Inference-Time Scaling in Diffusion Models: A Beam Search Approach to Visual Sequences Generation

Guilherme Fernandes[1], Vasco Ramos[1], Regev Cohen[2], Idan Szpektor[2], Joao Magalhaes [1]
[1]NOVA LINCS, NOVA School of Science and Technology, Portugal
[2]Google Research
jmag@fct.unl.pt, szpektor@google.com

Abstract

While diffusion models produce high-quality images, they often lack consistency in sequences, especially for non-linear visual sequences. To solve this, we introduce an inference-time scaling beam search strategy for latent space exploration, enabling the conditional generation of coherent image sequences through beam search diffusion. In contrast to earlier methods that rely on fixed latent priors, our method dynamically samples past latents to search for an optimal sequence of latent representations, ensuring coherent visual transitions. As the latent denoising space is explored, the beam search graph is pruned with a cross-attention mechanism that efficiently scores search paths, prioritizing alignment with both textual prompts and visual context. Human and automatic evaluations confirm that BeamDiffusion outperforms other baseline methods, producing full sequences with superior coherence, visual continuity, and textual alignment.[1]

1 Introduction

Image diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) have made significant advancements in generating high-quality images. These models, especially when combined with text prompts, have shown great potential in producing detailed and contextually accurate visuals (Rombach et al., 2022; Podell et al., 2024; Saharia et al., 2022). Despite the impressive results at generating individual images based on specific text prompts, diffusion models face challenges when it comes to creating coherent sequences of images. In most cases, each image is generated independently based on its own prompt, which does not naturally ensure visual continuity across a *sequence of correlated prompts*. This problem is further compounded when the narrative is non-linear, where a scene may be connected not only to the immediate prior scene but also to earlier scenes in the sequence.

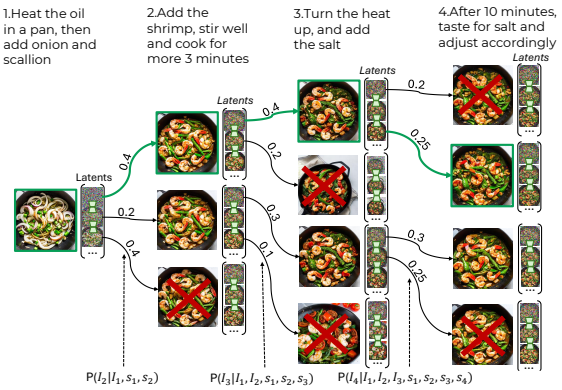


Figure 1: BeamDiffusion leverages search in the latent denoising space to generate a coherent sequence of images from stepwise text instructions. By conditioning each image I_j on prior latents s_j, the model forms a continuous information chain. It uses a pruning process to discard poor candidates and propagate only the best latents to the next step.

We propose to research beam search as a scientifically well-grounded solution to this problem. Beam search is a core text generation technique for tasks like machine translation and standard inference (Kasai et al., 2024; Wu et al., 2016; Koehn and Knowles, 2017; Rush et al., 2015), allowing the model to explore multiple possible paths during the generation process, evaluating different trajectories and refining its predictions. However, despite its success in other domains, it has been barely explored in this context, leaving its potential untapped. In full image sequence generation, beam search could help navigate the complex latent space, refining coherence by adjusting to multiple text prompts or adding semantic cues, as illustrated in Figure 1. In this work, we revisit the latent diffusion framework with a focus on structured sequence generation. We aim to dynamizally sample the latent space of past diffusion processes and

[1]Code and data are released under CC-BY 4.0 at https://huggingface.co/Gui28F/BeamDiffusion.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 44886–44909
October 24-29, 2026 ©2026 Association for Computational Linguistics

propose the BeamDiffusion method for improving multi-image visual consistency by leveraging shared latent representations in a principled way. BeamDiffusion explores the latent denoising trajectories with a beam search strategy for ensuring a conditional dependency between diffusion processes. As illustrated in Figure 2a, our approach leverages a sampling strategy in the latent space to identify the best sequence of images, minimizing the risk of falling into suboptimal or incoherent paths, while maintaining visual continuity across images. As the beam diffusion algorithm traverses the latent space, the number of hypothesis grows and the search paths with the lowest probability are pruned to keep a fixed number of candidate paths. To this end, we incorporate a cross-attention mechanism (Lin et al., 2022) that evaluates and scores candidate image paths based on their alignment, providing a reliable quality measure for each path, allowing us to retain the best ones while pruning the least likely. To test the generalization and robustness of BeamDiffusion, we evaluated it in two diffusion architectures (LDM (Rombach et al., 2022) and DiT (black-forest labs, 2024)) and two domains (recipes and visual storytelling domains). Results obtained with human annotations, automatic evaluations and LLM-as-a-judge, all showed that the proposed approach outperforms baseline methods for generated sequence quality, showcasing its adaptability to diverse contexts.

2 Related Work

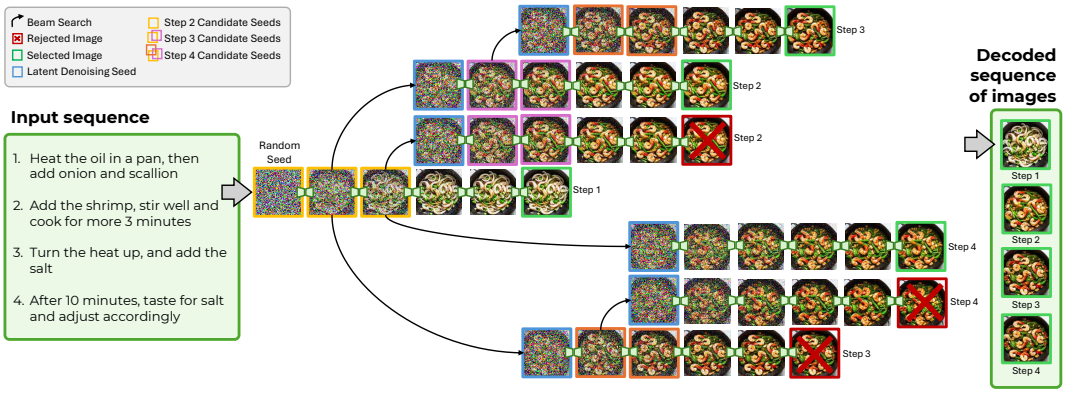
Decoding methods are crucial for enhancing the quality of large language models (LLMs). Techniques such as beam search, top-k and nucleus sampling are widely studied for their effectiveness in balancing computational cost with generating coherent outputs (Shi et al., 2024; Holtzman et al., 2020a; Furniturewala et al., 2024). Beam search has also proven effective in tasks like machine translation (Meister et al., 2022), image captioning (Vijayakumar et al., 2018; Ke et al., 2019), and visual storytelling (Hsu et al., 2018).

