epsilon = \max\left(0,\ 1 - \frac{p}{\rho}\right), \tag{7}$$

$$\text{score}_{\text{EOS}}(i) \leftarrow \text{score}_{\text{EOS}}(i) \cdot (1 - \epsilon), \tag{8}$$

where $p = N^t_{\text{dec,non-EOS}}/L$ is the fraction of non-EOS tokens decoded so far. For $p \geq \rho$ the penalty vanishes and EOS competes freely.

Table 1: Generation quality across three dMLLMs (within-model comparisons). **Bold**: best per model per metric. Underline: second best. LLaVABench-COCO for long-form generation and CHAIR for hallucination mitigation.

Model	Method	Understanding & Reasoning			Hallucination		
		MathVista↑	LLaVABench-COCO↑	ScienceQA↑	$CHAIR_s$↓	$CHAIR_i$↓	Recall↑
MMaDA-8B-Base							
	Original	23.50	35.30	50.24	24.58	11.27	39.46
	AdaBlock	24.80	47.40	51.61	22.84	11.27	**41.83**
	Wavefront	23.00	42.50	50.77	**18.62**	**9.43**	38.91
	CACD	**28.00**	**49.00**	**51.71**	19.76	10.28	40.63
LaViDa (SigLIP+LLaDA-8B)							
	Original	41.50	51.10	71.24	6.93	10.20	34.68
	AdaBlock	45.50	78.60	72.93	6.62	5.02	31.62
	Wavefront	45.30	82.10	72.09	5.87	**4.31**	31.72
	CACD	**45.70**	**88.70**	**73.03**	**5.08**	7.58	**35.31**
LLaDA-V (SigLIP+LLaDA-8B)							
	Original	30.60	50.30	79.23	5.60	3.95	36.55
	AdaBlock	31.70	72.90	79.18	5.82	4.17	37.05
	Wavefront	**33.50**	71.30	77.84	**5.21**	4.17	30.06
	CACD	**33.50**	**73.90**	**79.52**	5.38	**3.81**	**37.30**

By committing semantically anchored tokens first and propagating their local context outward, CACD helps each position decode with accurate local semantic support from its neighborhood.

5 Experiments

5.1 Experimental Setup

We evaluate on three dMLLMs: **MMaDA-8B-Base** (Yang et al., 2025), **LaViDa** (Li et al., 2025b), and **LLaDA-V** (You et al., 2025). We compare against three baselines: *Original* (standard fixed-step block-diffusion), *AdaBlock* (Lu et al., 2026) (ICLR 2026), and *Wavefront* (Yang et al., 2026) (ICLR 2026), originally proposed for text-only dLLMs, we evaluate them on a broad range of multimodal benchmarks. Benchmarks cover long-form generation and hallucination via *LLaVABench-COCO* (Liu et al., 2024) (Relative Score) and *CHAIR* (Rohrbach et al., 2018), while we evaluate reasoning and classification performance via *MathVista* (Lu et al., 2024) (LLM-judge accuracy) as well as *ScienceQA* (Lu et al., 2022).

5.2 Main Results

Table 1 reports generation quality across all models and benchmarks.

Long-form generation quality. CACD achieves the highest LLaVABench-COCO score across all three models: on LaViDa, CACD improves from 51.1 (Original) to 88.7 (+37.6); on LLaDA-V, from 50.3 to 73.9 (+23.6); on MMaDA, from 35.3 to 49.0 (+13.7). These gains confirm that neighbor-aware commitment consistently benefits long-form generation regardless of the underlying dMLLM architecture.

Hallucination reduction. Relative to Original, CACD reduces both CHAIR metrics on MMaDA: $CHAIR_s$ drops from 24.58 to 19.76 (-4.82), and $CHAIR_i$ from 11.27 to 10.28 (-0.99), although Wavefront performs better on both metrics. On LLaDA-V, CACD obtains the lowest $CHAIR_i$ (3.81) and the highest Recall (37.30), showing a favorable balance between object hallucination and coverage. Qualitative examples in Appendix G further illustrate that CACD can produce descriptions with fewer hallucinated objects while preserving relevant visual content.

Reasoning and classification. CACD achieves the highest MathVista accuracy on MMaDA (from 23.5% to 28.0%, +4.5%) and on LLaDA-V (from 30.6% to 33.5%, +2.9%), and delivers the best ScienceQA accuracy across all three models. Even on short outputs where void accumulation is minimal, committing the highest-readiness token first establishes a more reliable semantic foundation for subsequent positions, yielding modest but consistent quality gains across benchmarks.

Baseline comparison. AdaBlock and Wavefront underperform the original decoder on several model–benchmark combinations. For exam-

Table 2: Generation quality on LLaDA-8B-Instruct. All values are percentages. **Bold**: better result.

Benchmark	Metric	Original	CACD	Δ
GSM8K	Accuracy	61.00	**63.00**	+2.00
IFEval	Prompt strict	59.00	**60.00**	+1.00
	Instruction strict	64.74	**68.59**	+3.85
	Prompt loose	60.00	**61.00**	+1.00
	Instruction loose	67.95	**69.23**	+1.28
AlpacaEval	Win rate	4.50	**6.00**	+1.50
	Length controlled win rate	10.95	**11.45**	+0.50

ple, Wavefront obtains 23.0% on MMaDA MathVista, compared with 23.5% for the original decoder. The degradation is more pronounced on some long-form tasks. One possible explanation is that these methods were developed for text-only models and use confidence assumptions that do not transfer uniformly to dMLLMs.

Why existing decoding methods degrade on dMLLMs. Different visual integration strategies produce markedly different confidence distributions in dMLLMs, a variation not considered by text-only decoding methods (Figure 8). MMaDA's VQ-tokenizer yields a moderate mean token confidence of $\mu = 0.63$, whereas LLaDA-V's SigLIP integration produces a heavily left-skewed distribution with $\mu = 0.12$, with over 73% of its tokens falling below $c_i = 0.1$.

5.3 Extension to Text Only dLLMs

Although CACD is designed and primarily evaluated for multimodal diffusion models, its commitment rule depends only on token confidence and the positions of decoded neighbors. It does not require visual features or a model specific multimodal component. We therefore apply CACD to LLaDA-8B-Instruct and evaluate it on GSM8K, IFEval, and AlpacaEval.

