

Concurrent Image Understanding and Generation: Self-Correcting Coupled Markov Jump Processes

Minh-Quan Le^{1,3,†}, Armand Comas¹, Alexandros Lattas¹, Stylianos Moschoglou¹, Pedro Véllez², Amit Raj², Aaron Germuth¹, Thabo Beeler¹, Dimitris Samaras³ and Di Qiu¹

¹Google, ²Google DeepMind, ³Stony Brook University

Human cognition does not separate understanding and generation. A teacher at a whiteboard speaks and draws *together*, each modality reshapes the other. In this paper, we bring this coupled loop to artificial systems. Masked Diffusion Models (MDMs) are ideally suited to this task, yet existing samplers either decode text and image interleavedly or independently update them in parallel branches that share only previous-step history, but not the other modality’s latest decisions *within* the same step; combined with MDMs’ inability to remask, cross-modal contradictions are neither detected nor repaired. We introduce Self-Correcting Coupled Markov Jump Processes (SC-CMJP), a framework in which one modality’s transition rates are functionals of the other modality’s confidence score, as weighted by cross-modal attention. Furthermore, a remasking jump retracts commitments the moment cross-modal evidence turns against them. In conjunction with SC-CMJP, we introduce CO₂Jump (Self-CORRECTING COUPLED JUMP), a novel training-free single-pass sampler for joint multimodal generation. For training and evaluation purposes, we have created and will release three large-scale joint multimodal generation corpora: JEDIT-1M, JMAZE-200K, JNONO-200K, with matching in- and out-of-distribution benchmarks. CO₂Jump achieves best joint performance for image understanding and editing as well as visual reasoning (maze and nonogram solving). The performance of the sampler scales monotonically with the number of denoising steps, evidence that the benefits of cross-modal coupling *compound* across the trajectory. Project page: <https://coupled-jump.github.io>

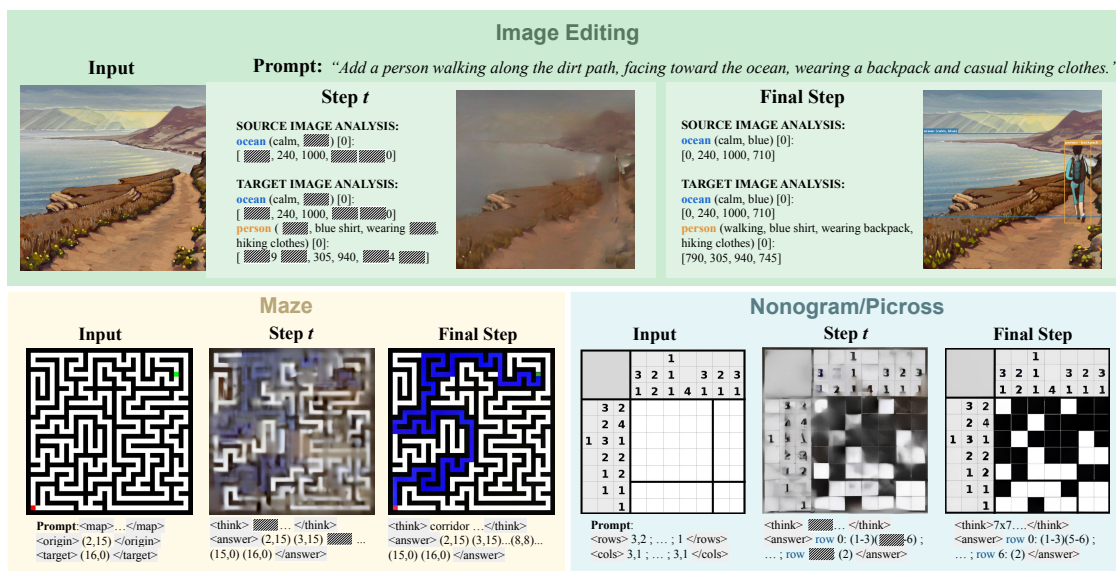


Figure 1 | CO₂Jump in action: text and image co-author the answer. Three trajectories of CO₂Jump on image editing, maze, and nonogram solving, showing the joint state at an intermediate step t and at the final step. The image-editing panel highlights the core mechanism: at step t the text branch has already begun committing a *target-image* bounding box in text for the new object *person*; by the final step the image branch has placed the hiker *exactly* inside the finalized box (we overlay the bounding boxes from generated text on edited image). The text branch *plans* where the edit should land, and the image branch *executes* that plan within the same denoising trajectory — no second forward pass, no external grounder. Maze and Nonogram show the same coupled-refinement pattern: partial-path and partial cell-fill commitments at step t converge with their text-side answers by the final step.

1. Introduction

When a teacher explains an idea at a whiteboard, language and drawing take place *together*: each utterance affects the sketch, and each new mark affects the next sentence. Understanding and generation are not separate stages but a tightly coupled loop, with each modality continuously informing and revising the other as the explanation unfolds. We aim to operationalize this loop for artificial systems, producing text and image content concurrently rather than sequentially, with one modality shaping the other as it is generated. While our method is modality-agnostic in principle, in this paper we focus on pairing text for understanding, with images for generation.

Masked Diffusion Models (MDMs) (Austin et al., 2021; Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024) are well-suited for parallel multimodal generation. Unlike autoregressive sequential pipelines (Chen et al., 2025; Deng et al., 2025; Xie et al., 2025), which first decode the entire textual reasoning trace and then condition image synthesis on it – a unidirectional flow that cannot retract early reasoning errors – MDMs predict all masked tokens simultaneously, admit a clean continuous-time formulation as Markov Jump Processes (Berghaus et al., 2024; Campbell et al., 2022), and scale naturally to multiple modalities under a unified vocabulary spanning text and image tokens (Xin et al., 2025; Yang et al., 2025). Their parallel structure makes them, in principle, a natural framework for joint text/image generation in a single decoding loop.

In practice, existing MDM samplers (Chen et al., 2026; Ouyang et al., 2026; Sahoo et al., 2024; Tian et al., 2026; Wang et al., 2025) fall short of *true* concurrent joint multimodal generation. Even samplers that nominally decode both modalities in parallel (Chen et al., 2026; Tian et al., 2026) factorize each denoising step so that the text and image updates depend only on the previous joint state and not *within* the same step; concurrency reduces to interleaving over a shared history. The resulting trajectories might drift: text might commit to descriptions the image cannot illustrate, the image might render content the text never described. Compounding this, standard masked diffusion is unable to remask (Austin et al., 2021; Campbell et al., 2022; Sahoo et al., 2024), once a token is committed it cannot be revised. Cross-modal contradictions introduced by an uncoupled parallel decoder persist for the rest of sampling.

We address both problems jointly. We introduce **Self-Correcting Coupled Markov Jump Processes (SC-CMJP)**, a general framework for concurrent joint multimodal generation in which the two modalities actively cross-weigh their commitments *within* every denoising step. One modality’s transition rates become functionals of the other modality’s unmasking confidence score, weighted by cross-modal attention extracted from the same backbone forward pass. The unmasking schedule of one modality adapts to the confidence of the other modality. Combined with a remasking jump in the spirit of ReMDM (Wang et al., 2025), this lifts decoding from one-way unmasking to a bidirectional birth-death (unmask-remask) process that can both reveal new tokens and retract earlier ones whenever cross-modal evidence turns against them.

Along with SC-CMJP, we design a single-pass training-free sampler CO₂Jump (Self-CORrecting COupled Jump) that highlights two core ideas: *Coupling* the per-modality transition rates through cross-modal attention, and *Correcting* earlier commitments via a remasking jump. CO₂Jump runs on a frozen MDM with no architectural change, no auxiliary evaluator, and a single forward pass per step. Figure 1 illustrates CO₂Jump on all three of our benchmarks; in the image-editing trajectory, the text branch commits a bounding box for the inserted object and the image branch fills it in. In the *same* step, the text *plans* where the edit should land, the image branch *executes* that plan.

To validate CO₂Jump, we instantiate it on three concurrent text-and-image tasks of increasing semantic difficulty: image editing on an extended ImgEditBench (Ye et al., 2025b) protocol with mAP-style (Lin et al., 2014) grounded-understanding metrics, and two new visual-reasoning tasks,

a maze and a nonogram (JMAZE and JNONO) where text and image are logically interlocked and jointly verifiable against algorithmic ground truth. To enable training and evaluation, we curate three corpora: JEDIT-1M, JMAZE-200K, and JNONO-200K, all of which we plan to release. Across all three tasks, CO₂Jump consistently improves both modalities, beats existing sampling methods (Sahoo et al., 2024; Tian et al., 2026; Wang et al., 2025) on concurrent joint image understanding and generation, especially joint performance metrics. In addition, CO₂Jump sampler’s performance scales monotonically with the number of denoising steps. In summary, our main contributions are:

- We introduce the first approach to model simultaneous image understanding and generation as a single, unified stochastic process. To achieve this, we propose Self-Correcting Coupled Markov Jump Processes, that integrate parallel joint multimodal generation with built-in self-correction.
- We design a novel coupled sampler CO₂Jump for joint multimodal sampling, running in a single forward pass per step.
- We curated three large-scale joint-generation corpora (JEDIT-1M, JMAZE-200K, JNONO-200K) along with matching benchmarks that probe both in-distribution and out-of-distribution performance. We have a plan to release model checkpoints, code, datasets to community.
- CO₂Jump improves the state-of-the-art on concurrent image understanding and editing, as well as visual reasoning tasks against existing sampling methods. Our sampler’s performance scales monotonically with the number of denoising steps – empirical evidence that the benefits of cross-modal coupling *compound* across the trajectory.

2. Background

2.1. Masked Diffusion Models and Remasking

Masked discrete diffusion models (Austin et al., 2021; Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024) corrupt a clean sample $\mathbf{x} \in \mathcal{V}^L$ by gradually replacing tokens with a special absorbing state $\mathbf{m}$, and learn to invert this corruption. We adopt the continuous-time formulation of Campbell et al. (2022) as our default view because it admits the cross-modal coupling and remasking extensions developed in this paper.

Forward process and CTMC equivalence. For $t \in [0, 1]$ and a monotonically decreasing noise schedule $\alpha_t \in [0, 1]$ with $\alpha_0 \approx 1$, $\alpha_1 \approx 0$, the per-position marginal of the forward process is

$$q(\mathbf{z}_t | \mathbf{x}) = \text{Cat}(\mathbf{z}_t; \alpha_t \mathbf{x} + (1 - \alpha_t) \mathbf{m}), \quad (1)$$

factorized across positions. The same dynamics admit an equivalent Continuous-Time Markov Chain (CTMC) description: states evolve by stochastic jumps between $\mathbf{x}$ and $\mathbf{m}$ at the infinitesimal rate

$$\mathbf{R}_t = -\frac{\dot{\alpha}_t}{\alpha_t}(\mathbf{I} - \mathbf{m}\mathbf{1}^\top), \quad (2)$$

under which the integrated transition probability $q(\mathbf{z}_t = \mathbf{x} | \mathbf{x}) = \alpha_t$ recovers Eq. (1) exactly. Sahoo et al. (2024) establish that the discrete-time absorbing-state formulation of MDLM and the CTMC formulation of Campbell et al. (2022) parameterize the same family of marginals, posteriors, and likelihood bounds; we use the two views interchangeably.