Ensuring coherence remains a major challenge in multimodal sequence generation (Geyer et al., 2024; Liu et al., 2024b; Chen et al., 2024; Khachatryan et al., 2023). Some approaches, like AR-LDM (Pan et al., 2024), encode the context of caption-image pairs into multimodal representations to maintain consistency across outputs. Likewise, Rahman et al. (2023) incorporates the history of U-net (Ronneberger et al., 2015; Salimans et al., 2017) latent vectors to preserve narrative coherence. Other approaches, such as Gill (Koh et al., 2023), combine LLMs with image encoder-decoder systems. StoryDiffusion (Zhou et al., 2024), changes the U-Net's cross-attention mechanism enabling training-free consistency by sampling tokens from previous images to maintain coherence in newly generated frames. StoryGen (Liu et al., 2024a) denoises the previous image and the resulting features serve as conditioning for the subsequent generation of images. GenHowTo (Soucek et al., 2024) uses a control net during inference to project the previous image directly into the latent space to guide the diffusion process for the next image step. Stacked-Diffusion (Menon et al., 2024) processes stacked image sequences as a single latent representation allowing the attention mechanism to consider all steps together, enhancing coherence across steps. In (Bordalo et al., 2024), authors proposed to use latent information from previous steps to guide coherent generation of subsequent images relying on a greedy heuristic-based selection. CoSeD (Ramos et al., 2025) improved on this idea and introduced a contrastive learning approach (Chen et al., 2020) to rank images generated from different latent seeds. However, both approaches focused on local refinements rather than optimizing throughout the sequence. Our beam search approach extends these local selection criteria to perform global optimization. While effective for certain tasks, these models suffer from alignment issues, with retrieved or generated images failing to match the narrative context, and they too often require complex training.

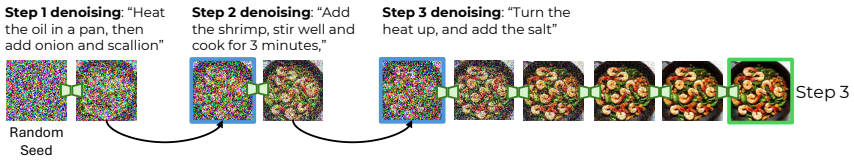
Our method addresses these previous constraints by maintaining image sequences' consistency via a beam search for optimal seeds, similar to methods such as (Singhal et al., 2025) that view image generation as a search process.

3 Generating Non-Linear Visual Sequences

Generating a full sequence of visually coherent images from textual scene descriptions is a challenging research problem. Given a sequence of steps $S = \{s_1, s_2, \ldots, s_L\}$, where each sequence step s_j is a textual description, our goal is to generate a corresponding sequence of images $\{I_1, I_2, \ldots, I_L\}$ that maintain consistency across all steps of the sequence. Diffusion Models generate high-quality images in a compressed latent space, but typically



(a) The beam denoising search tree.



(b) Example of one complete decoded denoising process using contextual captions.

Figure 2: The BeamDiffusion model works in the denoising latent space by denoising the initial seed to latents and them decoding different beam paths. This shows how our method evolves in the latent space and how it can explore the latent space.

treat each image independently from a randomly initialized latent z_T.

Within this framework, Latent Diffusion Models (LDMs) (Rombach et al., 2022) and Diffusion Transformers (DiTs) (Peebles and Xie, 2023) share a common structure: both encode an input image I into a latent representation z using an encoder $\mathcal{E}$ and reconstruct it through a decoder $\mathcal{D}$ (Rombach et al., 2022). The reverse diffusion process is performed by a denoising backbone ϵ_θ, which can be a U-Net (Ronneberger et al., 2015) (for LDMs) or a transformer (Peebles and Xie, 2023) (for DiT), and incorporates cross-attention to condition the denoising iterations on an external input, e.g., the output embeddings of a text encoder $\tau_\theta(s)$. Both models are trained to iteratively estimate a denoised latent with the loss:

$$L_{\text{diff}} = \mathbb{E}_{\mathcal{E}(I),s,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(s))\|_2^2\right] \quad (1)$$

This formulation highlights that each denoising process is *agnostic to other generations*, making it challenging to maintain coherence across a sequence of related prompts. This is particularly important in non-linear narratives (Smilevski et al., 2018), where preserving character identity, lighting, and environmental details is crucial. To address these challenges, prior work (Rahman et al., 2023; Liu et al., 2024a; Soucek et al., 2024; Bordalo et al., 2024; Ramos et al., 2025) has proposed *reusing latent denoising iterations from earlier steps*, raising the key question: *how can we determine the optimal set of latent states that maximizes coherence across different generations?*

In the following subsections, we introduce BeamDiffusion (Sec. 3.1), detail latent space exploration with diffusion beams (Sec. 3.2), and outline our beam pruning strategy (Sec. 3.3).

3.1 Beam Diffusion Model

To address the limitations of standard diffusion models in generating coherent image sequences, we introduce the BeamDiffusion model. The proposed approach aims to enhance both the consistency and quality of image sequences by adopting a strategy inspired by beam search, but with a twist, it operates on latent denoising space. BeamDiffusion incorporates latent representations from previously generated images, Figure 2a. These latent representations guide the generation of new candidates, ensuring that each new image aligns with the context of the evolving sequence. This approach allows visual elements from previous images to be carried over or referenced, maintaining contextual consistency and continuity, while also adapting to the next scene description.

In this article, the term *step* refers to an element

in the input sequence; and the term *iteration* refers to one denoising iteration of the diffusion model.

3.2 Latent Space Exploration with Diffusion Beams

To generate coherent and diverse image sequences, we explore the latent space of diffusion models using a beam search-inspired strategy. For each sequence step j we generate a set of candidate images $\mathcal{I}_j = \{I_j^{(1)}, I_j^{(2)}, \dots\}$, where each $I_j^{(k)}$ is conditioned on a different beam path.

Sampling the Denoising Latent Space. In our approach, the denoising process for each sequence step is conditioned by representations from prior steps, helping the model maintain coherence across the sequence. In the first step of the sequence, s_1, we generate multiple candidate images by running the diffusion process with r random noise seeds. This strategy enables broader exploration of the latent space and reduces Bayes risk (Bertsch et al., 2023), as detailed in Appendix D. For every generation process corresponding to a sequence step s_i, we store the first N latent space vectors $z_i = \{z_{i,t}\}_{t \in \subseteq \{1,\dots,N\}}$ produced during the denoising iterations. On each step prompt $s_{j>1}$ we need to generate diverse samples that (a) explore different denoising trajectories in the latent space, and (b) are strongly correlated with previous samples. Leveraging the stored latent space vectors, we sample the latent subspace of n previous denoising processes. Hence, for steps $j > 1$, to enable contextual consistency, we first accumulate representations from the previous m steps:

$$\mathcal{Z}_j = \bigcup_{i \in \{j-m,\dots,j-1\}} \{z_i\}. \tag{2}$$

To generate new candidate images $\mathcal{I}_j = \{I_j^{(1)}, \dots, I_j^{(k)}, \dots, I_j^{(N \cdot m)}\}$ at step j, we condition the diffusion model on the latent trajectories in $\mathcal{Z}_j$. Specifically, for each latent trajectory $z \in \mathcal{Z}_j$, the model performs a new diffusion process to produce a corresponding image candidate $I_j^{(k)}$.

Diffusion Beams. Inspired by beam search (Wiseman and Rush, 2016), which maintains and extends the most promising candidates at each sequence step, we introduce *diffusion beams* to structure sequential image generation. At each sequence step j, the set of candidate images $\mathcal{I}_j$ is used to compute the set of candidate diffusion beams $\hat{B}_j$: for each step j we consider all the beams existing in step $j-1$ and extend each one of such b beams $\hat{B}_{j-1}^{(b)}$ by pairing it with a new candidate image $I_j^{(k)} \in \mathcal{I}_j$. The resulting beams are defined as

$$\hat{B}_j = \bigcup_{(b,k)} \left\{ \hat{B}_{j-1}^{(b)} \oplus I_j^{(b \cdot k)} \mid I_j^{(b \cdot k)} \in \mathcal{I}_j \right\}, \tag{3}$$

where $\oplus$ denotes the concatenation of a prior beam with a new candidate image, and $(b \cdot k)$ indexes the pairing of beam b and candidate image $I_j^{(k)}$. This process builds a tree of candidate sequences, enabling the model to explore diverse yet coherent visual narratives over time.