As shown in Table 2, CACD improves all reported quality metrics over Original. Accuracy on GSM8K increases from 61.00 to 63.00. On IFEval, the largest improvement is observed for instruction level strict accuracy, which increases from 64.74 to 68.59. On AlpacaEval, the standard and length controlled win rates increase from 4.50 to 6.00 and from 10.95 to 11.45, respectively. These results provide preliminary evidence that context aware commitment can also benefit text generation.

Decoding efficiency. Original uses 128 fixed forward evaluations, whereas CACD dynamically determines the number of commitment rounds. Average NFE decreases to 48.00 on GSM8K, 112.73 on IFEval, and 121.75 on AlpacaEval, producing measured speedups of $2.68\times$, $1.20\times$, and $1.03\times$, respectively. Thus, CACD reduces NFE on all three benchmarks, although the corresponding runtime gain is modest on AlpacaEval. Complete efficiency results are provided in Appendix F.

Table 3: LLaVABench-COCO score and average output length. Output length alone does not account for the observed quality differences. **Bold**: best score per model.

Length	Original	AdaBlock	Wavefront	CACD
16	45.9	**52.6**	41.9	52.2
32	40.8	**61.7**	45.5	59.4
64	33.5	55.9	46.4	**58.8**
128	31.6	46.8	43.0	**50.4**
256	32.3	46.0	44.6	**47.3**
512	35.3	47.4	42.5	**49.0**
1024	33.0	45.5	43.2	**47.8**

5.4 Quality Scales with Sequence Length

Table 3 tests whether the quality advantage of CACD grows with output length. At 16 tokens, Original and CACD are close (45.9 vs. 52.2). As sequence length increases, the quality gap widens monotonically, reaching a 44.8% relative improvement at 1024 tokens (47.8 vs. 33.0) while Original continues to degrade, confirming that CACD specifically addresses the long-form quality degradation that block-based approaches increasingly suffer as generation extends.

Table 4 further confirms that CACD's quality gains are not attributable to longer outputs: Wavefront produces the longest outputs on MMaDA (402.8 chars) yet scores lower (42.5 vs. 49.0), ruling out length inflation as a confound.

Table 4: LLaVABench-COCO score and average output length. Higher score with shorter output confirms quality gains are not length-driven. **Bold**: best Score per model.

Method	MMaDA		LaViDa		LLaDA-V	
	Score↑	Len	Score↑	Len	Score↑	Len
Original	35.3	188.8	51.1	101.5	50.3	392.5
AdaBlock	47.4	160.2	78.6	256.1	72.9	409.7
Wavefront	42.5	402.8	82.1	481.8	71.3	400.9
CACD	**49.0**	158.1	**88.7**	461.2	**73.9**	367.5

Table 5: Component ablation on MMaDA LLaVABench-COCO. **Bold**: best.

Ctx	Dyn. τ	EOS	Score↑
✗	✓	✓	46.4
✓	✗	✓	46.7
✓	✓	✗	48.2
✓	✓	✓	**49.0**

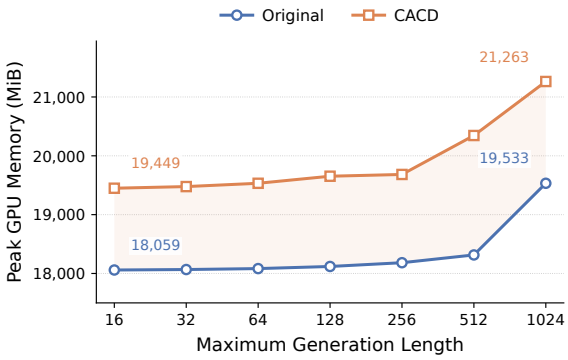


Figure 6: Peak GPU memory across maximum generation lengths on MMaDA, measured on an NVIDIA A800. Both methods show gradual memory growth, while CACD uses moderately more memory than Original across the evaluated lengths.

5.5 Ablation Study

Table 5 evaluates each component of CACD on MMaDA LLaVABench-COCO. Additional sensitivity results are provided in Appendix D. In these experiments, the context weight β=1.0 performs consistently well across all three architectures despite their different visual integration strategies.

Removing the context score (*w/o Ctx*) lowers the score from 49.0 to 46.4. In this setting, the composite score reduces to pure confidence, so tokens with stronger contextual support may remain deferred. Replacing the dynamic threshold schedule with a fixed threshold (*w/o Dyn.* τ) reduces the score to 46.7, reflecting the difficulty of balancing reliable early commitments with completion in later rounds. Without EOS suppression (*w/o EOS*), the score decreases to 48.2 because EOS may be committed before sufficient semantic content has been generated. Overall, each component provides a complementary contribution to the full method.

Computational cost. CACD achieves the lowest NFE on all three evaluated models. Figure 6 shows how peak GPU memory changes with the maximum generation length on MMaDA. Memory usage increases gradually for both methods, with CACD requiring approximately 1.4 to 2.0 GiB more memory than Original while substantially reducing the number of decoding rounds. Complete NFE, runtime, and memory comparisons across models are provided in Appendix E.

6 Conclusion

We presented CACD, a training-free decoding method that improves long-form generation quality in dMLLMs via a multiplicative confidence-context score that promotes contextually ready tokens regardless of their absolute confidence level, committing them as a cluster while operating block-free to keep high-readiness anchors globally accessible. Experiments across three dMLLMs and four benchmarks show consistent quality improvement and hallucination reduction, with the advantage compounding as outputs grow longer, confirming the void-compounding hypothesis that misdeferred tokens cascadingly degrade local semantic support downstream. A key finding is that visual integration strategies shape the confidence landscape in ways existing text-only decoding methods do not account for, a dimension that warrants explicit treatment in future dMLLM decoding method design.

7 Limitations

CACD has several practical limitations. Per-model threshold calibration is currently required because

the optimal τ varies with each architecture's confidence distribution; automatic calibration remains an important future direction. Although CACD reduces NFE, its runtime and memory benefits are model dependent, and integration with effective caching mechanisms may further improve overall efficiency. Finally, our experiments demonstrate cross-architecture generalization at the approximately 8B scale, but not cross-scale generalization, as larger dense dMLLMs are not currently publicly available. Evaluation at larger model scales is therefore left for future work.

References

Sudhanshu Agrawal, Risheek Garrepalli, Raghavv Goel, Mingu Lee, Christopher Lott, and Fatih Porikli. 2026. Spiffy: Multiplying diffusion LLM acceleration via lossless speculative decoding. *Preprint*, arXiv:2509.18085.