Training objective. A denoising network $\mathbf{x}_\theta$ is trained to predict the clean state $\mathbf{x}$ from $\mathbf{z}_t$, and the resulting NELBO collapses to a position-wise weighted cross-entropy (Sahoo et al., 2024; Shi et al.,

2024):

$$\mathcal{L}_{\text{NELBO}}^{\infty} = \mathbb{E}_{q,t} \int_0^1 \frac{\dot{\alpha}_t}{1 - \alpha_t} \sum_{\ell=1}^L \log \langle \mathbf{x}_{\theta}^{\ell}(\mathbf{z}_t, t), \mathbf{x}^{\ell} \rangle dt. \quad (3)$$

The corresponding reverse posterior of standard MDLM has a well-known inability to remask (Austin et al., 2021; Campbell et al., 2022; Sahoo et al., 2024): once a token is unmasked, no subsequent reverse step can revisit it, so any error committed at a timestep t persists for the rest of the trajectory.

Remasking via σ_t . ReMDM (Wang et al., 2025) repairs this by allowing committed tokens to revert to $\mathbf{m}$ with per-step probability $\sigma_t \in [0, 1]$, yielding the modified reverse posterior

$$q_{\sigma}(\mathbf{z}_s | \mathbf{z}_t, \mathbf{x}) = \begin{cases} \text{Cat}(\mathbf{z}_s; (1 - \sigma_t) \mathbf{x} + \sigma_t \mathbf{m}), & \mathbf{z}_t \neq \mathbf{m} \\ \text{Cat}\left(\mathbf{z}_s; \frac{\alpha_s - (1 - \sigma_t)\alpha_t}{1 - \alpha_t} \mathbf{x} + \frac{1 - \alpha_s - \sigma_t\alpha_t}{1 - \alpha_t} \mathbf{m}\right), & \mathbf{z}_t = \mathbf{m}, \end{cases} \quad (4)$$

which preserves the marginal in Eq. (1) when $\sigma_t \leq \min(1, (1 - \alpha_s)/\alpha_t)$ and recovers MDLM at $\sigma_t = 0$. The remask jump makes the forward process non-Markovian, but the reverse stays Markovian and tractable (Wang et al., 2025). Existing σ_t schedules (Wang et al., 2025) are *modality-agnostic* – they score remasks from intra-modal likelihoods alone, leaving cross-modal contradictions undetected in joint generation.

2.2. Markov Jump Processes and Coupled Multimodal Generation

A Markov Jump Process (MJP) (Berghaus et al., 2024; Campbell et al., 2022) is a continuous-time Markov process on a discrete state space specified by a rate matrix $R_t(z, z')$; the CTMC view (Campbell et al., 2022; Ouyang et al., 2026; Sahoo et al., 2024) of MDM is exactly such an MJP over $\mathcal{V}^L$ with rate matrix Eq. (2). Exact reverse simulation via Gillespie’s algorithm (Gillespie, 1976, 1977) updates one position per jump and is prohibitive at modern sequence lengths, so practical samplers use τ -leaping (Gillespie, 2001) – a parallel approximation that updates all positions simultaneously, of which the standard MDM reverse step is the first-order discretization (Campbell et al., 2022).

Coupled MJPs for Multimodal Generation. Coupled jump processes are classical in chemistry (Gillespie, 1977) and Glauber dynamics (Glauber, 1963), but their coupling structure is hand-specified and they target physical simulation; standard MJP-based diffusion likewise treats per-position rates as conditionally independent given $\mathbf{x}_{\theta}(\mathbf{z}_t)$ even in multimodal settings. We instead define a *Coupled Markov Jump Process* (CMJP) over $\mathbf{z}_t = (\mathbf{z}_t^{\text{text}}, \mathbf{z}_t^{\text{image}})$ with modality-specific rate matrices $\mathbf{R}_t^a$ whose intensities depend on the hidden representations and instantaneous confidence of the complementary modality through learned cross-modal attention. Combined with a ReMDM-style σ_t jump (Eq. (4)), birth (unmask) and death (remask) rates of one modality are informed by the current commitments of the other, enabling localized cross-modal self-correction at sampling time.

3. Related Work

Discrete Diffusion. Absorbing-state discrete diffusion was introduced in D3PM (Austin et al., 2021) and refined into score-entropy (Lou et al., 2024), simplified-ELBO (Sahoo et al., 2024; Shi et al., 2024), and any-order autoregressive (Ou et al., 2025) formulations. Scaling to LLMs has been driven by LLaDA (Nie et al., 2025), Dream (Ye et al., 2025a), and SDAR (Cheng et al., 2025). Decoding improvements include block-wise generation (Arriola et al., 2025), Top-K confidence selection (Kim et al., 2025b; Nie et al., 2025), and length-adaptive scheduling (Ou et al., 2025). Multimodal

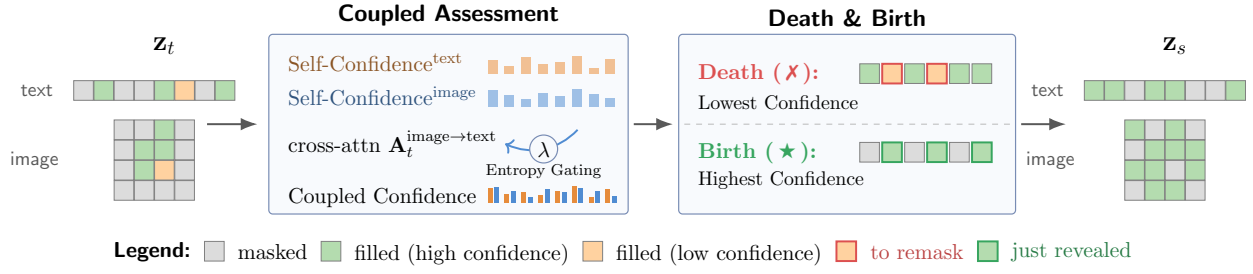


Figure 2 | **CO₂Jump sampler.** A single denoising step from $\mathbf{z}_t$ to $\mathbf{z}_s$. From one forward pass, the model produces per-token Self-Confidence for both modalities; cross-modal attention $\mathbf{A}_t^{\text{image} \rightarrow \text{text}}$ propagates text confidence to image positions, and an entropy-based gate λ mixes self and cross signals into Coupled Confidence. The *Death* jump remarasks the lowest-confidence committed tokens, and the *Birth* jump reveals the highest-confidence masked tokens under the noise schedule.

extensions tokenize images with VQ-VAE (van den Oord et al., 2017) and operate over a unified vocabulary spanning text and image tokens, e.g. Lumina-DiMOO (Xin et al., 2025) and MMaDA (Yang et al., 2025).

Self-Correction via Remasking. Three families of method address the failure-to-remask problem. *Predictor-corrector samplers* (Campbell et al., 2022, 2024; Gat et al., 2024; Lezama et al., 2023) reduce τ -leaping error via corrector steps without an explicit remask jump. *Training-based remasking* modifies the model: GIDD (von Rutte et al., 2025) generalizes forward and reverse processes; Zhao et al. (2026) train a separate hollow-transformer evaluator; Huang et al. (2025); Kim et al. (2025a) fine-tune the pretrained MDM to estimate per-token quality. *Training-free remasking* keeps the backbone frozen: ReMDM (Wang et al., 2025) adds a heuristic σ_t schedule, Peng et al. (2025) explore path-following corrections, and Ouyang et al. (2026) use cumulative-confidence signals. CO₂Jump sits within the training-free family, but is the first to couple the remasking signal across modalities and treat cross-modal contradiction as the self-correction trigger.

Concurrent Multimodal Samplers. UD-VLA (Chen et al., 2026) factorizes the joint into independent per-modality terms (an instance of MDM (Sahoo et al., 2024) sampling); MMaDA-Parallel (Tian et al., 2026) interleaves text and image updates across steps but samples them *independently within each step* – both modality updates condition only on the previous joint state, with no cross-modal feedback inside the step itself, so any coupling is realized only through shared history rather than instantaneous negotiation. Our experiments compare CO₂Jump against three representative samplers covering these regimes: MDM (Sahoo et al., 2024), ReMDM (Wang et al., 2025), and MMaDA-Parallel (Tian et al., 2026). Coupled jump processes in chemistry and statistical physics (Gillespie, 1977; Glauber, 1963) use hand-specified couplings for physical simulation, so neither methods transfer to our setting.

4. Self-Correcting Coupled Markov Jump Processes

Standard multimodal diffusion samplers often suffer from *modality-drift*, where the generated text and image diverge because their unmasking schedules are independent and irreversible. We address this by reformulating joint generation as a unified system where modalities actively negotiate their commitment through a shared birth-death jump process whose intensities are coupled across modalities. Figure 2 illustrates a single denoising step of the resulting sampler.

4.1. Continuous-Time Likelihood Bounds for Joint Generation

Following MDM derivation (Sahoo et al., 2024), for $t \in [0, 1]$ and a monotonically decreasing noise schedule $\alpha_t \in [0, 1]$, the per-position marginal of the forward process is

$$q(\mathbf{z}_t | \mathbf{x}) = \text{Cat}(\mathbf{z}_t; \alpha_t \mathbf{x} + (1 - \alpha_t) \mathbf{m}), \quad (5)$$

factorized across positions. We treat text and image as a single sequence over the union vocabulary $\mathcal{V} = \mathcal{V}_{\text{text}} \cup \mathcal{V}_{\text{image}}$ and share a single absorbing token $\mathbf{m}$ across both modalities, so that the per-position marginal in Eq. (5) applies uniformly to text and image positions; this is what allows the joint NELBO in Eq. (6) below to factor cleanly without a cross-modality term. A joint clean sample is $\mathbf{x} = (\mathbf{x}^{\text{text}}, \mathbf{x}^{\text{image}}) \in \mathcal{V}^L$ with total vocabulary size $L = L^{\text{text}} + L^{\text{image}}$, and is corrupted by $q(\mathbf{z}_t | \mathbf{x})$, applied position-wise across both modalities under a shared schedule α_t .

Because the per-position marginal factorizes and both modalities use the same α_t , the joint NELBO inherits the similar form in MDM (Sahoo et al., 2024) with no cross-modality term in the objective:

$$\mathcal{L}_{\text{NELBO}}^\infty = \mathbb{E}_{q,t} \int_0^1 \frac{\dot{\alpha}_t}{1 - \alpha_t} \sum_{\ell=1}^L \log \langle \mathbf{x}_\theta^\ell(\mathbf{z}_t, t), \mathbf{x}^\ell \rangle dt. \quad (6)$$

A crucial property is implicit in the conditioning: the network output $\mathbf{x}_\theta^\ell(\mathbf{z}_t, t)$ at any position ℓ has access to the *entire* joint state $\mathbf{z}_t = (\mathbf{z}_t^{\text{text}}, \mathbf{z}_t^{\text{image}})$, including the complementary modality. Training under Eq. (6) therefore implicitly forces $\mathbf{x}_\theta$ to learn cross-modal denoising signals – to reconstruct a masked image patch from surrounding pixels *and* concurrent textual clues, and conversely to reconstruct a masked text token from surrounding context *and* the partially decoded image. The NELBO never references this dependency explicitly, yet at convergence it is encoded in the network’s hidden representations and attention patterns. Our coupled sampler CO₂Jump extracts these latent cross-modal signals at inference time via a single forward pass through $\mathbf{x}_\theta$, with no architectural changes to the trained model and no auxiliary cross-modal evaluator.