3.3 Pruning Diffusion Beam Paths

Tracking multiple *diffusion beams* enables BeamDiffusion to maintain visual coherence while exploring diverse continuations across a sequence. However, as the sequence length increases, the number of candidate beams grows rapidly, making it computationally infeasible to retain them all. To address this, we apply a pruning strategy based on a learned contrastive scoring function, which allows us to retain only the most promising beams while preserving diversity.

At each sequence step j, we construct a set of candidate beams $\hat{B}_j$. Each candidate beam $\hat{B}_j^{(b)}$ is formed by extending a beam from the previous step $\hat{B}_{j-1}^{(b)}$ with a newly generated image $I_j^{(k)}$. A complete candidate beam at step j is a sequence of images $\hat{B}_j^{(b)} = [I_1^{(b)}, I_2^{(b)}, \dots, I_j^{(b)}]$, one per step. To evaluate the overall quality of a beam, we use a contrastive classifier (Ramos et al., 2025), denoted by φ, which is trained as described in Appendix I. Specifically, for each image $I_i^{(b)}$ in the beam, the classifier φ computes a compatibility score based on the image's prompt s_i and the previous images in the beam $B_{i-1}^{(b)}$, which includes earlier images and prompts in the same beam. The total beam score is then the sum of these per-image scores over all sequence steps up to j:

$$score(\hat{B}_j^{(b)}) = \sum_{i=1}^{j} \log \left(\frac{\exp\left(\varphi(I_i^{(b)}, s_i, B_{i-1}^{(b)})\right)}{\sum_{I_i' \in \mathcal{I}_i} \exp\left(\varphi(I_i', s_i, B_{i-1}^{(b)})\right)} \right), \tag{4}$$

where $\mathcal{I}_i$ is the set of all candidate images at sequence step i. The contrastive formulation ensures that each image is evaluated not in isolation, but relative to other candidates, capturing how well it fits the evolving sequence context.

Once scored, we prune the beam candidates by selecting the top w scoring beams:

$$B_j = \text{argmax}_{B_j \subseteq \hat{B}_j, |B_j| = w} \left(\sum_{i=1}^{w} score(\hat{B}_j^{(i)}) \right), \tag{5}$$

where B_j denotes the final set of retained beams for sequence step j. To balance exploration and efficiency, pruning is disabled for the initial steps, allowing all candidates to be retained initially. After a few step, we apply pruning to keep only the top w beams. This strategy ensures wide exploration early on and focused, high-quality continuation in later sequence steps.

4 Experimental Setup

In this section, we describe the experimental setup for evaluating BeamDiffusion's performance in generating coherent image sequences.

Implementation details. To ensure visual continuity across generations, BeamDiffusion replaces random seed initialization with a latent buffering strategy, caching early latents in a dedicated buffer and resuming the denoising process from these stored states to establish a continuous path across the sequence. Beam Diffusion remains agnostic to the specific pruning classifier used, requiring only that the model be trained for next-image prediction. For our implementation, we utilize the publicly available cross-attention classifier by (Ramos et al., 2025), which effectively scores candidate sequences by processing their integrated CLIP-T and CLIP-I path representations.

Decoding strategies. Our proposed method, BeamDiffusion, uses a beam search over latent sequences to generate coherent image sequences from textual prompts. To evaluate the effectiveness of this decoding strategy, we also compare it against two alternative approaches applied to the same model: **Greedy / CoSeD**$_{len=1}$ (Ramos et al., 2025), which selects at each step the top-scoring image according to CLIP similarity from the first few latents of the previous step, focusing on immediate best-match selection; and **Nucleus Sampling** (Holtzman et al., 2020b), a probabilistic strategy that samples images from the subset whose cumulative CLIP probability exceeds a threshold p, encouraging diversity while still prioritizing high-probability outcomes. All strategies are applied to the same sequences using two underlying diffusion backbones, FLUX.1-dev (black-forest labs, 2024) and SD2.1 (Rombach et al., 2022), allowing a direct comparison of their ability to maintain sequence-level coherence across different models.

Datasets. We used publicly available manual tasks in the Recipes domain (Bordalo et al., 2024), along with a small portion of the Visual Storytelling (VIST) dataset (Huang et al., 2016), using an equal number of tasks from each. Recipes provide detailed, step-by-step instructions, while the sampled VIST tasks offer additional diversity for training the classifier. This combination allows us to evaluate BeamDiffusion's performance on both familiar and novel task types. Together, these datasets enable a thorough assessment of sequence coherence, semantic alignment, and visual consistency. More details can be found in Appendix F.

Baselines. To assess the performance of BeamDiffusion, we compared against several competitive baselines that aim to generate coherent image sequences from text: **GILL (Koh et al., 2023)**, **StackedDiffusion (Menon et al., 2024)**, **StoryDiffusion (Zhou et al., 2024)**, and **One-Prompt-One-Story (1P1S) (Liu et al., 2025)**.

Human Evaluation. To find the optimal hyperparameters, the annotators compared pairs of beam search configurations to select the optimal beam width (w in Eq. 5) and steps back (m in Eq. 2). The winning configuration is selected iteratively by comparing different configurations and preserving always the better one. See Appendix B for the detailed results. Having chosen the optimal configuration, we compared BeamDiffusion to other methods. In the second annotation experiment, BeamDiffusion was compared with other baseline models. Annotators independently assessed the sequences, selecting the best method according to semantic and visual consistency with the possibility of choosing multiple options or none. For further information, see Appendix E.

LLM-as-a-judge Evaluation. To further validate our findings and annotate a wider range of tasks, we adopt the LLM-as-a-judge paradigm, leveraging Gemini, to evaluate both configurations and baselines. In Appendix E we detail how the selection process is aligned with human annotators.

Human and LLM-as-a-judge Agreement. We conducted an alignment assessment to measure agreement between human annotators and the LLM-as-a-judge. Seven annotators rated 26 tasks, which we compared to the model's evaluations. To quantify the agreement between the two, we computed Fleiss' kappa (Fleiss, 1971), obtaining a value of 70%, which indicates a substantial level of agreement. Building on this strong alignment,

we employed the LLM-as-a-judge to evaluate the remaining sequences.

Automatic Metrics. Following prior work (Bordalo et al., 2024; Ramos et al., 2025; Liu et al., 2024a), we evaluate each method using two complementary criteria: (1) *Visual Consistency*, measured via average image similarity using DINO (Oquab et al., 2024) (DINO-I) and CLIP (Radford et al., 2021) (CLIP-I); and (2) *Semantic Alignment*, computed with CLIP (CLIP-T) by comparing each image to the average of all step texts up to and including that step. To ensure comparability across metrics, we normalize CLIP-I, CLIP-T, and DINO-I by their respective maxima. We further define CLIP* as the product of CLIP-I and CLIP-T, and DINO* as the product of DINO-I and CLIP-T, balancing visual and semantic quality while emphasizing alignment with the stepwise input text. To better capture real performance, the results for DINO-I, CLIP-I, CLIP-T, CLIP*, and DINO* are reported as *win percentages* rather than mean values. This avoids overestimating methods that perform very well on some sequences but poorly on others, which could otherwise produce artificially high averages despite inconsistent quality. In addition, we adopt metrics from StackedDiffusion (Menon et al., 2024) to evaluate task-specific aspects of the generated sequences: *Goal Faithfulness*, measuring alignment of the final image with the overall textual goal; *Step Faithfulness*, measuring alignment of each intermediate image with its corresponding step text; and *Cross-Image Consistency*, capturing visual coherence across images in the sequence. These metrics complement the previous ones by explicitly assessing semantic fidelity and sequence-level consistency. See Appendix E for further details.