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. 2025. Block diffusion: Interpolating between autoregressive and diffusion language models. *CoRR*, abs/2503.09573.

Iskander Azangulov, Teodora Pandeva, Niranjani Prasad, Javier Zazo, and Sushrut Karmalkar. 2026. Parallel sampling from masked diffusion models via conditional independence testing. In *The Fourteenth International Conference on Learning Representations*.

Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, et al. 2025. Llada2. 0: Scaling up diffusion language models to 100b. *arXiv preprint arXiv:2512.15745*.

Changxiao Cai and Gen Li. 2026. Confidence-based decoding is provably efficient for diffusion language models. *Preprint*, arXiv:2603.22248.

Shuochen Chang, Xiaofeng Zhang, Qingyang Liu, and Li Niu. 2025. D3tom: Decider-guided dynamic token merging for accelerating diffusion mllms. *arXiv preprint arXiv:2511.12280*.

Kecheng Chen, Ziru Liu, Xijia Tao, Hui Liu, Xinyu Fu, Suiyun Zhang, Dandan Tu, Lingpeng Kong, Rui Liu, and Haoliang Li. 2025. Beyond confidence: Adaptive and coherent decoding for diffusion language models. ArXiv:2512.02044.

Jacob K. Christopher, Brian R. Bartoldson, Tal Ben-Nun, Michael Cardei, Bhavya Kailkhura, and Ferdinando Fioretto. 2025. Speculative diffusion decoding: Accelerating language generation through diffusion. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Justin Deschenaux and Caglar Gulcehre. 2024. Beyond autoregression: Fast LLMs via self-distillation through time. *CoRR*, abs/2410.21035.

Sujung Hong, Chanyong Yoon, and Seongjae Hwang. 2026. Mitigating mask prior drift and positional attention collapse in large diffusion vision-language models. *Preprint*, arXiv:2605.14530.

Jianuo Huang, Yaojie Zhang, Yicun Yang, Benhao Huang, Biqing Qi, Dongrui Liu, and Linfeng Zhang. 2025. Mask tokens as prophet: Fine-grained cache eviction for efficient dllm inference. *Preprint*, arXiv:2510.09309.

Pengcheng Huang, Tianming Liu, Zhenghao Liu, Yukun Yan, Shuo Wang, Tong Xiao, Zulong Chen, and Maosong Sun. 2026a. Empirical analysis of decoding biases in masked diffusion models. *Preprint*, arXiv:2508.13021.

Zemin Huang, Yuhang Wang, Zhiyang Chen, and Guo-Jun Qi. 2026b. Don't settle too early: Self-reflective remasking for diffusion language models. In *The Fourteenth International Conference on Learning Representations*.

Daniel Mingyi Israel, Guy Van den Broeck, and Aditya Grover. 2025. Accelerating diffusion LLMs via adaptive parallel decoding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Yuchu Jiang, Yue Cai, Xiangzhong Luo, Jiale Fu, Jiarui Wang, Chonghan Liu, and Xu Yang. 2026. d^2cache: Accelerating diffusion-based LLMs via dual adaptive caching. In *The Fourteenth International Conference on Learning Representations*.

Wonjun Kang, Kevin Galim, Seunghyuk Oh, Minjae Lee, Yuchen Zeng, Shuibai Zhang, Coleman Richard Charles Hooper, Yuezhou Hu, Hyung Il Koo, Nam Ik Cho, and Kangwook Lee. 2026. ParallelBench: Understanding the trade-offs of parallel decoding in diffusion LLMs. In *The Fourteenth International Conference on Learning Representations*.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. 2025. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In *Forty-second International Conference on Machine Learning*.

Minseo Kim, Chenfeng Xu, Coleman Hooper, Harman Singh, Ben Athiwaratkun, Ce Zhang, Kurt Keutzer, and Amir Gholami. 2026. CDLM: Consistency diffusion language models for faster sampling. *Preprint*, arXiv:2511.19269.

Pengxiang Li, Yefan Zhou, Dilxat Muhtar, Lu Yin, Shilin Yan, Li Shen, Yi Liang, Soroush Vosoughi, and Shiwei Liu. 2025a. Diffusion language models know the answer before decoding. In *NeurIPS 2025 Workshop on Efficient Reasoning*.

Shufan Li, Konstantinos Kallidromitis, Hritik Bansal, Akash Gokul, Yusuke Kato, Kazuki Kozuka, Jason Kuen, Zhe Lin, Kai-Wei Chang, and Aditya Grover. 2025b. Lavida: A large diffusion model for vision-language understanding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Aiwei Liu, Minghua He, Shaoxun Zeng, Sijun Zhang, Linhao Zhang, Chuhan Wu, Wei Jia, Yuan Liu, Xiao Zhou, and Jie Zhou. 2025. Wedlm: Reconciling diffusion language models with standard causal attention for fast inference. *Preprint*, arXiv:2512.22737.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Lingkun Long, Yushi Huang, Shihao Bai, Ruihao Gong, Jun Zhang, Ao Zhou, and Jianlei Yang. 2026. FocusdLLM: Accelerating long-context diffusion LLM inference via confidence-guided context focusing. *CoRR*, abs/2602.02159.

Aaron Lou, Chenlin Meng, and Stefano Ermon. 2024. Discrete diffusion language modeling by estimating the ratios of the data distribution.

Guanxi Lu, Hao Mark Chen, Yuto Karashima, Zhican Wang, Daichi Fujiki, and Hongxiang Fan. 2026. Adablock-dLLM: Semantic-aware diffusion LLM inference via adaptive block size. In *The Fourteenth International Conference on Learning Representations*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *The Twelfth International Conference on Learning Representations*.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *CoRR*, abs/2209.09513.

Nikita Mounier and Parsa Idehpour. 2025. Review, remask, refine: Process-guided block diffusion for text generation. In *ICML 2025 Workshop on Methods and Opportunities at Small Scale*.

Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. 2024. Scaling up masked diffusion models on text. *CoRR*, abs/2410.18514.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, JUN ZHOU, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. 2025. Large language diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. 2024. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *CoRR*, abs/2406.03736.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Yair Schiff, Edgar Mariano Marroquin, Justin T Chiu, Alexander M Rush, and Volodymyr Kuleshov. 2024. Simple and effective masked diffusion language models. In *NeurIPS 2024 Workshop on Mathematics of Modern Machine Learning*.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. 2024. Simplified and generalized masked diffusion for discrete data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yingte Shu, Yuchuan Tian, Chao Xu, Yunhe Wang, and Hanting Chen. 2026. Deferred commitment decoding for diffusion language models. ArXiv:2601.02076.

Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, Yuwei Fu, Jing Su, Ge Zhang, et al. 2025. Seed diffusion: A large-scale diffusion language model with high-speed inference. *Preprint*, arXiv:2508.02193.

Alexander Swerdlow, Mihir Prabhudesai, Siddharth Gandhi, Deepak Pathak, and Katerina Fragkiadaki. 2025. Unified multimodal discrete diffusion. *arXiv preprint arXiv:2503.20853*.

Runchu Tian, Junxia Cui, Xueqiang Xu, Feng Yao, and Jingbo Shang. 2025. Finish first, perfect later: Test-time token-level cross-validation for diffusion large language models.

Xu Wang, Chenkai Xu, Yijie Jin, Jiachun Jin, Hao Zhang, and Zhijie Deng. 2025. Diffusion LLMs can do faster-than-AR inference via discrete diffusion forcing. *CoRR*, abs/2508.09192.

Yunhe Wang, Kai Han, Huiling Zhen, Yuchuan Tian, Hanting Chen, Yongbing Huang, Yufei Cui, Yingte Shu, Shan Gao, Ismail Elezi, Roy Vaughan Miles, Songcen Xu, Feng Wen, Chao Xu, Sinan Zeng, and Dacheng Tao. 2026. Top 10 open challenges steering the future of diffusion language model and its variants. *Preprint*, arXiv:2601.14041.

Qingyan Wei, Yaojie Zhang, Zhiyuan Liu, Puyu Zeng, Yuxuan Wang, Biqing Qi, Dongrui Liu, and Linfeng Zhang. 2026. Accelerating diffusion large language models with slowfast sampling: The three golden principles. In *The Fourteenth International Conference on Learning Representations*.

Chengyue Wu, Hao Zhang, Shuchen Xue, Shizhe Diao, Yonggan Fu, Zhijian Liu, Pavlo Molchanov, Ping Luo, Song Han, and Enze Xie. 2026a. Fast-dLLM v2: Efficient block-diffusion LLM. In *The Fourteenth International Conference on Learning Representations*.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. 2026b. Fast-dLLM: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding. In *The Fourteenth International Conference on Learning Representations*.

Yanzheng Xiang, Lan Wei, Yizhen Yao, Qinglin Zhu, Hanqi Yan, Chen Jin, Philip Alexander Teare, Dandan Zhang, Lin Gui, Amrutha Saseendran, and Yulan He. 2026. Stop the flip-flop: Context-preserving verification for fast revocable diffusion decoding. *Preprint*, arXiv:2602.06161.

Haojin Yang, Rui Hu, Zequn Sun, Rui Zhou, Yujun Cai, and Yiwei Wang. 2026. Wavefrontdiffusion: Dynamic decoding schedule for improved reasoning. In *The Fourteenth International Conference on Learning Representations*.

Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. 2025. MMaDA: Multimodal large diffusion language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jiacheng Ye, Shansan Gong, Jiahui Gao, Junming Fan, Shuang Wu, Wei Bi, Haoli Bai, Lifeng Shang, and Lingpeng Kong. 2025a. Dream-VL & dream-VLA: Open vision-language and vision-language-action models with diffusion language model backbone. *CoRR*, abs/2512.22615.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. 2025b. Dream 7b: Diffusion large language models. *Preprint*, arXiv:2508.15487.

Zebin You, Shen Nie, Xiaolu Zhang, Jun Hu, Jun Zhou, Zhiwu Lu, Ji-Rong Wen, and Chongxuan Li. 2025. LLaDA-V: Large language diffusion models with visual instruction tuning. *CoRR*, abs/2505.16933.

Runpeng Yu, Qi Li, and Xinchao Wang. 2025a. Discrete diffusion in large language and multimodal models: A survey. *CoRR*, abs/2506.13759.

Runpeng Yu, Xinyin Ma, and Xinchao Wang. 2025b. Dimple: Discrete diffusion multimodal large language model with parallel decoding. *arXiv preprint arXiv:2505.16990*.

Lunbin Zeng, Jingfeng Yao, Bencheng Liao, Hongyuan Tao, Wenyu Liu, and Xinggang Wang. 2026. Diffusionvl: Translating any autoregressive models into diffusion vision language models. *Preprint*, arXiv:2512.15713.

Xiaofeng Zhang, Yihao Quan, Chen Shen, Chaochen Gu, Xiaosong Yuan, Shaotian Yan, Jiawei Cao, Hao Cheng, Kaijie Wu, and Jieping Ye. 2025a. Shallow focus, deep fixes: Enhancing shallow layers vision attention sinks to alleviate hallucination in LVLMs. In *Knowledgeable Foundation Models at ACL 2025*.

Xiaofeng Zhang, Chen Shen, Xiaosong Yuan, Shaotian Yan, Liang Xie, Wenxiao Wang, Chaochen Gu, Hao Tang, and Jieping Ye. 2024a. From redundancy to relevance: Enhancing explainability in multimodal large language models. *NAACL*.

Xiaofeng Zhang, Fanshuo Zeng, and Chaochen Gu. 2024b. Simignore: Exploring and enhancing multimodal large model complex reasoning via similarity computation. *Neural Networks*, page 107059.

Xiaofeng Zhang, Fanshuo Zeng, Yihao Quan, Zheng Hui, and Jiawei Yao. 2025b. Enhancing multimodal large language models complex reason via similarity computation. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Xiaofeng Zhang, Yuanchao Zhu, Chaochen Gu, Jiawei Cao, Hao Cheng, and Kaijie Wu. 2026a. What drives attention sinks? a study of massive activations and rotational positional encoding in large vision–language models. *Information Processing & Management*, 63(2):104431.

Xiaofeng Zhang, Yuanchao Zhu, Chaochen Gu, Xiaosong Yuan, Qiyan Zhao, Jiawei Cao, Feilong Tang, Sinan Fan, Yaomin Shen, Chen Shen, et al. 2026b. Hallucination begins where saliency drops. In *The Fourteenth International Conference on Learning Representations*.