4.2. Asymmetric Sequential Sampling via Chain-Rule Decomposition

Let c denote the task conditioning context (e.g., the editing prompt together with the source image for image editing, or the puzzle specification for visual reasoning). A naive sampler for the joint reverse posterior $p_\theta(\mathbf{z}_s^{\text{text}}, \mathbf{z}_s^{\text{image}} | \mathbf{z}_t, c)$ updates both modalities independently from the same $\mathbf{z}_t$, ignoring that the latest text decision $\mathbf{z}_s^{\text{text}}$ provides immediate context for inferring the corresponding image state $\mathbf{z}_s^{\text{image}}$. We instead exploit the chain-rule factorization

$$p(\mathbf{z}_s^{\text{text}}, \mathbf{z}_s^{\text{image}} | \mathbf{z}_t, c) = \underbrace{p(\mathbf{z}_s^{\text{text}} | \mathbf{z}_t, c)}_{\text{(I) text update}} \cdot \underbrace{p(\mathbf{z}_s^{\text{image}} | \mathbf{z}_t, \mathbf{z}_s^{\text{text}}, c)}_{\text{(II) image update conditioned on text at step } s}, \quad (7)$$

which makes explicit that the text update needs no information beyond the current $\mathbf{z}_t$, while the image update would ideally condition on the latest sampled $\mathbf{z}_s^{\text{text}}$.

Evaluating Term (II) exactly demands a second forward pass through $\mathbf{x}_\theta$ at every diffusion step – prohibitive at inference time. We avoid this cost by approximating $\mathbf{z}_s^{\text{text}}$ with quantities already produced by the single forward pass at $\mathbf{z}_t$: the image side reads the model’s self-belief over $\mathbf{z}_t^{\text{text}}$, propagates it through cross-modal attention, and uses the result as a low-cost surrogate for the unavailable $\mathbf{z}_s^{\text{text}}$. This yields an *asymmetric scoring rule*: text positions are scored by their pure self-confidence, while image positions are scored by an entropy-gated mixture of self-confidence and a cross-modal signal – the *Coupled Confidence*. Both scores then drive the death-and-birth dynamics.

4.3. Self-Confidence and Coupled Confidence

Self-Belief via Gumbel-Max. At each timestep, the network produces beliefs $p_\theta(\mathbf{x}^\ell \mid \mathbf{z}_t, c)$ over each position’s clean token. To escape local minima, we apply Gumbel-max sampling, $\hat{\mathbf{x}}^\ell = \arg \max_{v \in \mathcal{V}} (\log p_\theta(v \mid \mathbf{z}_t, c) + \gamma_v)$ with i.i.d. Gumbel noise γ_v . The *Self-Confidence* of position ℓ is the model’s probability for its sampled choice: $\text{SelfConf}_\ell = p_\theta(\hat{\mathbf{x}}^\ell \mid \mathbf{z}_t, c)$.

Text Scoring. Because Term (I) of Eq. (7) has no cross-modal dependency, the per-position text score is the self-confidence directly:

$$\text{Score}_{\text{text}, \ell} = \text{SelfConf}_{\text{text}, \ell}. \quad (8)$$

Cross-Modal Negotiation. For Term (II), we approximate the conditioning on $\mathbf{z}_s^{\text{text}}$ using $\mathbf{z}_t^{\text{text}}$ via cross-attention. Let $\mathbf{H}_t^{\text{image}} \in \mathbb{R}^{L^{\text{image}} \times D}$ and $\mathbf{H}_t^{\text{text}} \in \mathbb{R}^{L^{\text{text}} \times D}$ be the hidden representations of the two modalities, extracted post-hoc from the model’s final layer during the same forward pass that produced SelfConf – no architectural modification or auxiliary attention head is introduced. We compute the image→text cross-attention

$$\mathbf{A}_t^{\text{image} \rightarrow \text{text}} = \text{Softmax} \left(\frac{\mathbf{H}_t^{\text{image}} (\mathbf{H}_t^{\text{text}})^\top}{\sqrt{D}} + \mathbf{B}_t \right), \quad (9)$$

where $\mathbf{B}_t$ is a mask-aware bias that downweights attention toward currently masked positions. The cross-modal negotiation is implemented as the cross signal for image position ℓ , which is the attention-weighted text self-confidence:

$$\text{CrossSignal}_\ell = \sum_{j=1}^{L^{\text{text}}} \mathbf{A}_{t, \ell j}^{\text{image} \rightarrow \text{text}} \cdot \text{SelfConf}_{\text{text}, j}. \quad (10)$$

Dynamic Trust through Entropy-Based Gating. The weight given to the cross signal versus the self signal on the image side is a *single scalar gate* $\lambda \in [0, 1]$, computed once per denoising step and applied uniformly to every image position. We derive λ from per-modality predictive uncertainty: letting $\bar{\mathcal{H}}_a = \frac{1}{L^a} \sum_{\ell=1}^{L^a} [-\sum_v p_\theta(v \mid \mathbf{z}_t, c)_{a, \ell} \log p_\theta(v \mid \mathbf{z}_t, c)_{a, \ell}]$ denote the mean token-level Shannon entropy in modality a , averaged over *all* positions of that modality,

$$\lambda = \frac{\bar{\mathcal{H}}_{\text{image}}}{\bar{\mathcal{H}}_{\text{image}} + \bar{\mathcal{H}}_{\text{text}} + \epsilon}. \quad (11)$$

When the image is locally chaotic (high $\bar{\mathcal{H}}_{\text{image}}$), $\lambda \rightarrow 1$ and the image modality defers to the text for evidentiary support; when the text is the uncertain side, $\lambda \rightarrow 0$ and the image relies on its own self-belief. The single-gate formulation follows directly from Eq. (7) – only the image term carries a cross-modal dependency to be resolved.

Shared Percentile Rank Space. Because the text and image vocabularies differ in size ($|\mathcal{V}_{\text{text}}| \gg |\mathcal{V}_{\text{image}}|$), absolute confidence values are not directly comparable across modalities. We therefore project each scalar score into its empirical percentile rank within its own modality, ensuring the image-side mixture below is well-conditioned regardless of vocabulary scale.

Coupled Confidence for Image Scoring. Combining the rank-normalized self and cross signals:

$$\text{Score}_{\text{image},\ell} = \text{CoupledConf}_{\ell} = (1 - \lambda) \cdot \text{Rank}(\text{SelfConf}_{\text{image},\ell}) + \lambda \cdot \text{Rank}(\text{CrossSignal}_{\ell}). \quad (12)$$

4.4. The CO₂Jump Sampler

We interpret the joint reverse generation as a modality-specific *birth-death* jump process. Standard masked diffusion only “births” tokens (unmask) and never updates them. CO₂Jump additionally allows tokens of either modality to “die” (remask) when their scores fall below the schedule-driven death rate, allowing both intra-modal errors and cross-modal contradictions to be retracted.

Death Step (Remasking). At each discretization step ($t \rightarrow s$) we set the death rate σ_t to satisfy the valid-posterior constraint $\sigma_t \leq \min(\eta, (1 - \alpha_s)/\alpha_t)$. For each modality a , the idealized unmasked count is $U_t^a = \alpha_t L^a$ and the idealized masked count is $M_t^a = (1 - \alpha_t)L^a$ (we use $\lfloor \cdot \rfloor$ to obtain integer token counts). The death quota is $N_{\text{remask}}^a = \lfloor U_t^a \cdot \sigma_t \rfloor$, and the N_{remask}^a unmasked tokens with the lowest Score_a are reverted to absorbing token $\mathbf{m}$. Per the asymmetric rule (Eq. (7)), remasking uses SelfConf for text (Eq. (8)) and CoupledConf for image (Eq. (12)).

Birth Step (Unmasking). To keep the global signal-to-noise progression matched to the schedule, the birth quota $N_{\text{unmask}}^a = \lfloor \frac{\alpha_s - \alpha_t}{1 - \alpha_t} \cdot M_t^a \rfloor + \Delta N^a$ combines the base unmask count with an extra-unmask compensation $\Delta N^a = \lfloor \sigma_t \cdot \frac{\alpha_t}{1 - \alpha_t} \cdot M_t^a \rfloor$ derived from the ReMDM (Wang et al., 2025) posterior. Among the post-remask masked tokens of modality a , we reveal the N_{unmask}^a with the highest Score_a , again with the asymmetric definition. Algorithm 1 gives the full pseudocode.

Algorithm 1 CO₂Jump Sampler (Asymmetric Scoring)

```

1: Input: Pretrained  $\mathbf{x}_\theta$ , schedule  $\alpha_t$ , remasking ratio  $\eta$ , steps  $T$ , modality lengths  $L^{\text{text}}, L^{\text{image}}$ ,
   conditioning  $c$ .
2: Initialize  $\mathbf{z}_1 \leftarrow \{\mathbf{m}\}^{L^{\text{text}} + L^{\text{image}}}$ .
3: for  $i = T$  down to 1 do
4:    $t \leftarrow i/T, s \leftarrow (i - 1)/T$ 
5:    $\mathbf{z}_s \leftarrow \mathbf{z}_t$  // initialize next-step state; death/birth jumps below write into  $\mathbf{z}_s$ 
6:   // Single forward pass
7:   Run  $\mathbf{x}_\theta(\mathbf{z}_t, c)$ ; extract  $\mathbf{H}_t^{\text{text}}, \mathbf{H}_t^{\text{image}}$  and mean entropies  $\bar{\mathcal{H}}_{\text{text}}, \bar{\mathcal{H}}_{\text{image}}$  over all positions
8:   Sample  $\hat{\mathbf{x}}^\ell$  via Gumbel noise; record  $\text{SelfConf}_\ell$  for both modalities
9:   // Asymmetric scoring (Eq. (7))
10:  Text  $\text{Score}_{\text{text},\ell} \leftarrow \text{SelfConf}_{\text{text},\ell}$  // Term (I): no cross-modal dependence
11:  Image compute  $\mathbf{A}_t^{\text{image} \rightarrow \text{text}}, \text{CrossSignal}_\ell$  (Eqs. 9 and 10), scalar gate  $\lambda$  (Eq. 11)
12:   $\text{Score}_{\text{image},\ell} \leftarrow \text{CoupledConf}_\ell$  via Eq. (12) // Term (II): approx.  $\mathbf{z}_s^{\text{text}}$  via attention
13:  // Death & Birth Jumps
14:   $\sigma_t \leftarrow \min(\eta, (1 - \alpha_s)/\alpha_t)$  // valid-posterior constraint
15:  for  $a \in \{\text{text}, \text{image}\}$  do
16:     $U_t^a \leftarrow \lfloor \alpha_t L^a \rfloor, M_t^a \leftarrow \lfloor (1 - \alpha_t)L^a \rfloor$  // idealized counts, floored at runtime
17:     $N_{\text{remask}}^a \leftarrow \lfloor U_t^a \sigma_t \rfloor, N_{\text{unmask}}^a \leftarrow \lfloor \frac{\alpha_s - \alpha_t}{1 - \alpha_t} M_t^a \rfloor + \lfloor \sigma_t \frac{\alpha_t}{1 - \alpha_t} M_t^a \rfloor$ 
18:    Death: in  $\mathbf{z}_s$ , remask the  $N_{\text{remask}}^a$  tokens of modality  $a$  with lowest  $\text{Score}_a$  to  $\mathbf{m}$ 
19:    Birth: in  $\mathbf{z}_s$ , reveal the  $N_{\text{unmask}}^a$  masked tokens of modality  $a$  with highest  $\text{Score}_a$ 
20:  end for
21: end for
22: Output: Unified multimodal sample  $\mathbf{z}_0$ 