5 Results and Discussion

In this section, we evaluate the effectiveness of BeamDiffusion through three complementary approaches: human evaluations, LLM-as-a-judge, and automatic metrics. Additional details on the annotation setup are provided in Appendix E.

5.1 Test-time Scaling Methods

To evaluate the effectiveness of different decoding strategies, we compare our proposed method, BeamDiffusion, against Greedy/CoSeD$_{len=1}$ (Ramos et al., 2025) and Nucleus Sampling (Holtzman et al., 2020b) using the same sequences of prompts with two types of diffusion backbones.

Table 1: Comparison of human (H) and LLM-as-a-judge (L) evaluations of decoding strategies across two backbones.

	FLUX.1-dev		SD2.1	
Method	**H (%)**	**L (%)**	**H (%)**	**L (%)**
Greedy/CoSeD$_{len=1}$	16.35	10.00	11.11	10.00
Nucleus Sampling	<u>24.04</u>	<u>16.67</u>	<u>33.33</u>	<u>26.67</u>
BeamDiffusion	**59.61**	**73.33**	**55.56**	**63.33**

As shown in Table 1, BeamDiffusion consistently achieves the highest preference scores according to both human annotators (H) and LLM-as-a-judge (L). In particular, BeamDiffusion substantially outperforms Greedy decoding and Nucleus Sampling, indicating that beam-based decoding produces sequences that are not only more coherent but also better aligned with human and automatic judgments. While Nucleus Sampling improves over Greedy decoding by encouraging diversity, its performance remains notably below that of BeamDiffusion in both evaluation settings. The agreement between H and L further strengthens the evidence for the robustness of our decoding strategy. Given these results, we adopt BeamDiffusion as our default decoding method in subsequent experiments (see Appendix C.3 for qualitative and automatic evaluation results and Appendix B for hyperparameter configurations). Table 1 also compares two diffusion architectures: Stable Diffusion (SD2.1) and Diffusion in Transformers (FLUX-1-dev). BeamDuffusion exhibits consistent performance with both architecture, with Diffusion in Transformers outperforming SD2.1. Hence, in the following experiments we will use the Diffusion in Transformers architecture.

5.2 Comparison to Training-time Methods

Human Evaluation. Table 2 compares the sequence generation methods on both the Recipes and VIST datasets. BeamDiffusion dominates in both domains, leading Recipes with 64.52% (H) and 73.33% (L), and VIST with 41.10% (H) and 48.28% (L). StoryDiffusion consistently ranks second, with 22.58% (H) and 20.00% (L) in Recipes, and 36.99% (H) and 41.38% (L) in VIST. In contrast, 1P1S receives moderate preference (12.90% H in Recipes, 19.18% H in VIST), while GILL and StackedDiffusion record the lowest selection rates.

These results show that BeamDiffusion achieves

Table 2: Frequency of method selection by human annotators (H) and the LLM-as-a-judge (L).

Method	Recipes		VIST	
	H (%)	L (%)	H (%)	L (%)
GILL	0.00	0.00	2.74	0.00
StackedDiffusion	0.00	3.33	0.00	0.00
StoryDiffusion	22.58	20.00	36.99	41.38
1P1S	12.90	3.33	19.18	10.34
BeamDiffusion	**64.52**	**73.33**	**41.10**	**48.28**

the strongest performance across settings, with especially large margins in Recipes where BeamDiffusion surpasses the next-best method by over 40% for human preference. The consistently low selection rates of GILL and StackedDiffusion highlight that these methods are rarely preferred by either humans or LLM-as-a-judge, in contrast to the stronger performance of BeamDiffusion and StoryDiffusion. Overall, we conclude that search and selection strategies enable models to maintain narrative coherence and quality within complex domains like VIST.

Automatic Metrics. Table 3 compares all methods using metrics that measure sequence quality, including CLIP-I, DINO-I, CLIP-T, CLIP*, DINO*, and Goal Faithfulness (Menon et al., 2024), Step Faithfulness (Menon et al., 2024), and Cross-Image Consistency (Menon et al., 2024). BeamDiffusion achieves the highest combined scores (CLIP*: 40.00%, DINO*: 30.00%), reflecting strong performance in both visual and semantic consistency. This is further confirmed by faithfulness and consistency metrics, where BeamDiffusion leads in Goal Faithfulness (0.82), Step Faithfulness (0.92), and Cross-Image Consistency (0.61). StoryDiffusion performs well on CLIP-T (35.00%) and achieves high scores on the alignment metrics Goal Faithfulness: 0.78, Step Faithfulness: 0.87, Cross-Image Consistency: 0.61, showing strong semantic alignment and visual coherence despite lower combined metrics. 1P1S achieves the highest DINO-I (30.00%) but lower CLIP-T (10.00%) and moderate alignment scores, indicating strong visual consistency but weaker semantic alignment. StackedDiffusion maintains balanced performance across CLIP-I, DINO-I, and CLIP* (around 25.00%) with moderate alignment scores, while GILL consistently underperforms across both automatic and alignment metrics.

Overall, the relationships between CLIP-I/DINO-I and Cross-Image Consistency, and between CLIP-T and Goal/Step Faithfulness, show that the combined metrics CLIP* and DINO* reliably reflect both semantic and visual consistency. Together, these results demonstrate that BeamDiffusion is the most effective method for generating coherent and semantically faithful image sequences.

Out-of-domain results Besides the evaluation report previously, we evaluated BeamDiffusion on the DIY domain to demonstrate the generalization of our work. We generated 30 image sequences to illustrate home improvement tasks and evaluated it with Humans and an LLM as a judge. Results in Table 4 confirm that BeamDiffusion generalizes to a third domain, while out-performing the other baselines.

5.3 Inference-time Cost Analysis

BeamDiffusion is an inference-time-scaling method with a fixed and parametrized computational cost, which does not grow with the sequence length. This cost is fixed, independent of the sequence length, and is proportional to the upper bound on the number of memorized denoising paths. Hence, the impact of the computational cost on the method's performance is analyzed in Figure 3 as a function of the beam paths memory. We normalized the costs by the greedy-decoding method which uses 4 generations per sequence step (90 secs on an A100). We can observe that for a low computational cost, BeamDiffusion exhibits poor performance, reaching a maximum for a computational cost of 8 beam pahts. Past this point, the model exhibits diminishing returns: performance drops as compute increases. This is likely because extended look-back steps introduce an overly aggressive smoothing factor that dilutes the distinctions between distant sequence elements.

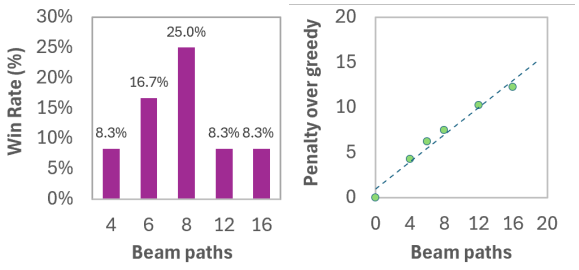


Figure 3: Inference-time cost and performance are directly influenced by beam diffusion search paths.