Yu Zhang, Xinchen Li, Jialei Zhou, Hongnan Ma, Zhongwei Wan, Yiwei Shi, Duoqian Miao, Qi Zhang, and Longbing Cao. 2026c. Swordsman: Entropy-driven adaptive block partition for efficient diffusion language models. *CoRR*, abs/2602.04399.

Qiyan Zhao, Xiaofeng Zhang, Shuochen Chang, Qianyu Chen, Xiaosong Yuan, Xuhang Chen, Luoqi Liu, Jiajun Zhang, Xu-Yao Zhang, and Da-Han Wang. 2026a. Context tokens are anchors: Understanding the repetition curse in dmllms from an information flow perspective. *CoRR*, abs/2601.20520.

Qiyan Zhao, Xiaofeng Zhang, Shuochen Chang, Qianyu Chen, Xiaosong Yuan, Xuhang Chen, Luoqi Liu, Jiajun Zhang, Xu-Yao Zhang, and Da-Han Wang. 2026b. Context tokens are anchors: Understanding the repetition curse in dmllms from an information flow perspective. In *The Fourteenth International Conference on Learning Representations*.

Yunxiao Zhao and Changxiao Cai. 2026. Adaptation to intrinsic dependence in diffusion language models. *Preprint*, arXiv:2602.20126.

Liang Zheng, Bowen Shi, Yitao Hu, Jiawei Zhang, Ruofan Li, Sheng Chen, Wenxin Li, and Keqiu Li. 2026.

Mosaic: Unlocking long-context inference for diffusion llms via global memory planning and dynamic peak taming. *Preprint*, arXiv:2601.06562.

Cai Zhou, Chenyu Wang, Dinghuai Zhang, Shangyuan Tong, Yifei Wang, Stephen Bates, and Tommi Jaakkola. 2025. Next semantic scale prediction via hierarchical diffusion language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Shun Zou, Yong Wang, Zehui Chen, Lin Chen, Chongyang Tao, Feng Zhao, and Xiangxiang Chu. 2026. Breaking block boundaries: Anchor-based history-stable decoding for diffusion large language models. *CoRR*, abs/2604.08964.

Haomin Zuo, Yidi Li, Luoxiao Yang, and Xiaofeng Zhang. 2026. Diffusion-cam: Faithful visual explanations for dmllms. *arXiv preprint arXiv:2604.11005*.

A Extended Related Work

Masked Diffusion Foundations and the Broader dLLM Ecosystem. The probabilistic backbone of modern dLLMs is established by foundational works on discrete and masked diffusion, including SEDD (Lou et al., 2024), MD4 (Shi et al., 2024), MDLM (Sahoo et al., 2024), RADD (Ou et al., 2024), and SMDM (Nie et al., 2024), which together develop the training objectives, scaling recipes, and theoretical properties that 8B+ models such as LLaDA build upon. At the frontier, Dream-7B (Ye et al., 2025b), LLaDA2.0 (Bie et al., 2025), and SeedDiffusion (Song et al., 2025) scale masked diffusion to larger parameter counts, while DiffusionVL (Zeng et al., 2026) and Dream-VL (Ye et al., 2025a) extend it to vision-language tasks beyond the three dMLLMs we evaluate. Surveys by Yu et al. (2025a) and Wang et al. (2026) map the full landscape, with open challenges around long-form coherence and decoding quality directly motivating this work.

Decoding Structure: Block Variants and Commitment Strategies. Beyond the methods discussed in the main text, Swordsman (Zhang et al., 2026c) uses entropy signals to set block size adaptively; R3 (Mounier and Idehpour, 2025) applies a review-remask-refine cycle within blocks; and D2F (Wang et al., 2025) introduces block-wise autoregressive generation to enable KV-cache reuse with inter-block parallel decoding. On the commitment-order side, COVER (Xiang et al., 2026), Parallel Sampling (Azangulov et al., 2026), and Finish First, Perfect Later (Tian et al., 2025) propose distinct criteria for token commitment. Zhao and Cai (Zhao and Cai, 2026) show theoretically that schedules adapting to the intrinsic dependence structure of the target distribution yield improved convergence, grounding dependency-aware strategies such as CACD. Speculative decoding (Christopher et al., 2025; Agrawal et al., 2026) offers a complementary acceleration path without modifying commitment order.

Efficiency: Caching, Distillation, and Consistency. Several methods address KV-cache reuse under diffusion's bidirectional attention (Wu et al., 2026a; Jiang et al., 2026; Long et al., 2026; Zheng et al., 2026; Huang et al., 2025); notably, WeDLM (Liu et al., 2025) replaces bidirectional attention with causal attention via topological reordering, making prefix KV-cache compatible with parallel diffusion decoding. Distillation and consistency training (Deschenaux and Gulcehre, 2024; Kim et al., 2026) reduce required forward passes by training the model to denoise in fewer steps, providing orthogonal strategies to further accelerate generation that are fully complementary to our inference-time improvements. More broadly, developing effective KV-cache mechanisms tailored for dMLLMs remains an active area of research.

B Decoding Order Comparison

Figure 7 compares the two decoding strategies on a representative sample. In block diffusion (left), the high-confidence token (0.95) remains inaccessible because it lies outside the active block. Lower-confidence tokens within the block are therefore committed first, producing the *void-compounding* effect described in Section 3.1. In contrast, CACD (right) considers the full sequence in every round. Committed anchors provide immediate contextual support to neighboring masked positions and increase their composite scores, shown as "+ctx" increments. This process forms a dense commitment front supported by local context and reduces the propagation of generation errors.