```

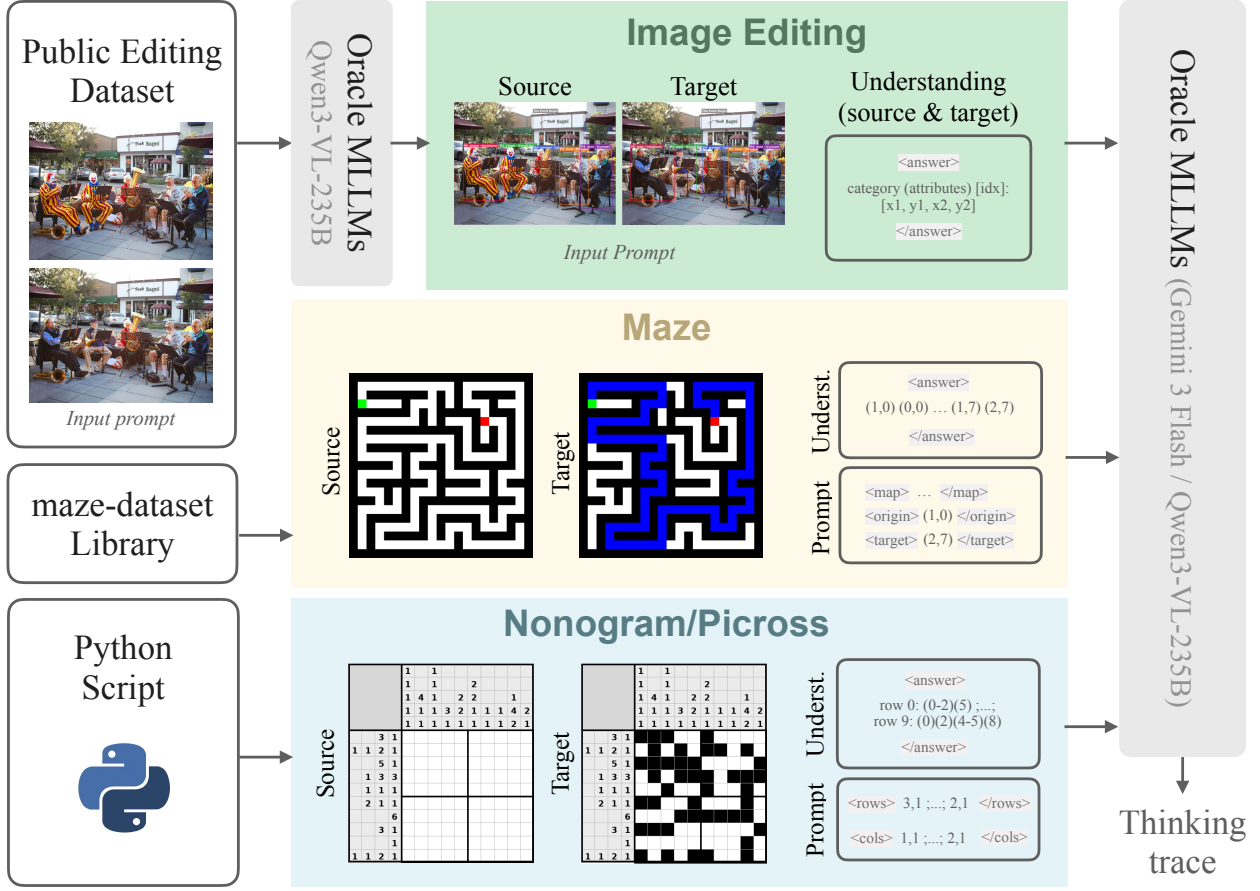


Figure 3 | **Dataset curation pipeline.** For JEDIT-1M, raw editing pairs are augmented by an oracle Qwen3-VL-235B that produces both the per-image scene-graph understanding and the thinking trace. For JMAZE-200K and JNONO-200K, source/target images and the structured understanding are produced algorithmically.

5. Datasets and Benchmarks for Joint Multimodal Generation

To train and evaluate CO₂Jump across both photographic and logical multimodal tasks, we curate three joint-generation datasets: JEDIT-1M for image editing, JMAZE-200K for maze solving, and JNONO-200K for nonogram solving, each paired with a held-out benchmark. All three corpora share the same record schema (prompt, source image, target image, structured understanding, thinking trace); Figure 3 shows the curation pipeline.

5.1. JEDIT-1M: Joint Image Editing and Understanding

Training corpus. JEDIT-1M combines a 724k-pair subset of ImgEdit (Ye et al., 2025b) with a 368k-pair subset of OmniEdit (Wei et al., 2025), yielding 1M (prompt, source, target) tuples. Because raw editing data lacks the joint understanding and thinking supervision that joint multimodal generation requires, we augment each pair with two fields synthesized by Qwen3-VL-235B (Bai et al., 2025): (i) a pixel-aligned source/target scene-graph extraction with per-category indices preserved across panels, and (ii) a logic-based thinking trace conditioned on both images, the prompt, and the scene graph from (i). Figure 4 provides a concrete example of a training sample of the JEDIT-1M dataset.

<think>

The transition from the source to the target image executes a targeted semantic swap of subjects while maintaining environmental and compositional integrity.

Static Objects: The background architecture (The Posh Bagel building [487, 85, 804, 401]), parked vehicles (silver sedan [0] and dark gray car [1]), and burgundy umbrellas [0, 1] remain spatially invariant. Crucially, the three musicians on the right (Persons [2, 3, 4]) and all five black metal music stands [0-4] remain fixed in their respective coordinates, anchoring the scene's layout.

Dynamic/Deleted & Added Objects:

- **Person [0]:** The clown in a red and yellow striped costume playing a trombone at [56, 363, 315, 804] is removed. He is replaced by an older man with glasses, wearing a blue shirt and black sweater vest [48, 371, 321, 803], who continues to play the trombone.
- **Person [1]:** The second clown in striped attire at [281, 363, 458, 775] is removed. He is replaced by an older man with a white beard and blue cap wearing a tan jacket [290, 372, 457, 779], playing the French horn.

Quantifying the Change: The transformation involves a precise substitution of the leftmost figures. The bounding box for Person [0] shifts marginally (-8px left, +8px top) to accommodate the new musician's posture, while Person [1] shifts (+9px left, +9px top). The core change is attribute-based: costumes and face paint are replaced by "casual clothing" (sweater vests, jackets, and caps).

Critical Alignment Check: The edit strictly adheres to the prompt. The two clowns are successfully swapped for "older musicians" in "casual clothing." Both new subjects are depicted playing "brass instruments" (trombone and French horn), matching the musical context of the scene. There are no hallucinations; the replacement is seamless and logical, transforming the ensemble into a stylistically unified group of seniors.

</think>

<answer>

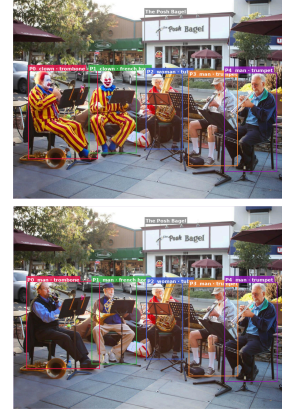
SOURCE IMAGE ANALYSIS:

building (white facade, "The Posh Bagel" sign)	[0]: [487, 85, 804, 401]
person (clown, red and yellow striped costume, face paint, playing trombone)	[0]: [56, 363, 315, 804]
person (clown, red and yellow striped costume, mask, playing French horn)	[1]: [281, 363, 458, 775]
person (older woman, white hair, red jacket, playing tuba)	[2]: [493, 378, 629, 755]
person (older man, tan hat, grey jacket, khaki shorts, playing trumpet)	[3]: [651, 395, 834, 839]
person (older man, glasses, blue shirt, dark navy jacket, playing trumpet)	[4]: [783, 376, 980, 866]

TARGET IMAGE ANALYSIS:

building (white facade, "The Posh Bagel" sign)	[0]: [487, 85, 804, 401]
person (older man, glasses, blue shirt, black sweater vest, playing trombone)	[0]: [48, 371, 321, 803]
person (older man, white beard, blue cap, tan jacket, playing French horn)	[1]: [290, 372, 457, 779]
person (older woman, white hair, red jacket, playing tuba)	[2]: [493, 378, 629, 755]
person (older man, tan hat, grey jacket, khaki shorts, playing trumpet)	[3]: [651, 395, 834, 839]
person (older man, glasses, blue shirt, dark navy jacket, playing trumpet)	[4]: [783, 376, 980, 866]

</answer>



"Swap the two clowns in red and yellow striped costumes with two older musicians wearing casual clothing and playing brass instruments."

Figure 4 | **JEDIT-1M training sample.** A natural-language edit prompt (bottom) paired with the source and target images (right), the oracle Qwen3-VL-235B thinking trace, and the per-panel source/target scene-graph understanding. The bounding boxes and labels from MLLM's grounding solution are overlaid on the corresponding images.

Benchmark and metrics. We extend ImgEditBench (Ye et al., 2025b) with two metric families that score joint understanding + generation behavior. **Text quality** is generative perplexity under a frozen GPT-2 Large (Radford et al., 2019) plus token-distribution entropy as a mode-collapse safeguard, following MDLM (Sahoo et al., 2024) and ReMDM (Wang et al., 2025). **Image-edit fidelity** is the standard ImgEditBench oracle score from Gemini 3 Flash (Google, 2026). **Image-grounded understanding** is COCO-style mAP@0.5:0.95 (Lin et al., 2014) against a *pseudo* scene graph constructed per-sample by Gemini on the model's own (source, generated-target) pair: pseudo grounding is necessary because each model produces a different target image. We report mAP separately for source and target sections plus their average.

5.2. JMAZE-200K and JNONO-200K: Visual Reasoning

To stress-test joint generation on tasks where text and image are *logically interlocked*, we add two synthetic visual-reasoning corpora. In both, the input fully specifies a unique solution and the two modalities are independently verifiable against algorithmic ground truth.