Inference-time-scaling cost to generate the considered beam paths, is what allows BeamDiffusion

Table 3: Comparison of different methods across embedding-based sequence quality metrics. CLIP-I, DINO-I, CLIP-T, CLIP*, and DINO* are reported as win percentages, while Goal Faithfulness, Step Faithfulness, and Cross-Image Consistency are reported as absolute scores.

Method	CLIP-I	DINO-I	CLIP-T	CLIP*	DINO*	Goal Faith.↑	Step Faith.↑	Cross-Image Cons.↑
GILL	0.00	0.00	0.00	0.00	0.00	0.67	0.52	0.54
StackedDiffusion	0.25	0.25	0.05	0.25	0.25	0.60	0.67	0.57
StoryDiffusion	0.20	0.25	0.35	0.25	0.20	0.78	0.87	**0.61**
1P1S	25.00	**30.00**	10.00	10.00	25.00	0.75	0.83	0.57
BeamDiffusion	**30.00**	20.00	**50.00**	**40.00**	**30.00**	**0.82**	**0.92**	**0.61**

Table 4: Placeholder Caption

Method	Humans (%)	LLM (%)
GILL [46]	0.0	20.0
StackedDiffusion [66]	12.5	15.6
Greedy / CoSeD [55]	0.0	23.3
Nucleus Sampling [67]	37.5	27.7
BeamDiffusion	50.0	30.0

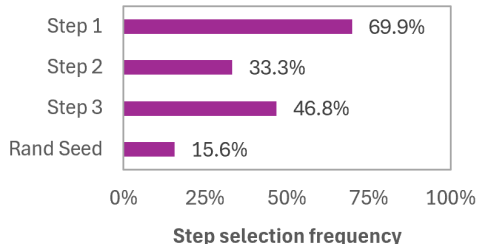


Figure 4: Non-linear steps selection, according to selection opportunities.

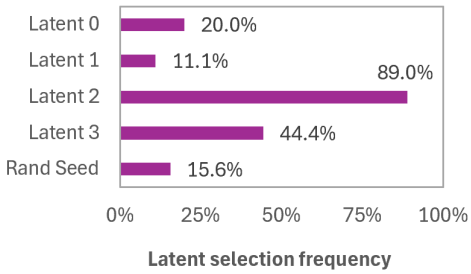


Figure 5: The average number of times that the model selected a given **latent** to generate the next image.

to avoid local maxima and steadily approach global maxima. In the initial steps, it has an early increase in computational complexity. The search space grows steadily until it stabilizes when the pruning strategy (0.012 sec) is applied to retain only the top candidates, which fixes the complexity on a fixed, parametrized, value. Detailed results in terms of steps back and width is provided in Appendix C and a theoretical analysis is presented in Appendix I.

5.4 BeamDiffusion Space Search Analyis

To analyze how BeamDiffusion explores the latent space, we analysed (1) the number of times that a sequence step provides latents for the generation of subsequent steps of the sequence, and (2) the number of times a step is selected to generate a future step. In this experiment we consider sequences of four steps.

Non-Linear Sequence Denoising Latents from step 1 were chosen 69.9% of the times they had a chance of being selected, highlighting the role of step 1 in shaping the sequence. Selection rates decrease for later steps, with Step 3 at 46.8%, Step 2 at 33.3%, and the random seed at 15.6%. Earlier steps, such as Step 1, have more chances of being chosen, leading to a higher selection rate. Thus, the percentages reflect how often each step is selected given the different possibilities for their inclusion, rather than a simple additive total. This pattern demonstrates how beam search prioritizes crucial steps, particularly Step 1, to establish sequence coherence while strategically weighing intermediate sequence steps.

Latent Selection Frequencies Following our evaluation of the impact of latent groups on performance, we now examine the role of individual latents within the higher-performing group (Latents 0-3). As shown in Figure 5, Latent 3 is selected most frequently (44.4%), followed by Latent 0 (20%), while Latents 1 and 2 are used less often (11.1% and 8.9%, respectively). This uneven distribution suggests that Latents 0 and 3 contribute disproportionately to the generation of sequences with better text alignment. One possible explanation is that these latents encode more semantically relevant or visually discriminative features that align well with the text prompts. Their frequent selection by beam search indicates that these latents are particularly effective at resolving ambiguities or



Figure 6: Example task from the Recipes domain (*"Crema Catalana with salted caramel sauce"*).

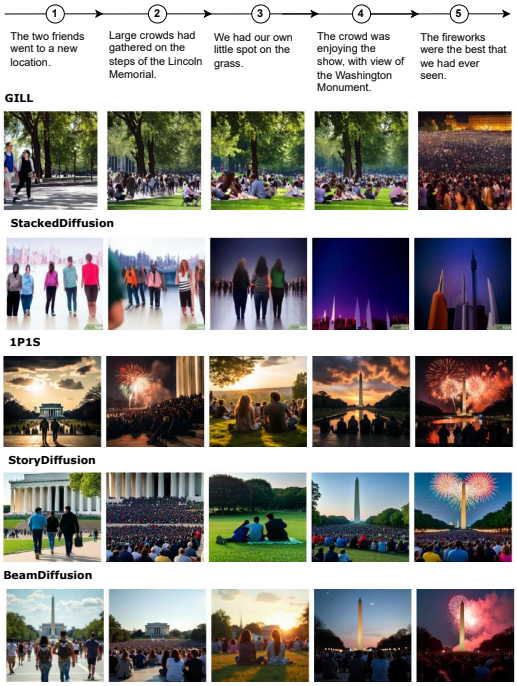


Figure 7: Example task from the VIST domain (*"Fireworks - 4th July in Washington, DC"*).

reinforcing important visual cues during sequence generation. The lower usage of Latents 1 and 2 may reflect their tendency to encode less salient or redundant information, which contributes less to text-conditioned coherence. Interestingly, the Rand seed, which injects stochasticity into the generation process, accounts for 15.6% of selections. This suggests that while beam search primarily exploits strong latents like 0 and 3, it also values diversity. The inclusion of randomized elements allows the search to escape local optima and sample from alternative yet plausible paths, enhancing the robustness of the final sequence.

5.5 Qualitative Analysis

To highlight the impact of our approach on semantic and visual consistency, we present two examples, one from the Recipes domain (Figure 6) and one from VIST (Figure 7). In the recipe example, BeamDiffusion avoids major semantic errors and maintains consistent object shapes across steps; for instance, while StoryDiffusion and 1P1S show inconsistencies in the shape of the tins between steps 3 and 4, BeamDiffusion preserves them accurately. In the VIST example, BeamDiffusion maintains both semantic alignment and visual coherence throughout the sequence, accurately depicting two friends and preserving consistent colors and lighting across all steps. In contrast, baselines such as 1P1S introduce a temporal inconsistency, shifting from sunset to night and back to sunset, while StoryDiffusion misinterprets the step text by showing three individuals instead of two friends.

6 Conclusions

In this paper, we proposed a novel method to explore the latent diffusion space with a beam search strategy, to generate image sequences. BeamDiffusion explores the latent denoising space and leverages this strategy to locate optimal denoising strategies. It is a theoretically sound framework, drawing on the strengths decoding techniques (Shi et al., 2024) and search methods, with clear experimental gains over alternative methods, as we demonstrated in generating image sequences with enhanced coherence and consistency.

Experimental results show that effectively balances textual alignment with visual continuity. While most methods align images with individual prompts, our results show that sequence decoding naturally improves alignment across an entire series.

Acknowledgments

We thank anonymous reviewers for their valuable comments and suggestions. This work was partially supported by the FCT project NOVA LINCS Ref. (UID/04516/2025) and by a Google Research Gift.