C Confidence Distribution Analysis

Figure 8 presents the complete confidence distributions for the three models. MMaDA has a relatively uniform distribution with a mean confidence of 0.63, while LaViDa exhibits a bimodal distribution with a mean of 0.47. LLaDA-V has a strongly left skewed distribution with a mean of 0.12, and more than 73% of its tokens have confidence below c_i=0.1. These differences explain why a single

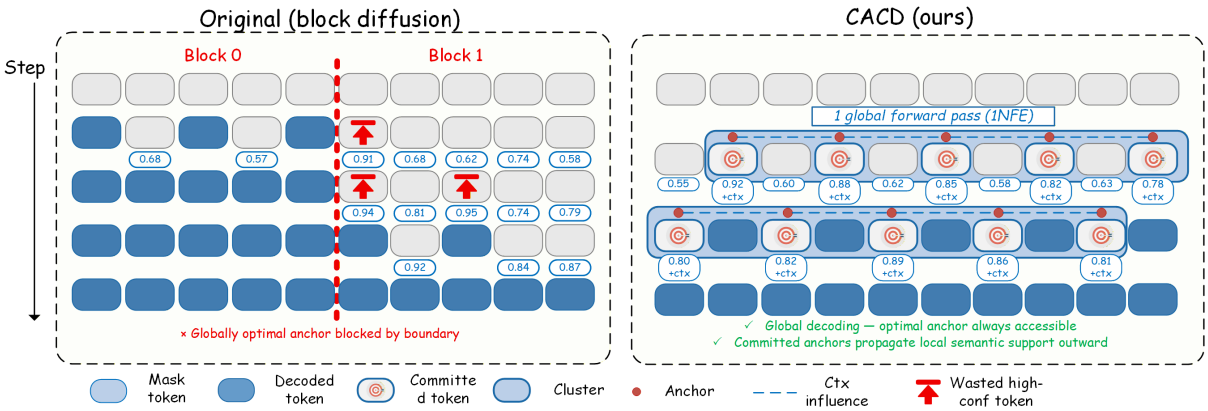


Figure 7: Decoding order comparison: block diffusion (left) vs. CACD (right). Block diffusion commits tokens within rigid sequential blocks, leaving the high-readiness anchor structurally inaccessible until its specific block is reached. CACD performs a global forward pass each round, committing the highest-readiness tokens as a dense cluster with context-amplified scores.

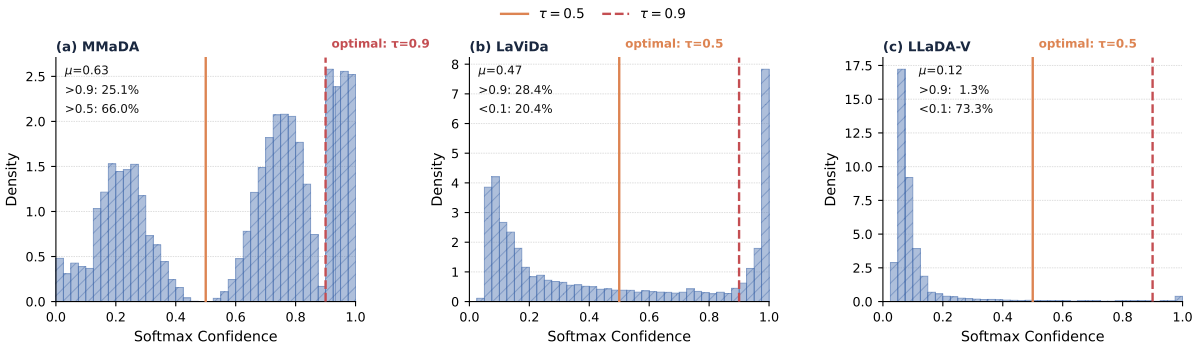


Figure 8: Token confidence distributions measured on 100 MathVista samples per model. MMaDA, LaViDa, and LLaDA-V exhibit substantially different distributions. LaViDa and LLaDA-V share the same SigLIP encoder, suggesting that the confidence landscape also depends on other model and training design choices.

raw confidence threshold does not transfer reliably across the evaluated architectures.

D Hyperparameter Sensitivity

Effect of β. Table 6 shows β=0 consistently underperforms β=1.0; values β>1.0 yield diminishing returns, as over-weighting neighborhood context suppresses the contribution of a token's own confidence, which remains an essential signal for commitment quality. We adopt β=1.0 as the optimal balance between the two.

Table 7: Effect of τ on MathVista accuracy. Architecture-specific thresholds are required: the optimal τ differs substantially across models due to confidence distribution heterogeneity. **Bold**: best.

Model	τ	Acc↑
MMaDA (8B-Base)	0.5	21.1
	0.7	24.3
	0.9	**28.0**
LaViDa (SigLIP+LLaDA-8B)	**0.5**	**45.7**
	0.7	44.6
	0.9	44.5
LLaDA-V (SigLIP+LLaDA-8B)	**0.5**	**33.5**
	0.7	33.3
	0.9	32.8

$\beta \times \tau$ interaction. Figure 9 reports a grid search over $\beta \in \{0, 0.5, 1.0, 1.5\}$ and $\tau \in \{0.5, 0.7, 0.9\}$ on MMaDA MathVista. Accuracy increases monotonically with τ for fixed $\beta \geq 0.5$, confirming inde-

Table 6: Effect of β on LLaVABench-COCO quality. β=0 recovers pure confidence scoring; β=1.0 provides the best context amplification across all three models. **Bold**: best.

Model	β	Score↑
MMaDA (8B-Base)	0.0	48.2
	0.5	48.7
	1.0	**49.0**
	1.5	48.9
	2.0	48.6
LaViDa (SigLIP+LLaDA-8B)	0.0	79.3
	0.5	82.3
	1.0	**88.7**
	1.5	81.3
	2.0	77.5
LLaDA-V (SigLIP+LLaDA-8B)	0.0	67.1
	0.5	70.6
	1.0	**73.9**
	1.5	71.7
	2.0	71.3

pendent tuning. Non-monotonic behavior at β=0 is attributable to the absence of context weighting: without neighbor support in the scoring signal, pure confidence alone is insufficient to maintain stable commitment ordering, and force-commit events occasionally promote low-quality tokens. The optimal β=1.0, τ=0.9 achieves the highest accuracy (28.0%). This interaction also clarifies why β generalizes across architectures while τ does not. The context score $\mathrm{ctx}(i) \in [0, 2]$ is a purely relative, positional quantity independent of absolute confidence values; its amplification effect therefore remains consistent regardless of how a model's visual integration strategy shapes the confidence distribution. By contrast, τ operates directly on raw confidence and is thus sensitive to distributional shift across architectures, as confirmed by the marked difference in optimal τ across models (Table 7) and the confidence heterogeneity documented in Appendix C.

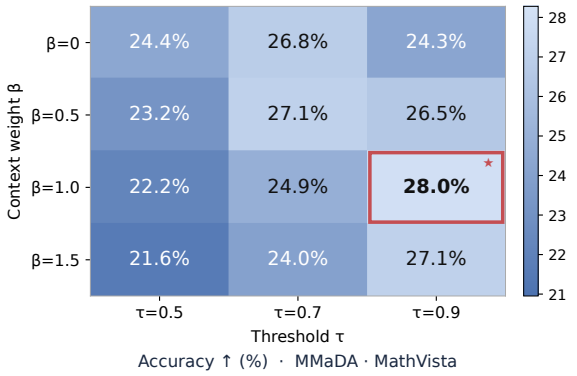


Figure 9: $\beta \times \tau$ interaction on MMaDA MathVista (accuracy %). Red box: adopted configuration (β=1.0, τ=0.9).