Corpora. We use the maze-dataset (Ivanitskiy et al., 2023) library to generate JMAZE-200K comprising 200k DFS-perfect mazes with grid sizes uniformly sampled from $\{6, \dots, 20\}$; the input shows walls plus a green start and red end, the output overlays the solution path in blue, and the text answer is the path as an (r, c) sequence. JNONO-200K is 200k nonogram puzzles with grid sizes $\{5, \dots, 25\}$, generated by a mixture of pattern synthesizers biased toward multi-run row/column clues; the input renders the empty $N \times N$ grid with marginal clue numbers, and the output fills cells black to satisfy every clue. For both corpora, the thinking trace is synthesized by an oracle Gemini 3 Flash conditioned on the source/target images and the algorithmic ground-truth answer.

Parallel-form supervision. Both the structured understanding and the thinking trace are written in a form native to parallel decoding rather than autoregressive ordering: maze solutions use global (r, c) coordinates instead of relative moves (*Left/Right/Up/Down*), and nonogram supervision emphasizes bidirectional row/column constraint propagation instead of row-by-row solving. A relative direction is only well-defined once the previous coordinate is committed, while a global coordinate keeps its meaning regardless of which other tokens have been unmasked, aligning the supervision with the parallel denoising the sampler actually performs.

Benchmarks and metrics. For each task we hold out a 500-sample test set whose grid sizes *exceed* the training range on both ends: JMAZE-TEST500 spans $\{3, \dots, 22\}$ (in-dist $\{6, \dots, 20\}$, OOD $\{3, 4, 5, 21, 22\}$); JNONO-TEST500 spans $\{3, \dots, 27\}$ (in-dist $\{5, \dots, 25\}$, OOD $\{3, 4, 26, 27\}$). Our headline metric is **joint accuracy**, a sample counts correct only when both the text answer (judged by a tolerant Gemini extractor against the algorithmic ground truth) and the generated image (compared side-by-side with the ground-truth image, by Gemini 3 Flash) are correct. **Text accuracy** and **image accuracy** are reported separately as diagnostics; on in-distribution and OOD splits.

6. Experiments

We evaluate CO₂Jump against three sampler families across the three tasks of Section 5. To ensure a strictly controlled comparison: we initialize from Lumina-DiMOO (Xin et al., 2025), fine-tune on the corresponding corpus, and swap only the inference-time sampler. For each task, we fine-tune the model on 64× H100 80 GB GPUs with a total batch size of 512 and learning rate 2×10^{-5} . Baselines includes: **MDM** (Sahoo et al., 2024), independent modalities with no remasking; **ReMDM** (Wang et al., 2025), independent modalities with remasking; and **MMaDA-Parallel** (Tian et al., 2026), modalities interleave and are independent with no remasking. CO₂Jump is the only entry that generates joint modalities concurrently with cross-modal coupling and self-correction. All samplers use a cosine α_t schedule; for the remasking variants (ReMDM and CO₂Jump), we use a remasking schedule with $\sigma_t = 0.01$ on $t \in [0.25, 0.75]$ and 0 elsewhere.

6.1. Joint Image Editing and Understanding

Table 1 reports results on extended ImgEditBench. Our headline metric is per-section mAP@0.5:0.95 from the pseudo-grounding pipeline; target-mAP is the cleanest test of cross-modal coupling because

Table 1 | **Image-editing results on the extended ImgEditBench protocol.** Bottom block: apple-to-apple sampler comparison; above the rule: pre-trained Lumina-DiMOO and Qwen3-VL-8B references. **Bold/underline** = best/second best across the four fine-tuned samplers.

Method	# turns ↓	Text Generation		Image Generation	Image Understanding (mAP@0.5:0.95 ↑)		
		GPT-2L PPL ↓	Entropy ↑	ImgEditBench ↑	Source	Target	Overall
Qwen3-VL-8B	2 (I2I → I2T)	11.9128	4.7523	–	0.391	0.330	0.360
Lumina-DiMOO	3 (I2I + 2×I2T)	3.1647	3.0513	1.68	0.019	0.019	0.019
MDM	1	8.5780	<u>4.5160</u>	1.78	0.386	0.322	0.354
ReMDM	1	8.4863	4.4929	1.73	0.388	0.318	0.353
MMaDA-Parallel	1	8.8363	4.5353	1.44	0.366	0.304	0.335
CO ₂ Jump (Ours)	1	<u>8.5649</u>	4.5005	1.93	0.392	0.346	0.369

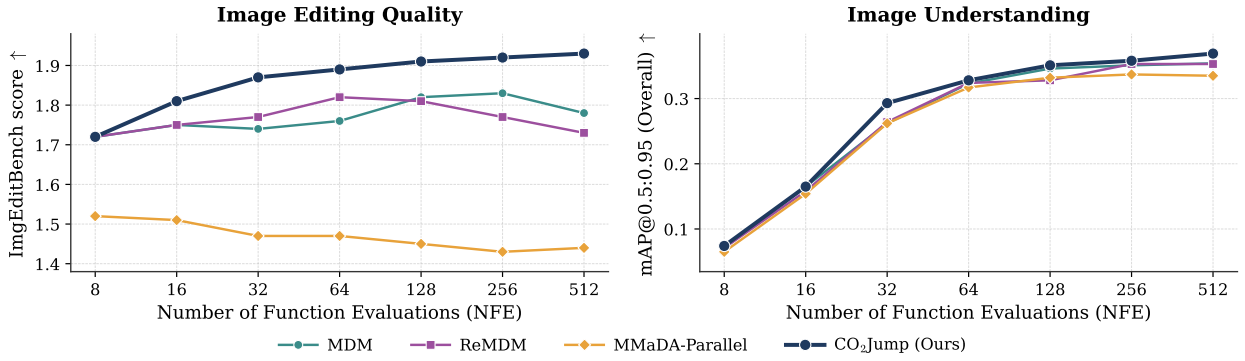


Figure 5 | **Scaling sampling steps.** ImgEditBench (left) and overall mAP (right) vs. NFE. CO₂Jump is the only sampler with monotonically rising curves; the gap to baselines widens with NFE.

it requires the image branch to place each edited object inside the bounding boxes that the text branch declares *and* the text branch to describe what the image branch actually produces.

Coupled generation beats a strong sequential grounder. Qwen3-VL-8B is a stringent reference: same scale as our backbone, and the supervision used to fine-tune all four samplers comes from the Qwen3-VL family (Bai et al., 2025). Because Qwen3-VL-8B itself cannot generate images, we evaluate its grounding on the same (source, generated-target) image pairs produced by our CO₂Jump sampler, effectively giving it the generated target image as input for free. Yet CO₂Jump produces grounded text *concurrently* with the image and still exceeds Qwen3-VL-8B on the same images, on both target mAP (0.346 vs. 0.330) and overall mAP (0.369 vs. 0.360).

Better understanding leads to better generation. Pre-trained Lumina-DiMOO has near-zero understanding mAP (0.019); fine-tuning unlocks it across all samplers, and the sampler with the strongest understanding (CO₂Jump, 0.369 overall) also achieves the best ImgEditBench score (1.93), while the weakest (MMaDA-Parallel, 0.335) is worst on ImgEditBench (1.44). This is the empirical signature of the chain-rule decomposition (Section 4.2): once text decisions inform image decisions within a single step, sharper text grounding tightens the image-side score distribution.

Scaling sampling steps. Sweeping NFE from 8 to 512 (Figure 5), CO₂Jump is the only sampler that improves *monotonically* on both metrics: ImgEditBench rises from 1.72 to 1.93 and overall mAP from 0.074 to 0.369. Single-modality baselines plateau and regress at high NFE, and MMaDA-Parallel

Table 2 | **Joint accuracy on JMAZE-TEST500 and JNono-TEST500.** A sample is correct only when text and image both match the algorithmic ground truth. **Bold/underline** = best/second best.

Method	Maze Solving (JMAZE-TEST500)			Nonogram Solving (JNono-TEST500)		
	In-Distribution 6×6–20×20	OOD {3, 4, 5, 21, 22} ²	Total 3×3–22×22	In-Distribution 5×5–25×25	OOD {3, 4, 26, 27} ²	Total 3×3–27×27
MDM	<u>0.461</u>	<u>0.312</u>	<u>0.424</u>	0.110	0.050	0.100
ReMDM	0.229	0.032	0.180	0.002	0.000	0.002
MMaDA-Parallel	0.419	0.304	0.390	<u>0.143</u>	<u>0.113</u>	<u>0.138</u>
CO ₂ Jump (Ours)	0.469	0.320	0.432	0.167	0.175	0.168

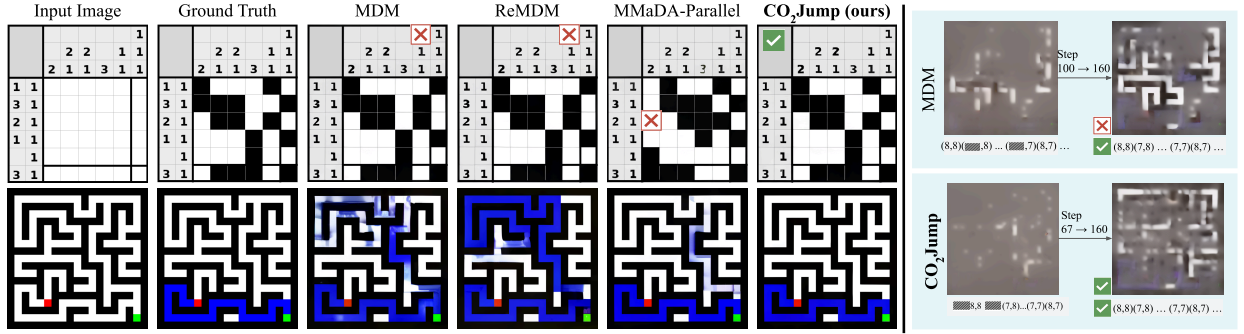


Figure 6 | **Qualitative comparison.** Top left: Nonogram (red X marks clue violations); Bottom left: Maze; Right: per-step MDM vs. CO₂Jump trajectory. Only CO₂Jump satisfies all clues and paths, and its joint trajectory stays consistent across both modalities.

degrades from 1.52 to 1.44, its uncoupled schedule cannot productively use additional steps. The gap also widens: at 8 NFE the four methods sit within 0.009 mAP, but at 512 NFE CO₂Jump is +0.015/+0.016/+0.034 ahead of MDM / ReMDM / MMaDA-Parallel. Coupling compounds across steps rather than saturating.

6.2. Visual Reasoning: Maze and Nonogram Solving

Table 2 reports joint accuracy on visual reasoning benchmarks on in-distribution and OOD grid sizes.

CO₂Jump outperforms other samplers in all six columns. Improvements over the best baseline range from +0.008 (Maze, In-Dist) to +0.062 (Nonogram, OOD). No baseline is the runner-up everywhere — MDM is second on Maze but fourth on Nonogram, and MMaDA-Parallel is the reverse — so consistent joint correctness across both task structures is itself evidence that the coupling mechanism is generic rather than benchmark-specific.