Limitations

This work shows that the latent diffusion space can be steered, offering more reliable and factual outputs. The current solution supports non-linear sequences, but does that on a maximum-likelihood framework. However, there are cases, where specific conditions should be enforced. Hence, a possible improvement would be the the exploration constraint decoding methods that assess the likelihood of each denoise latent with respect to the target prompt under a specific constraint. Solving this challenge, would require a non-linear optimization procedure.

Exploring the latent denoising space with BeamDiffusion requires Diffusion Models that provide access to fine-grain latents across denoising iterations, where BeamDiffusion still has the opportunity to steer the reverse diffusion process. The theoretical complexity of BeamDiffusion is linear with the number of beams that we configure the model to explore. The drawback of this strategy is that exploring the space of hypothesis increases the computational complexity of the process, discussed in Appendix G. However, this provides a solid ground to explore exciting new ideas where stronger constraints can be enforced in the explored beams.

Ethical Considerations

Dual-use risk. The method we propose can generate multiple visual scenes to illustrate a narrative. Hence, in terms of risks, we acknowledge that our work could potentially be used to generate false information.

Annotator welfare. Throughout the project, annotators maintained open communication with the authors. This included an initial discussion to clarify definitions and align on the guidelines. This collaborative approach was especially valuable because several annotators are active users of image generation methods

References

Amanda Bertsch, Alex Xie, Graham Neubig, and Matthew R. Gormley. 2023. It's MBR all the way down: Modern generation techniques through the lens of minimum bayes risk. *CoRR*, abs/2310.01387.

black-forest labs. 2024. flux: Official inference repo for flux.1 models. `https://github.com/black-forest-labs/flux`. GitHub repository.

João Bordalo, Vasco Ramos, Rodrigo Valerio, Diogo Glória-Silva, Yonatan Bitton, Michal Yarom, Idan Szpektor, and João Magalhães. 2024. Generating coherent sequences of visual illustrations for real-world manual tasks. pages 12777–12797.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Weifeng Chen, Jiacheng Zhang, Jie Wu, Hefeng Wu, Xuefeng Xiao, and Liang Lin. 2024. Id-aligner: Enhancing identity-preserving text-to-image generation with reward feedback learning. *arXiv preprint arXiv:2404.15449*.

Shaurya Dewan, Rushikesh Zawar, Prakanshul Saxena, Yingshan Chang, Andrew Luo, and Yonatan Bisk. 2024. Diffusionpid: Interpreting diffusion via partial information decomposition. *CoRR*, abs/2406.05191.

Joseph Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–.

Shaz Furniturewala, Kokil Jaidka, and Yashvardhan Sharma. 2024. Impact of decoding methods on human alignment of conversational llms. In *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, WASSA 2024, Bangkok, Thailand, August 15, 2024*, pages 273–279. Association for Computational Linguistics.

Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. 2024. Tokenflow: Consistent diffusion features for consistent video editing. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020a. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020b. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Chao-Chun Hsu, Szu-Min Chen, Ming-Hsun Hsieh, and Lun-Wei Ku. 2018. Using inter-sentence diverse beam search to reduce redundancy in visual storytelling. *arXiv preprint arXiv:1805.11867*.

Ting-Hao K. Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Aishwarya Agrawal, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, and 1 others. 2016. Visual storytelling. In *15th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Dragomir Radev, Yejin Choi, and Noah A Smith. 2024. A call for clarity in beam search: How it works and when it stops. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 77–90.

Lei Ke, Wenjie Pei, Ruiyu Li, Xiaoyong Shen, and Yu-Wing Tai. 2019. Reflective decoding network for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8888–8897.

Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. 2023. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation, NMT@ACL 2017, Vancouver, Canada, August 4, 2017*, pages 28–39. Association for Computational Linguistics.

Jing Yu Koh, Daniel Fried, and Russ Salakhutdinov. 2023. Generating images with multimodal language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Hezheng Lin, Xing Cheng, Xiangyu Wu, and Dong Shen. 2022. Cat: Cross attention in vision transformer. In *2022 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE.

Shaobo Lin, Kun Wang, Xingyu Zeng, and Rui Zhao. 2023. Explore the power of synthetic data on few-shot object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023 - Workshops, Vancouver, BC, Canada, June 17-24, 2023*, pages 638–647. IEEE.

Chang Liu, Haoning Wu, Yujie Zhong, Xiaoyun Zhang, Yanfeng Wang, and Weidi Xie. 2024a. Intelligent grimm - open-ended visual storytelling via latent diffusion models. In *CVPR*.

Tao Liu, Kai Wang, Senmao Li, Joost van de Weijer, Fahad Shahbaz Khan, Shiqi Yang, Yaxing Wang, Jian Yang, and Ming-Ming Cheng. 2025. One-prompt-one-story: Free-lunch consistent text-to-image generation using a single prompt. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. 2024b. Syncdreamer: Generating multiview-consistent images from a single-view image.

Yujie Lu, Pan Lu, Zhiyu Chen, Wanrong Zhu, Xin Wang, and William Yang Wang. 2024. Multimodal procedural planning via dual text-image prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 10931–10954. Association for Computational Linguistics.

Richard E. Mayer. 2002. Multimedia learning. volume 41 of *Psychology of Learning and Motivation*, pages 85–139. Academic Press.

Clara Meister, Gian Wiher, and Ryan Cotterell. 2022. On decoding strategies for neural text generators. *Trans. Assoc. Comput. Linguistics*, 10:997–1012.

Sachit Menon, Ishan Misra, and Rohit Girdhar. 2024. Generating illustrated instructions. In *CVPR*.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, and 7 others. 2024. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024.

Xichen Pan, Pengda Qin, Yuhong Li, Hui Xue, and Wenhu Chen. 2024. Synthesizing coherent story with auto-regressive latent diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2920–2930.

William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 4172–4182. IEEE.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2024. SDXL: improving latent diffusion models for high-resolution image synthesis.

Jordi Pont-Tuset, Jasper Uijlings, Soravit Changpinyo, Radu Soricut, and Vittorio Ferrari. 2020. Connecting vision and language with localized narratives. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 647–664. Springer.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International*

conference on machine learning, pages 8748–8763. PMLR.

Tanzila Rahman, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, Shweta Mahajan, and Leonid Sigal. 2023. Make-a-story: Visual memory conditioned consistent story generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2493–2502.

Vasco Ramos, Yonatan Bitton, Michal Yarom, Idan Szpektor, and Joao Magalhaes. 2025. Contrastive sequential-diffusion learning: Non-linear and multi-scene instructional video synthesis. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4645–4654.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, and 34 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389. The Association for Computational Linguistics.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, and 1 others. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494.

Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. 2017. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications.

Chufan Shi, Haoran Yang, Deng Cai, Zhisong Zhang, Yifan Wang, Yujiu Yang, and Wai Lam. 2024. A thorough examination of decoding methods in the era of llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 8601–8629. Association for Computational Linguistics.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. 2025. A general framework for inference-time scaling and steering of diffusion models. *CoRR*, abs/2501.06848.

Marko Smilevski, Ilija Lalkovski, and Gjorgji Madjarov. 2018. Stories for images-in-sequence by using visual and narrative components. In *ICT Innovations 2018. Engineering and Life Sciences: 10th International Conference, ICT Innovations 2018, Ohrid, Macedonia, September 17–19, 2018, Proceedings 10*, pages 148–159. Springer.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2021. Denoising diffusion implicit models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Tomás Soucek, Dima Damen, Michael Wray, Ivan Laptev, and Josef Sivic. 2024. Genhowto: Learning to generate actions and state transformations from instructional videos. In *CVPR*.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. 2023. What the DAAM: interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 5644–5659. Association for Computational Linguistics.