Effect of δ. Too-small δ increases deadlock; too-large δ admits low-confidence tokens early. δ=0.3 is the universal default (Table 8).

Table 8: Effect of δ on MMaDA MathVista. **Bold**: selected default.

δ	Acc↑
0.1	26.9
0.2	27.8
0.3	**28.0**
0.5	26.8

Effect of ρ. Table 9 ablates the EOS release ratio ρ on MMaDA LLaVABench-COCO. Score is stable across ρ while output length increases monotonically, indicating output length is governed by the VQ-tokenizer training prior rather than ρ. We adopt ρ=0.8 as the universal default.

Table 9: Effect of EOS release ratio ρ on MMaDA LLaVABench-COCO. **Bold**: selected default.

ρ	Score↑	Avg. Length (chars)
0.5	47.7	154.9
0.6	48.1	157.8
0.8	**49.0**	158.1
1.0	47.8	161.0

E Decoding Cost

We compare the number of function evaluations, denoted as NFE, and the measured decoding speed of all methods. The Original decoder uses a fixed number of forward evaluations determined by the maximum generation length. By contrast, CACD performs a full sequence forward pass in each round and can commit multiple positions simultaneously.

As shown in Table 10, CACD substantially reduces NFE on MMaDA across all three benchmarks. On LLaVABench-COCO with a maximum generation length of 512 tokens, it reduces NFE from 512.0 to 45.9 and provides a measured speedup of 11.2×. This reduction results from the cluster commitment procedure, which often allows multiple positions to be committed within the same decoding round.

Table 10: Decoding cost on MMaDA-8B-Base without KV caching. NFE denotes the number of function evaluations and Spd. denotes the measured speedup. **Bold**: best.

Method	MathVista		LLaVABench-COCO		ScienceQA	
	NFE↓	Spd.↑	NFE↓	Spd.↑	NFE↓	Spd.↑
Original	128.0	1.0×	512.0	1.0×	16.00	1.0×
AdaBlock	25.9	3.9×	74.0	6.5×	1.62	**6.6×**
Wavefront	60.0	3.8×	128.1	4.0×	4.00	3.4×
CACD	**16.8**	**4.5×**	**45.9**	**11.2×**	**1.32**	5.1×

Table 11 reports the corresponding NFE results on LaViDa and LLaDA-V. CACD requires the fewest forward evaluations on every evaluated benchmark for both models, although the magnitude of the reduction varies across models and tasks.

Table 11: NFE on LaViDa and LLaDA-V. Lower values are better. **Bold**: best.

Method	MathVista	LLaVABench-COCO	ScienceQA
LaViDa			
Original	128.0	512.0	16.0
AdaBlock	27.4	52.1	2.1
Wavefront	36.0	71.3	3.0
CACD	**16.0**	**45.6**	**1.1**
LLaDA-V			
Original	128.0	512.0	16.0
AdaBlock	45.5	226.6	7.2
Wavefront	64.2	256.1	8.2
CACD	**33.1**	**164.2**	**6.5**

E.1 Peak GPU Memory

We additionally measure peak GPU memory on LLaVABench-COCO using `max_new_tokens=512` on an NVIDIA A800 GPU. Table 12 reports NFE, measured decoding speed, and peak memory under the same evaluation setting.

The practical tradeoffs depend on the model. On MMaDA, CACD provides the largest speedup, while Original uses less peak memory. On LaViDa, CACD requires the fewest forward evaluations and uses the least peak memory, while AdaBlock is slightly faster. On LLaDA-V, CACD again requires the fewest forward evaluations and uses the least peak memory, while AdaBlock provides the largest speedup. Thus, the NFE reduction of CACD does not imply uniform superiority in runtime or memory usage.

F Text Only Decoding Efficiency

Table 13 reports the decoding efficiency results for the text generation experiments in Section 5.3.

Table 12: Decoding efficiency and peak GPU memory on LLaVABench-COCO using `max_new_tokens=512` on an NVIDIA A800. Memory is reported in MiB. The best result for each model and metric is shown in **bold**.

Model	Method	NFE↓	Spd.↑	Memory↓
MMaDA	Original	512.0	1.0×	**18,315**
	AdaBlock	74.0	6.5×	22,775
	Wavefront	128.1	4.0×	21,157
	CACD	**45.9**	**11.2×**	20,345
LaViDa	Original	512.0	1.0×	33,737
	AdaBlock	52.1	**2.1×**	29,205
	Wavefront	71.3	1.3×	29,519
	CACD	**45.6**	2.0×	**28,585**
LLaDA-V	Original	512.0	1.0×	34,175
	AdaBlock	226.6	**2.6×**	36,509
	Wavefront	256.1	1.1×	26,703
	CACD	**164.2**	1.8×	**26,529**

CACD reduces NFE on all three benchmarks, although the resulting speedup varies across tasks and output structures.

Table 13: Decoding efficiency on LLaDA-8B-Instruct. Lower NFE and higher speedup are better.

Benchmark	Original NFE	CACD NFE	Speedup
GSM8K	128.00	**48.00**	**2.68×**
IFEval	128.00	**112.73**	**1.20×**
AlpacaEval	128.00	**121.75**	**1.03×**

G Qualitative Case Study

Figures 10, 11, and 12 compare the Original decoding method and CACD on representative MathVista, CHAIR, and LLaVABench-COCO samples.

Case A:Mathvista max_new_tokens=128

What is the total volume of the measuring cup? (Unit: g)

Ground Truth: 1000

Input image

Llada-V *69 words*

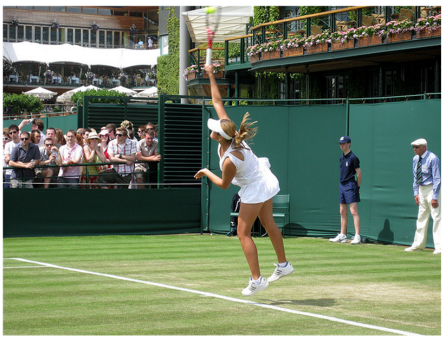
The measuring cup is filled with flour, and the weight of the flour is 500g. The capacity of the measuring cup is 1000g. The density of the measuring cup is 1g/cm^3. The volume of the measuring cup is 500g / 1g/cm^3 = 500 cm^3. The capacity of the measuring cup is 1000g. Therefore, the total volume of the measuring cup is 500g. So the answer is 500.