Out-of-distribution generalization. CO₂Jump also holds up best on grid sizes outside the training range. On Maze it leads on the OOD split (0.320 vs. 0.312 for the runner-up), and on Nonogram it widens its lead substantially (0.175 vs. 0.113). MDM drops 55% from In-Dist to OOD on Nonogram and MMaDA-Parallel 21%, while CO₂Jump stays roughly flat. It shows that cross-modal coupling transfers solution-style information across grid sizes more robustly than uncoupled samplers.

Table 3 | **Ablation on CO₂ Jump**. Each row removes one mechanism: Shared Percentile Rank, Entropy-Based Gating (λ_{image}), or Self-Correction. **Bold/underline** = best/second best.

Method	Text Generation		Image Generation	Image Understanding (mAP@0.5:0.95 $\uparrow$)		
	GPT-2L PPL $\downarrow$	Entropy $\uparrow$	ImgEditBench $\uparrow$	Source	Target	Overall
CO₂ Jump (full)	8.5649	<u>4.5005</u>	1.93	0.392	0.346	0.369
– Shared Percentile Rank	8.7117	4.4957	1.87	<u>0.374</u>	0.315	0.344
– Entropy-Based Gating	<u>8.7078</u>	4.4984	1.91	0.392	<u>0.316</u>	<u>0.354</u>
– Self-Correction	8.8299	4.5180	<u>1.92</u>	0.363	0.310	0.337

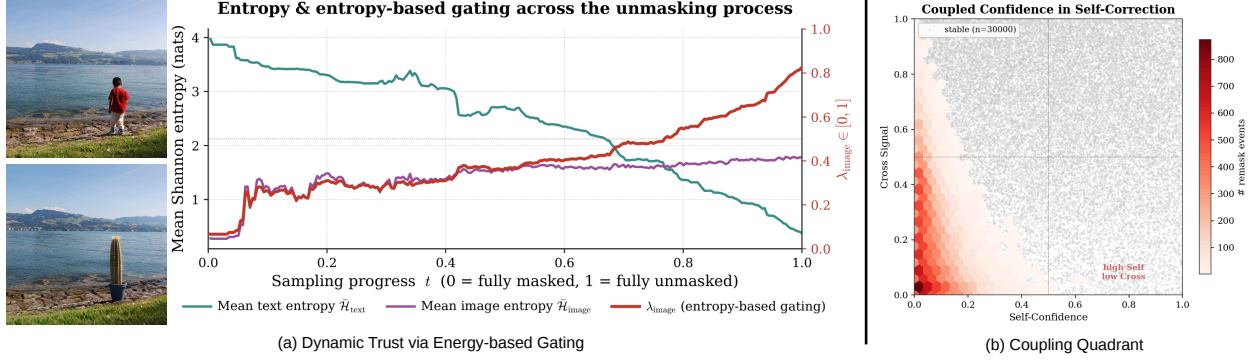


Figure 7 | (a) Per-step text/image entropy and the gate λ_{image} . The image starts confident (source prior), text starts uncertain; λ_{image} rises from ≈ 0.05 to ≈ 0.85 as text commits. (b) Image-side remask events in (Self-Confidence, Cross Signal) space: most fire in the lower-left region; a smaller hotspot at high-Self / low-Cross corresponds to coupling-driven revocations.

Qualitative trajectories expose the uncoupling failure mode. In Figure 6, every baseline violates at least one Nonogram clue (red $\times$), while CO₂Jump satisfies all clues; on the maze, baselines wander into wrong corridors or cut through walls and CO₂Jump reproduces the ground-truth path. The right column exposes *why*: by step 160 of MDM’s trajectory the text answer is correct but the image has drifted onto a different path, because under MDM’s independence factorization the image branch never sees the latest committed text. CO₂Jump’s chain-rule decomposition propagates each text commitment into image scoring at the same step, so both modalities converge on the same solution.

6.3. Ablation Study and Analysis

Ablation. Table 3 removes each mechanism in isolation. Removing Shared Percentile Rank causes the largest drop in image-edit fidelity ($1.93 \rightarrow 1.87$): without a common scale, one of Self-Confidence and Cross Signal arbitrarily dominates the blend. Removing Entropy-Based Gating leaves source-mAP unchanged but cuts target-mAP by 0.030 and overall mAP by 0.015, the gate’s main role is in late-stage sampling where the target image is committed under freshly grounded text. Removing Self-Correction is the most damaging change for understanding (-3.2 on overall mAP) and drops editing by 0.01: without remasking, late-discovered cross-modal contradictions cannot be repaired.

Dynamic trust via entropy-based gating. Figure 7 (a) traces per-step entropies and λ_{image} for a representative editing sample. The image branch starts confident (source-image prior) while text starts uncertain (only the editing prompt as context), so $\lambda_{\text{image}} \approx 0.05$ and the image relies more on self-confidence. As text commits, its entropy falls and the image moves into regions outside its prior; λ_{image} rises to ≈ 0.85 , and the image branch increasingly leans on the text-grounded cross signal —

the empirical signature of the chain-rule decomposition.

Coupling quadrant. Figure 7 (b) shows every image-side remask event in (Self-Confidence, Cross Signal) space. Most events concentrate in the lower-left region where both signals are low, so the coupled confidence $(1 - \lambda) \cdot \text{Self} + \lambda \cdot \text{Cross}$ is small and the death jump is triggered. A smaller hotspot in the high-Self / low-Cross quadrant corresponds to coupling-driven unmasking — the image was locally confident but the text disagreed — and these account for the gap between the full CO₂Jump and the “– Self-Correction” ablation row in Table 3.

7. Conclusion

We presented **Self-Correcting Coupled Markov Jump Processes (SC-CMJP)**, a framework in which the two modalities of a unified MDM negotiate their commitments *within* every denoising step, and CO₂Jump, a training-free single-pass sampler that instantiates it on a frozen backbone. By coupling one modality’s transition rates to the other’s emerging confidence via cross-modal attention, and by allowing committed tokens to be retracted through a remasking jump triggered by cross-modal contradictions, CO₂Jump closes the concurrent-generation loop that prior parallel samplers leave open. CO₂Jump delivers best performance across joint image understanding and editing, as well as visual reasoning tasks and it scales monotonically with denoising steps, evidence that cross-modal coupling compounds across the trajectory. Our experiments instantiate the framework on the canonical text-and-image pair; the SC-CMJP recipe is modality-agnostic, and extending it to audio, video, or structured outputs is left to future work.

A. Qualitative Example from CO₂Jump: Joint Image Editing and Understanding

Figure 8 shows a complete CO₂Jump sample from IMGEDITBENCH: prompt, source image, generated target image, the per-panel scene-graph understanding, and the thinking trace, all produced concurrently in a single denoising loop. Bounding boxes overlaid on each image are exactly those emitted by the model’s own text branch (no external grounder), normalized to a 1001×1001 canvas. The example illustrates the planning-then-execution pattern that drives the gain on extended ImgEditBench: the text branch isolates *shell* as the only dynamic object and keeps every other entity (*tortoise*, *arm*, *eye*, *ground*, *rock*) at identical bounding boxes between source and target, while changing only the shell’s attributes from (*scaly*) to (*smooth*); the image branch realizes precisely that delta, removing the high-frequency scaly pattern on the shell while leaving body, limbs, and surroundings untouched.

<think> The transition from the source to the target image follows a precise, localized logic based on the prompt: "change the tortoise's shell texture to a smooth surface."

Object Categorization:

- **Static Objects:** The 'tortoise' [0,0 to 1000,1000], 'arm' [0, 305, 335, 915; 555, 305, 995, 995], 'eye' [305, 655, 350, 755; 445, 655, 525, 755], 'ground' [0, 800, 1000, 1000], and 'rock' [215, 865, 515, 985] remain constant in both spatial coordinates and visual attributes.

- **Dynamic Objects:** The 'shell' [0, 0, 1000, 450] is the sole dynamic element. While its bounding box remains identical, its internal attributes undergo a significant transformation.

Quantifying Change:

The visual delta is confined to the 'shell' region. In the source state, the shell exhibits a high, high-frequency "scaly" texture characterized by sharp, irregular ridges and Inlay-like patterns. In the state state, the texture is smoothed into a "smooth" finish. The color palette shifts from a dark, desaturated brown to a lighter, uniform tan and tan, and the surface geometry transitions from a complex, segmented scaly pattern to a continuous, smooth, smooth surface.

Critical Alignment Check:

The edit perfectly aligns with the instruction. The "smooth surface" is achieved by removing the high-frequency details of the sc scales while preserving the underlying lighting and lighting interactions on the tortoise's skin, such as the subtle highlights on the eyes and the soft shadows of the snout. There are no hallucinations or unintended alterations to the tortoise's anatomy or the complex textures of the surrounding environment.

Connection to Intent:

The intent was a specific texture modification. By maintaining the static nature of the tortoise and environment while exclusively altering the surface properties of the shell, the edit successfully fulfills the user's intent while compromising the integrity of the the original composition </think>

<answer>

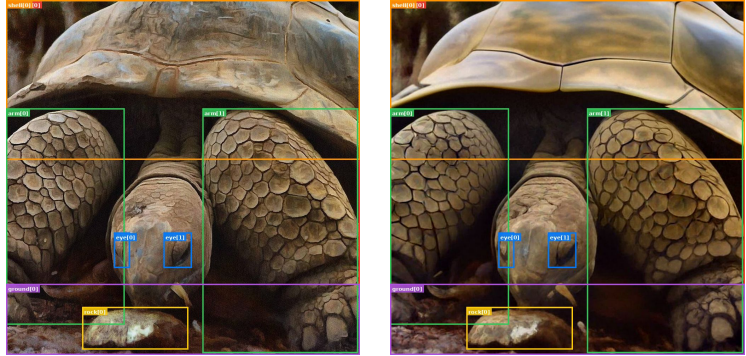
SOURCE IMAGE ANALYSIS:

tortoise (brown, textured, scaly) [0]: [0, 0, 1000, 1000]
 shell (textured, brown, tan, scaly) [0]: [0, 0, 1000, 450]
 arm (scaly, brown, textured) [0]: [0, 305, 335, 915]
 arm (scaly, brown, textured) [1]: [555, 305, 995, 995]
 eye (dark, round) [0]: [305, 655, 350, 755]
 eye (dark, round) [1]: [445, 655, 525, 755]
 ground (dark, earthy) [0]: [0, 800, 1000, 1000]
 rock (brown, textured) [0]: [215, 865, 515, 985]

TARGET IMAGE ANALYSIS:

tortoise (brown, textured, scaly) [0]: [0, 0, 1000, 1000]
 shell (smooth, brown, tan, smooth) [0]: [0, 0, 1000, 450]
 arm (scaly, brown, textured) [0]: [0, 305, 335, 915]
 arm (scaly, brown, textured) [1]: [555, 305, 995, 995]
 eye (dark, round) [0]: [305, 655, 350, 755]
 eye (dark, round) [1]: [445, 655, 525, 755]
 ground (dark, earthy) [0]: [0, 800, 1000, 1000]
 rock (brown, textured) [0]: [215, 865, 515, 985]

</answer>



Prompt: Change the tortoise's shell texture to a smooth surface.