Ashwin Vijayakumar, Michael Cogswell, Ramprasaath Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zhongqi Wang, Jie Zhang, Shiguang Shan, and Xilin Chen. 2024. T2ishield: Defending against backdoors on text-to-image diffusion models. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXV*, volume 15143 of *Lecture Notes in Computer Science*, pages 107–124. Springer.

Sam Wiseman and Alexander M. Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP*

2016, Austin, Texas, USA, November 1-4, 2016, pages 1296–1306. The Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, and 12 others. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. 2024. Storydiffusion: Consistent self-attention for long-range image and video generation. *CoRR*, abs/2405.01434.

A Contextual Scene Descriptions

In many cases, step descriptions alone may lack context as precise visual prompts. These step descriptions are often interdependent, relying on prior steps for contextual understanding, or may miss visual details, e.g. *"Put the ice cream in freezer for 1 hour"*, or actions that involve multiple tasks, such as *"Slice the tomatoes, grate the cheese, and stir the sauce"*. To address this challenge, we leverage Gemini (Reid et al., 2024), specifically the Gemini 1.5 Flash model, to refine step descriptions into visually detailed prompts while considering sequence context. Specifically, Gemini produces a contextualized caption c_j based on the current step description s_j and all preceding steps as follows:

$$c_j = \phi(s_j | \{s_1, s_2, \ldots, s_{j-1}\}) \tag{6}$$

Following this approach we can improve image generation prompts by maintaining consistency and preserving the dependencies between sequential steps.

B Best Beam Search Configuration

The hyperparameters beam width (w in Eq. 5) and steps back (m in Eq. 2), play a crucial role in the performance of BeamDiffusion. In this section, we report an exaustive evaluation on how different beam widths (2, 3, 4, and 6) and steps back (2, 3, and 4) improve the latent selection and the overall quality of the generation. A more detailed discussion on the impact of the latents selection is provided in Appendix C.

We first perform human annotations on a set of sequences to identify the optimal beam search configuration, see Appendix E for details. To determine the best configuration, annotations were used to discard non-optimal configurations. The results in Table 5 showed that the best performing configuration was 2 steps back with a beam width of 4, achieving a selection rate of 25%. Other configurations, such as 3 steps back with a beam width of 2 and 4 steps back with a beam width of 2, also showed a strong performance (16.67%), but none outperformed the 2 steps back and beam width of 4 configuration. The configurations with beam width 6 performed the worst in all the settings tested.

To verify the reliability of our findings, the LLM-as-a-judge was applied to six tasks, yielding an 85% Fleiss' Kappa agreement (Fleiss, 1971), reflecting near-perfect alignment. The extended LLM-as-a-judge evaluation (Table 6) confirmed that the optimal configuration, 2 steps back with a beam width of 4, remained best, reaching a 48.57% selection rate compared to 33.33% in the initial six-task evaluation. Further, the configuration with 3 steps back and a beam width of 2 still performed well, yielding a 28.57% selection rate, confirming its robustness as an alternative. However, larger beam widths, such as beam width 6, continued to underperform, reinforcing that, in this task, a higher beam width does not always improve performance.

C Impact of Latent Selection on Beam Search Performance

Latent selection plays a crucial role in determining the quality and coherence of image sequences generated through beam search. By analyzing how different latent groups contribute to generation performance, we can better understand their impact on semantic alignment and visual coherence.

This aligns with recent work on interpreting diffusion models. For example, Pont-Tuset et al. (2020) ground textual descriptions in image regions, while Tang et al. (2023) and Dewan et al. (2024) explore how textual prompts influence generation through attention analysis and partial information decomposition. In contrast, our method contributes to interpretability by examining how latent space traversal, guided by beam search, affects the generation of coherent, semantically aligned image sequences.

To systematically investigate this, we evaluate several sequences, each consisting of four steps, allowing us to assess the effects of latent selection across a diverse set of sequences. Using both automatic evaluation and human annotations, we analyze different latent configurations to determine which groups contribute most to generating coherent and semantically meaningful sequences.

C.1 Non-Linear Sequence Denoising

Following the initial analysis of beam search configurations (Appendix B), where we used latent indexes 0, 1, 2, and 3, we further investigated the impact of using different latent configurations on the generated sequences. While latents 0, 1, 2, and 3 show promising results, latents 1, 3, 5, and 7 exhibit significantly poorer performance. In particular, beam search configurations for these underperforming latents result in very limited selection rates, indicating they are rarely chosen in both human and LLM-as-a-judge annotations. For human

Table 5: Influence of different hyperparameter values according to human annotations.

Steps Back	Beam Width			
	2	3	4	6
2	8.33	8.33	**25.00**	8.33
3	16.67	-	-	-
4	16.67	8.33	8.33	-

Table 6: Influence of different hyperparameter values according to LLM-as-a-judge annotations.

Steps Back	Beam Width			
	2	3	4	6
2	5.71	2.86	**48.57**	5.71
3	28.57	-	2.86	-
4	-	-	2.86	-

annotations across 6 tasks, the only configuration that yielded any selection was the one with 4 steps back and a beam width of 3, achieving a selection rate of 8.33%. All other configurations resulted in 0% selection. In contrast, none of the configurations showed any non-zero selection rate under LLM-as-a-judge annotations, reinforcing the lack of preference for these latents.

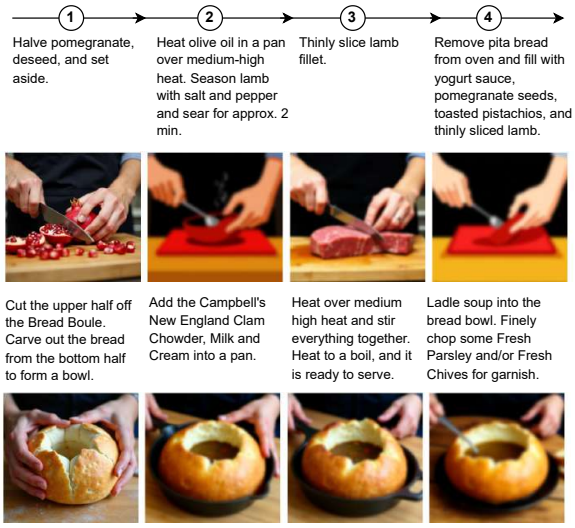


Figure 8: Impact of using late-index latents configurations on sequences generation.

Similarly, for LLM-as-a-judge annotations evaluated on the same 6 sequences, the configurations do not yield any notable results, with no selection rates. When the evaluation is extended to 35 sequences, the performance for latents 1, 3, 5, and 7 remains sparse, with a single configuration (2 steps back, beam width of 3) being selected, with a selection rate of 2.86%. Figure 8 visually demonstrates the poor results, where the images appear nearly identical, further supporting the findings of limited diversity and low performance, which aligns with the findings of (Ramos et al., 2025; Bordalo et al., 2024).