Llada-V + CACD *92 words*

To determine the total volume of the measuring cup, we need to consider the markings visible on the cup. The measuring cup is marked with measurements in grams (g) and liters (L). The markings indicate that the cup has a capacity of 1000 g or 1 L. Since the question asks for the total volume in grams, we can use the value marked on the top side of the cup, which is 1000 g. Therefore, the total volume of the measuring cup is 1000 g. So the answer is 1000.

Case B :Chair

Input image



Llada-V

The image captures a lively scene of a frisbee game, with a woman in a white shirt reaching up to catch it frisbee.

Llada-V + CACD

The image features a woman playing tennis on a court, holding a tennis racket in her hand. She is in the middle of a serve, preparing to hit the ball.

Case C :Chair

Input image

Llada-V

The image features a brown fire hydrant sitting on a sidewalk next to a yellow line. The fire hydrant is positioned in the middle of road.

Llada-V + CACD

The image features a yellow fire hydrant situated on the sidewalk of a city street. The fire hydrant is positioned close to a yellow line and a blue sign.

Figure 10: **Case A: MathVista — Measuring Cup.** *LLaDA-V*: reasoning drift produces the wrong answer (500). *LLaDA-V* + CACD: anchors the visual token ("1000g/1L") as the first committed cluster and propagates context outward, yielding the correct answer (1000). **Case B: CHAIR — Tennis Court.** *LLaDA-V*: object hallucination produces an incorrect scene description ("frisbee game"). *LLaDA-V* + CACD: anchors the visual token ("tennis court / racket") and propagates context outward, yielding the correct description (woman playing tennis, preparing to serve). **Case C: CHAIR — Fire Hydrant.** *LLaDA-V*: attribute hallucination produces an incorrect spatial description ("positioned in the middle of road"). *LLaDA-V* + CACD: anchors the visual token ("sidewalk / yellow line") and propagates context outward, yielding the correct description (hydrant on the sidewalk, close to a yellow line and a blue sign).

Case D: Llavabench-COCO max_new_tokens=512

Input image

Caption: A clock on a short tower and potted plants along the sidewalk

Question: Imagine you are walking down this street; how would you know the time or which direction the sidewalk leads?

gpt ans: The time can be read from the tall clock mounted on a pole along the sidewalk. The street direction is indicated by the tree-lined path with potted plants, parked bicycles, and cars leading further down.

Llada-V ***401 words***

To determine the time and direction, I would look at the large clock labeled "BRISTOW'S AT THEROADWAY" and the pedestrian crossing sign, which shows a pedestrian figure pointing to the right...
To determine the time and direction the sidewalk leads, I would look at the large clock on the pole and the pedestrian crossing sign. The clock, labeled "BRISTOW'S AT THEROADWAY," has a white face with black numerals and hands, and the pedestrian crossing sign shows a pedestrian figure pointing to the right, indicating the direction the sidewalk leads.
To determine the time and direction the sidewalk leads, I would look at the large clock on the pole and the pedestrian crossing sign. The clock, labeled "BRISTOW'S AT THEROADWAY," has a white face with black numerals and hands, and the pedestrian crossing sign shows a pedestrian figure pointing to the right, indicating the direction the sidewalk leads.
To determine the time and direction the sidewalk leads, I would look at the large clock on the pole and the pedestrian crossing sign. The clock, labeled "BRISTOW'S AT THEROADWAY," has a white face with black numerals and hands, and the pedestrian crossing sign shows a pedestrian figure pointing to the right, indicating the direction the sidewalk leads.
The sidewalk leads to the right.

Llada-V + CACD ***392 words***

The clock reads "BRISTOW'S AT THE BROAD" and displays the time as 2:10...
Time: Look for the clock on the pole showing 2:10. You can also use a smartphone or ask a local.

Direction: There is a yellow pedestrian crossing sign indicating a crossing area. Follow the direction of the sidewalk... You can also use the sun to determine direction.

By using these methods, you can easily determine the time and which direction the sidewalk leads.

Figure 11: **Case D: LLaVABench-COCO — Clock Tower (doc_id=41).** *Original*: repeated paragraphs and missing time readings. CACD: well-structured response with specific readings and multiple candidate interpretations.

Case E: Llavabench-COCO max_new_tokens=512

Input image

Caption: Two people who are taking apart a video game controller.

Question: What are the key elements in this picture?

gpt ans: Two people at a wooden table disassembling Wii remote controllers, possibly replacing batteries. A laptop is visible on the left side of the table.

Llada-V *276 words*

The image depicts three hands holding white Wii remote controllers, disassembled to reveal green circuit boards with resistors, capacitors, and microcontrollers. The remotes are placed on a wooden surface. The background appears to be a workshop setting...
The image depicts a close-up view of three hands, each holding a white Wii remote controller. The remotes are disassembled, revealing their internal components. The hands are positioned over a wooden surface, which appears to be a table or desk.
The internal components of the remotes are visible, including a green circuit board with various components and connectors. The circuit board is populated with numerous small electronic components, including resistors, capacitors, and possibly microcontrollers. The remotes have buttons on their front sides, which are not functional in this disassembled state.
The background of the image is blurred, but it appears to be a workshop or a similar setting. The presence of various tools and equipment suggests that the individuals might be engaged in some form of technical or repair work

Llada-V + CACD *432 words*

The image depicts hands holding Wii remote controllers in a DIY workshop environment. The controllers are disassembled, revealing green circuit boards with capacitors and resistors.
The wooden surface is covered with a protective sheet...
In the background, a plastic bag and a cardboard box can be seen, hinting at a workspace filled with tools and equipment. The bag and box are not just objects but contribute to the story of the scene...
It's a snapshot of exploration and discovery — a testament to the determination to uncover the secrets hidden within the circuit boards.

Figure 12: **Case E: LLaVABench-COCO — Wii Controller (doc_id=49).** *Original*: verbatim repetition and truncation. CACD: complete, non-repetitive response with broader scene coverage including background objects absent from the Original output.