Figure 8 | **Joint image editing with concurrent understanding on JEDIT-1M.** Prompt, model-generated thinking trace, scene-graph source/target analyses, and the source/generated images with predicted bounding boxes overlaid (boxes are the model’s own text-side predictions, not ground truth). The text branch declares only the *shell* as a dynamic object — every other entity has identical bounding boxes in the source and target analyses — and the image branch realizes exactly this localized texture edit: *scaly* → *smooth* on the shell, with body, limbs, ground and rock preserved.

B. Qualitative Example from CO₂Jump: Maze Solving

Figure 9 shows a CO₂Jump sample from JMAZE: the input maze (with green start and red end), the model-generated thinking trace and coordinate path, and the solved maze with the blue path overlaid. Both the textual coordinate sequence and the rendered solution path are produced concurrently in a single denoising loop. The thinking trace reasons globally over the maze — identifying the upper half as a dead-end region, the lower corridors as the main artery, and the central zig-zag through the bottom row as the southern bypass — before committing the explicit (r, c) sequence in `<answer>`, and the image branch traces exactly that sequence, demonstrating that the path drawn on the grid agrees position-by-position with the coordinates the text branch declares.

`<think>`

The maze is a 12x12 grid with the start at (8, 1) and the target at (6, 10). Analyzing the global structure, the upper half of the maze (rows 0–5) is largely disconnected from the target, acting as a massive dead-end region. The primary challenge must occur through the lower quadr, the long horizontal corridors between rows 8 and 11 serve as the main artery connecting the start and right sides.

The path begins by a descent from from the start at (8, 1) through the bottom row 11 via (8, 1) and (9, 1) to reach the, albeit,, bottom-left corner at (9, 0) and (10, 0). Simultaneously, we can see the target at (6, 10) requires be reached from the left through (6, 9) and (6, 8). This creates a clear objective: connect the bottom-left corner (11, 0) to the vertical column on the right..

The central forms involves a series of vertical zig-zags navigate the internal walls. From the bottom corner at (11, 0), the path moves through (11, 1), (11, 2), (10, 2), and (9, 2) to bypass a wall at (9, 3). It then wevers through a vertical at (8, 2) through (8, 3), (9, 3), (10, 3) and (11, 3). The final stretch is a rhythmic sequence along the bottom row: (11, 3) to (11, 4) to (11, 5) to (10, 5) to (10, 6) to (10, 7) to (11, 7) to (11, 8) to (11, 9) to (10, 9) to (9, 9) to (9, 8) to (8, 8). This southern bypass effectively bridges the northern traps and the central barriers. The is completed by moving from (7,8) through (6,8) and (6,9) directly to the target at (6,10).

`</think>`

`<answer>` (8,1) (9,1) (9,0) (10,0) (11,0) (11,1) (11,2) (10,2) (9,2) (8,2) (8,3) (9,3) (10,3) (11,3) (11,4) (11,5) (10,5) (10,6) (10,7) (11,7) (11,8) (11,9) (10,9) (9,9) (9,8) (8,8) (7,8) (6,8) (6,9) (6,10) `</answer>`

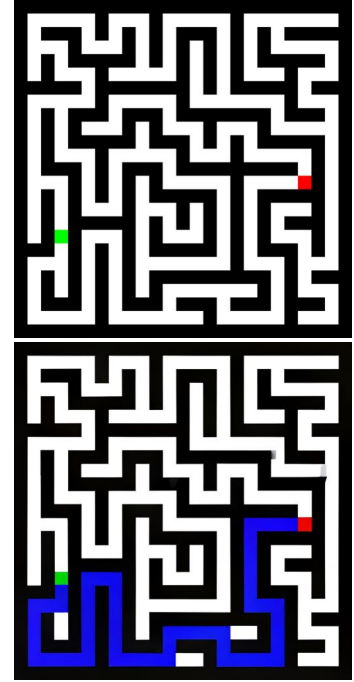


Figure 9 | **Joint maze solving on JMAZE.** CO₂Jump’s thinking trace and answer (left) alongside the input maze (top right; green start, red end) and the solved maze with the blue path overlaid (bottom right). The text branch first reasons globally about which corridors are dead-ends and which form the main artery, then commits the path as an explicit (r, c) coordinate sequence; the image branch traces precisely that sequence on the grid in the same denoising trajectory.

C. Qualitative Example from CO₂Jump: Nonogram Solving

Figure 10 shows a CO₂Jump sample from JNono: the input nonogram (an empty 7×7 grid with row and column clue numbers), the model-generated thinking trace and per-row filled-cell answer, and the solved grid with the corresponding cells filled black. The thinking trace performs bidirectional constraint propagation rather than row-by-row solving: it identifies the most globally-constraining lines first (rows whose largest clue forces a 4-cell block, columns whose clue sums match the grid width), explains how the row and column constraints tighten each other concurrently, and only then commits the row-by-row filled-cell ranges in <answer>; the image branch fills precisely those ranges on the grid, satisfying every clue.

<think>

The 7x7 grid presents a compact puzzle that reveals a staggered, blocky shape. Solving begins by identifying the most constrained lines, specifically 5 and R 6, where the clues (3,3) and (2,3) respectively with the mandatory gap gap account the entire 7. This immediately fixes the bottom-left corner (0-2 and 4-6) and the bottom-right corner (0-1 and 4-6). Simultaneously, Columns 0, 1, 2, 4, and 5, 6 all have clues of 4, which, when combined with the forced fills from R 5 and R 6, create a solid 4x4 block in the lower-left quadrant.

As these vertical and horizontal, the central " (R, 3, 4) 5" clues in Rows 3 and 4 are almost the only way to satisfy a 5 5 clue in a 7x7 space, as they must be the intersection of Columns 0- 4 and Columns 5- 6. This central "5"5 forces the columns to fills in R R R R R in Columns 5 and 6. Consequently, the "2" clues (Row 02) are pushed to the far right, which aligns perfectly with the (2,2) clues in Columns 5 and 6. This leaves a solid 2x2 block in the top-right corner, completing the image..
row 0: (5-6) ; row 1: (5-6) ; row 2: (2-3) ; row 3: (0-4) ; row 4: (0-4) ; row 5: (0-4) ; row 6: (0-1)

</think>

<answer>

row 0: (5-6) ; row 1: (5-6) ; row 2: (2-3) ; row 3: (0-4) ; row 4: (0-4) ; row 5: (0-2)(4-6) ; row 6: (0-1)(4-6) </answer>

						2	2
	4	4	4	3	4	2	2
2							
2							
2							
5							
5							
3	3						
2	3						

						2	2
	4	4	4	3	4	2	2
2							
2							
2							
5							
5							
3	3						
2	3						

Figure 10 | **Joint nonogram solving on JNono.** CO₂Jump’s thinking trace and answer (left) alongside the input puzzle (top right; row clues at left margin, column clues at top margin) and the solved grid with cells filled black (bottom right). The text branch reasons in a bidirectional, constraint-propagation style — starting from the most globally-constrained rows and columns and using each forced fill to tighten the other dimension — before committing the per-row filled-cell ranges; the image branch fills exactly those ranges on the grid in the same denoising trajectory, satisfying every clue.

D. System Prompts for Dataset Curation

D.1. Understanding Prompt: Pixel-Aligned Scene Graph Extraction

The following system prompt was used to extract the structured understanding (bounding boxes and attributes) from the source and target images.

Task: Pixel-Aligned Scene Graph Extraction for Image Editing Pairs.

Objective:

Analyze the transition from the Source Image to the Target Image. You must extract a high-fidelity grounding dataset that maps how visual tokens change in response to the Editing Prompt. Focus on maintaining spatial "anchors" for static regions while precisely detailing the "delta" in edited regions.

Inputs:

1. Source Image: (The original state)
2. Target Image: (The modified state)
3. Editing Prompt: "{editing_prompt_var}"

Strict Constraints:

THINKING BEFORE ANSWERING: In your <think> tags, you MUST explicitly compare the two images. List:

1. Static entities (Background/Context).
2. Transformed entities (Attribute/State changes).
3. New or Deleted entities (Structural changes).

FIDELITY OVER INTENT: Describe what is visually rendered in the Target Image. If the prompt asks for a "red car" but the Target Image contains a "pink car," you MUST label it as "pink car."

COMPLETENESS: Annotate the main subject, all objects involved in the edit, and key background anchors (e.g., floor, sky, walls) to provide global context.

COORDINATES: Use normalized integer bounding boxes [x1, y1, x2, y2] on a [0-1000] scale. Bounding boxes must be tight.

IDENTITY CONSISTENCY: For any entity present in both images, the 'category_name' and '[index]' MUST be identical.

PER-CATEGORY INDEXING: Every new object category MUST start its index at [0]. For example, the first 'car' is 'car [0]', and the first 'person' is 'person [0]'. Do NOT use a single global incrementing index for the whole scene.

STRUCTURAL LOGIC: If an object is deleted in the Target, do not list it in the TARGET IMAGE ANALYSIS. If an object is added, list it only in the TARGET section.

Prohibited Actions:

DO NOT use generic placeholder strings (e.g., "category_name", "attribute_list").

DO NOT include the brackets around the category name or attributes in the final answer; only the index should remain in brackets as per format.

Output Format:

<answer>

SOURCE IMAGE ANALYSIS:

category_name (attribute_list) [index]: [x1, y1, x2, y2]

TARGET IMAGE ANALYSIS:

category_name (attribute_list) [index]: [x1, y1, x2, y2]

</answer>

Reference Example:

<answer>

SOURCE IMAGE ANALYSIS:

mountain (snowy, jagged) [0]: [0, 50, 1000, 450]

person (black hoodie, walking) [0]: [420, 550, 580, 910]

person (blue jeans, standing) [1]: [600, 550, 700, 910]

TARGET IMAGE ANALYSIS:

mountain (snowy, jagged) [0]: [0, 50, 1000, 450]

person (yellow raincoat, walking) [0]: [420, 550, 580, 910]

person (blue jeans, standing) [1]: [600, 550, 700, 910]

</answer>

D.2. JEDIT-1M Thinking Prompt: Reasoning Trace Synthesis

The following system prompt was used with Qwen3-VL-235B to synthesize the logic-based reasoning trace for the JEDIT-1M dataset.

```
Task: Synthesize a Logic-Based Reasoning Trace for Image Editing.

### Inputs:
1. Source and Target Images: Visual evidence.
2. Editing Prompt (The intended transformation): {editing_prompt_var}
3. Image Understanding (Bounding box coordinates and object labels
for both states): {answer}

### Objective:
Synthesize a logic-based reasoning trace that analyzes the visual
transition relative to the Editing Prompt. Critically evaluate
whether the target state successfully fulfills the prompt's
instructions or deviates from the intended logic.

### Strict Reasoning Guidelines:
Categorize Objects: Group objects into: Static (persistent
coordinates/attributes), Dynamic (modified coordinates/attributes),
and Deleted/Added (present in only one state).

Quantify Change: Reference bounding box shifts or attribute
updates as direct evidence of the transition.

Critical Alignment Check: Explicitly compare the visual delta
against the Editing Prompt. Identify hallucinations or instruction
mismatches.

Connect to Intent: Explain how the detections fulfill-or fail
to fulfill-the specific requirements of the Editing Prompt.

Constraint: Keep the response dense and information-rich, within
a max of 350 words. Avoid generic phrases. Start with the direct
logical flow.

### Output Format:
<think> [Your concise logical trace here] </think>
```

D.3. Thinking Prompt: Maze Solving Parallel Reasoning

The following system prompt was used to synthesize the parallel-reasoning trace for the maze-solving problem in the JMaze-200K dataset.

```

Task: Synthesize a parallel-reasoning trace for a maze-solving problem.

### Inputs:
1. Input maze image: walls, a green start cell, and a red end cell. No path drawn.
2. Output maze image: the same maze with the correct solution path drawn in blue.
3. Maze structure as text (adjacency list + start + end):
{prompt}
4. Ground-truth solution path:
{answer}

### Objective:
Write a natural reasoning trace that reflects a global-to-local refinement process, illustrating
how the maze is solved simultaneously rather than purely sequentially.
Briefly describe the maze boundaries (grid size, start position, end position).
Identify the global structure first: note major bottlenecks, large dead-end regions to mask
out, or key anchor points/corridors between the start and end.
Describe the path formation as a concurrent process-e.g., establishing a
bridge through a central corridor while simultaneously connecting the start
and end regions to that main artery, verifying local wall-openings via the
adjacency list.
Write in natural prose, as a holistic chain of thought. No bullet lists.
Keep it under 350 words.

### Output format (strict - respond with exactly this tag and nothing else):
<think>
{}[Your reasoning trace]
</think>

```

D.4. Thinking Prompt: Nonogram Parallel Reasoning

The following system prompt was used to synthesize the parallel-reasoning trace for the nonogram puzzle in the JNono-200K dataset.

```

Task: Synthesize a parallel-reasoning trace for a nonogram (picross) puzzle.

### Inputs:
1. Source image: empty NxN grid with row clue numbers along the left margin and column clue
numbers along the top margin.
2. Target image: the same grid with cells filled to satisfy every clue, revealing a pixel-art
picture.
3. Puzzle structure as text (size + row clues + column clues):
{prompt}
4. Ground-truth solution (filled cell ranges per row):
{answer}

### Objective:
Write a holistic reasoning trace that reflects parallel constraint propagation,
NOT row-by-row sequential solving.
Briefly describe the puzzle (grid size, rough shape of the revealed picture).
Identify the most globally-constraining lines first: rows or columns where
the sum of clues plus required gaps equals N (full-line forces), or lines
whose single largest clue produces "definite-fill" cells via the
left-most / right-most overlap argument.
Describe how column constraints and row constraints tighten each other
concurrently: a forced fill produced by a row clue confirms a position in
some column's clue list, which can in turn unlock more rows.
Avoid sequential "first solve row 0, then row 1" prose. Emphasize the
bidirectional, global interplay between row and column constraints.
Write in natural prose, no bullet lists. Keep it under 250 words.

### Output format (strict - respond with exactly this tag and nothing else):
<think>
{}[Your reasoning trace]
</think>

```

References

- M. Arriola, A. Gokaslan, J. T. Chiu, J. Han, Z. Yang, Z. Qi, S. S. Sahoo, and V. Kuleshov. Interpolating autoregressive and discrete denoising diffusion language models. In *ICLR*, 2025.
- J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *NeurIPS*, 2021.
- S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu. Qwen3-vl technical report, 2025. URL <https://arxiv.org/abs/2511.21631>.
- D. Berghaus, K. Cvejski, P. Seifner, C. Ojeda, and R. J. Sanchez. Foundation inference models for markov jump processes. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=f4v7cmm5sC>.
- A. Campbell, J. Benton, V. De Bortoli, T. Rainforth, G. Deligiannidis, and A. Doucet. A continuous time framework for discrete denoising models. *NeurIPS*, 2022.
- A. Campbell, J. Yim, R. Barzilay, T. Rainforth, and T. Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *ICML*, 2024.
- J. Chen, W. Song, P. Ding, Z. Zhou, H. Zhao, F. Tang, D. Wang, and H. Li. Unified diffusion VLA: Vision-language-action model via joint discrete denosing diffusion process. In *ICLR*, 2026.
- X. Chen, Z. Wu, X. Liu, Z. Pan, W. Liu, Z. Xie, X. Yu, and C. Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.
- S. Cheng, Y. Bian, D. Liu, L. Zhang, Q. Yao, Z. Tian, W. Wang, Q. Guo, K. Chen, B. Qi, and B. Zhou. Sdar: A synergistic diffusion-autoregression paradigm for scalable sequence generation. *ArXiv*, abs/2510.06303, 2025.
- C. Deng, D. Zhu, K. Li, C. Gou, F. Li, Z. Wang, S. Zhong, W. Yu, X. Nie, Z. Song, G. Shi, and H. Fan. Emerging properties in unified multimodal pretraining, 2025. URL <https://arxiv.org/abs/2505.14683>.
- I. Gat, T. Remez, N. Shaul, F. Kreuk, R. T. Chen, G. Synnaeve, Y. Adi, and Y. Lipman. Discrete flow matching. *NeurIPS*, 2024.
- D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of computational physics*, 22(4):403–434, 1976.
- D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25):2340–2361, 1977.
- D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of chemical physics*, 115(4):1716–1733, 2001.
- R. J. Glauber. Time-dependent statistics of the ising model. *Journal of mathematical physics*, 4(2):294–307, 1963.

- Google. Gemini 3 flash, 2026. URL <https://gemini.google.com/>.
- Z. Huang, Y. Wang, Z. Chen, and G.-J. Qi. Don't settle too early: Self-reflective remasking for diffusion language models. *ArXiv*, abs/2509.23653, 2025.
- M. I. Ivanitskiy, R. Shah, A. F. Spies, T. Räuker, D. Valentine, C. Rager, L. Quirke, C. Mathwin, G. Corlouer, C. D. Behn, and S. W. Fung. A configurable library for generating and manipulating maze datasets, 2023. URL <http://arxiv.org/abs/2309.10498>.
- J. Kim, S. Kim, T. Lee, D. Z. Pan, H. Kim, S. M. Kakade, and S. Chen. Fine-tuning masked diffusion for provable self-correction. *ArXiv*, abs/2510.01384, 2025a.
- J. Kim, K. Shah, V. Kontonis, S. M. Kakade, and S. Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. *ICML*, 2025b.
- J. Lezama, T. Salimans, L. Jiang, H. Chang, J. Ho, and I. Essa. Discrete predictor-corrector diffusion models for image synthesis. In *International Conference on Learning Representations*, 2023.
- T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.
- A. Lou, C. Meng, and S. Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *ICML*, 2024.
- S. Nie, F. Zhu, Z. You, X. Zhang, J. Ou, J. Hu, J. ZHOU, Y. Lin, J.-R. Wen, and C. Li. Large language diffusion models. In *NeurIPS*, 2025. URL <https://openreview.net/forum?id=KnqiC0znVF>.
- J. Ou, S. Nie, K. Xue, F. Zhu, J. Sun, Z. Li, and C. Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *ICLR*, 2025. URL <https://openreview.net/forum?id=sMyXP8Tanm>.
- Y. Ouyang, P. Hu, Z. Wan, Z. Wang, L. Xie, D. Bespalov, Y. N. Wu, G. Cheng, H. Zha, and Q. Sun. Training-free self-correction for multimodal masked diffusion models, 2026. URL <https://arxiv.org/abs/2602.02927>.
- Z. Peng, Z. Bezemek, S. Patel, J. Rector-Brooks, S. Yao, A. Tong, and P. Chatterjee. Path planning for masked diffusion model sampling. *ArXiv*, abs/2502.03540, 2025.
- A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- S. S. Sahoo, M. Arriola, Y. Schiff, A. Gokaslan, E. Marroquin, J. T. Chiu, A. Rush, and V. Kuleshov. Simple and effective masked diffusion language models. *NeurIPS*, 2024.
- J. Shi, K. Han, Z. Wang, A. Doucet, and M. Titsias. Simplified and generalized masked diffusion for discrete data. *NeurIPS*, 37:103131–103167, 2024.
- Y. Tian, L. Yang, J. Yang, A. Wang, Y. Tian, J. zheng, H. Wang, Z. Teng, Z. Wang, Y. Wang, Y. Tong, M. Wang, and X. Li. MMAaDA-parallel: Multimodal large diffusion language models for thinking-aware editing and generation. In *ICLR*, 2026. URL <https://openreview.net/forum?id=mKQAd11ovN>.
- A. van den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017.

- D. von Rutte, J. Fluri, Y. Ding, A. Orvieto, B. Scholkopf, and T. Hofmann. Generalized interpolating discrete diffusion. *ICML*, 2025.
- G. Wang, Y. Schiff, S. S. Sahoo, and V. Kuleshov. Remasking discrete diffusion models with inference-time scaling. In *NeurIPS*, 2025. URL <https://openreview.net/forum?id=IJryQA0yOp>.
- C. Wei, Z. Xiong, W. Ren, X. Du, G. Zhang, and W. Chen. Omniedit: Building image editing generalist models through specialist supervision. In *ICLR*, 2025. URL <https://openreview.net/forum?id=Hlm0cga0sv>.
- J. Xie, Z. Yang, and M. Z. Shou. Show-o2: Improved native unified multimodal models. In *NeurIPS*, 2025. URL <https://openreview.net/forum?id=7VMg7Jb7AL>.
- Y. Xin, Q. Qin, S. Luo, K. Zhu, J. Yan, Y. Tai, J. Lei, Y. Cao, K. Wang, Y. Wang, J. Bai, Q. Yu, D. Jiang, Y. Pu, H. Chen, L. Zhuo, J. He, G. Luo, T.-X. Li, M. Hu, J. Ye, S. Ye, B. Zhang, C. Xu, W. Wang, H. Li, G. Zhai, T. Xue, B. Fu, X. Liu, Y. Qiao, and Y. Liu. Lumina-dimoo: An omni diffusion large language model for multi-modal generation and understanding. *ArXiv*, abs/2510.06308, 2025.
- L. Yang, Y. Tian, B. Li, X. Zhang, K. Shen, Y. Tong, and M. Wang. MMaDA: Multimodal large diffusion language models. In *NeurIPS*, 2025. URL <https://openreview.net/forum?id=wczmXLuLGd>.
- J. Ye, Z. Xie, L. Zheng, J. Gao, Z. Wu, X. Jiang, Z. Li, and L. Kong. Dream 7b: Diffusion large language models. *ArXiv*, abs/2508.15487, 2025a.
- Y. Ye, X. He, Z. Li, B. Lin, S. Yuan, Z. Yan, B. Hou, and L. Yuan. Imgedit: A unified image editing dataset and benchmark. In *NeurIPS*, 2025b. URL <https://openreview.net/forum?id=uUCSrMlfD3>.
- Y. Zhao, J. Shi, F. Chen, S. Druckmann, L. Mackey, and S. Linderman. Informed correctors for discrete diffusion models. In *NeurIPS*, 2026. URL <https://openreview.net/forum?id=HDeIb67lJe>.