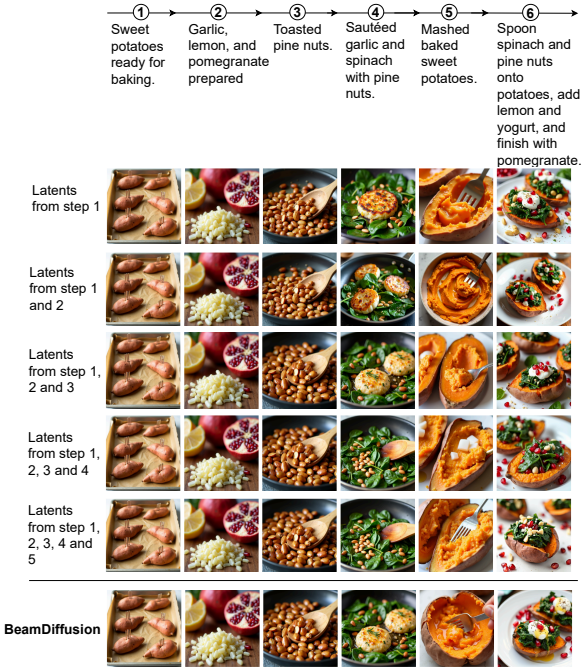


Figure 9: Visual comparison of sequences generated with increasing cumulative latent access. Each row shows a sequence generated using latents from only the first step, the first two steps, and so on, up to all previous steps. The bottom row shows the result from BeamDiffusion, which uses latents from only the two preceding steps.

C.2 Effect of Cumulative Latent Access

To better understand the role of latent reuse across generation steps, we study how access to latents from previous steps influences sequence quality. We conduct an ablation study in which we constrain the set of available latents throughout sequence generation, ranging from using only the latent from the first step, to progressively including latents from additional steps, up to full cumulative access to all previous steps. We further compare these settings with BeamDiffusion, which conditions only on the two most recent steps.

As shown in Figure 9, when the model has access only to the first-step latent, step 4 exhibits an

unnatural close-up. Expanding access to the first two steps alleviates this issue but makes it harder for the model to accurately follow the step-wise instructions while preserving overall coherence, an effect that is particularly evident in later steps such as step 5. Further increasing the latent history improves sequence quality, with the most pronounced gains observed when expanding access from two to three steps.

However, including latents from all previous steps does not yield further meaningful improvements in coherence or visual fidelity, indicating diminishing returns with increasing temporal access. In contrast, BeamDiffusion, which relies only on the two most recent latents, achieves results that are comparable to, or even superior to, the full cumulative setting while being substantially more efficient.

This behavior can be explained by the latent chain effect inherent to sequential generation. To generate a new image, the model conditions on a latent from the previous step, which already encodes information from earlier generations. The resulting latent is then updated with the new step content, effectively carrying forward both historical and newly introduced information (see Figure 2b). As a result, the most recent latents act as a compressed summary of the entire sequence history, making explicit access to all earlier latents redundant.

Overall, this experiment demonstrates that while cumulative latent access improves generation quality up to a point, full historical memory is unnecessary. Efficient strategies such as BeamDiffusion, which leverage a limited yet informative temporal window, can preserve global coherence while significantly reducing computational overhead. These findings further highlight the importance of the nonlinear sequence denoising strategy introduced in Section C.1, which enables effective reuse of past latents without requiring full-history conditioning.

C.3 Automatic and Qualitative Evaluation of Decoding Strategies

Automatic evaluation. Table 7 reports automatic comparisons across similarity metrics and faithfulness/consistency measures. The combined scores CLIP* and DINO* offer a more holistic signal than their individual components. Here, BeamDiffusion achieves the best CLIP* score (40.00) and matches the highest DINO* (30.00), indicating strong performance across both text–image and image–image similarity. By contrast, Greedy/CoSeD lags significantly on CLIP* (25.00) and only reaches 30.00 on DINO*, while Nucleus Sampling improves slightly on CLIP* (35.00) and DINO* (35.00) but at the expense of weaker instruction alignment and coherence.



Figure 10: Illustrative comparison of decoding approaches. BeamDiffusion prevents cascading inconsistencies across steps while preserving semantic and visual consistency.

BeamDiffusion also obtains the strongest scores on goal faithfulness (0.82) and step faithfulness (0.92), showing that its generations remain closely aligned with both the overall task and individual step instructions. Greedy/CoSeD achieves reasonably high step faithfulness (0.90) and the best cross-image consistency (0.62), but falls behind in goal alignment. Nucleus Sampling, in contrast, is weaker across all three measures (0.72, 0.80, 0.60). Taken together, the similarity metrics and the faithfulness/consistency measures highlight that BeamDiffusion avoids the sharp trade-offs of the other decoding methods, achieving a favorable balance between multimodal similarity, instruction following, and visual coherence.

Qualitative evaluation. Figures 10 and 11 illustrate characteristic error modes of baseline strategies. Nucleus Sampling often suffers from blur propagation, where low-frequency artifacts introduced in one step persist across subsequent generations, degrading the sequence. Greedy decoding avoids blur but instead produces visual inconsistencies, with objects abruptly shifting style or geome-

Table 7: Comparison of different decoding methods across embedding-based sequence quality metrics. CLIP-I, DINO-I, CLIP-T, CLIP*, and DINO* are reported as win percentages, while Goal Faithfulness, Step Faithfulness, and Cross-Image Consistency are reported as absolute scores.

Method	CLIP-I	DINO-I	CLIP-T	CLIP*	DINO*	Goal Faith.↑	Step Faith.↑	Cross-Image Cons.↑
Greedy/CoSeD$_{len=1}$	30.00	30.00	35.00	25.00	30.00	0.76	0.90	**0.62**
Nucleus Sampling	25.00	**40.00**	**40.00**	35.00	**35.00**	0.72	0.80	0.60
BeamDiffusion	**45.00**	30.00	25.00	**40.00**	**35.00**	**0.82**	**0.92**	0.61

try across steps (e.g., the changing appearance of the pan between step 1 and step 3, in Figure 10).

In contrast, BeamDiffusion avoids both pitfalls. By dynamically selecting informative latent trajectories, it prevents error accumulation while preserving smooth semantic and visual transitions. The generated sequences remain sharper, semantically faithful, and more coherent over time. This demonstrates that decoding is not merely a technical detail but a decisive factor in determining the fidelity and consistency of long-horizon generations.

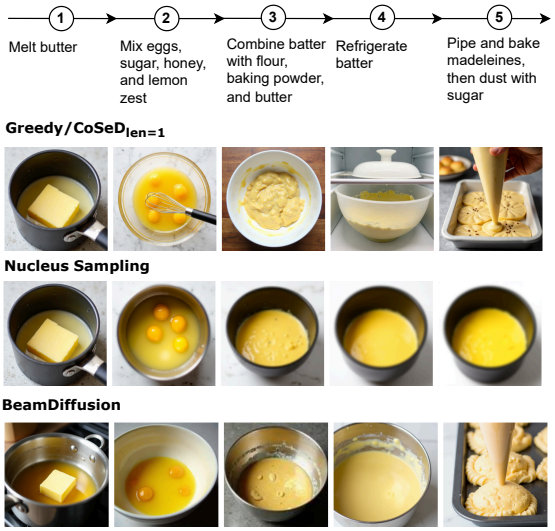


Figure 11: Illustrative comparison of decoding approaches. BeamDiffusion prevents cascading inconsistencies across steps while preserving semantic and visual consistency.

Summary. Overall, both automatic and qualitative results show that BeamDiffusion delivers a principled improvement over standard strategies. By explicitly managing latent trajectories, it achieves robustness against error propagation and inconsistency while preserving semantic faithfulness and cross-step coherence.

D Minimizing Decoding Risk

The generation process using diffusion models begins with a seed that determines the initial conditions of the model. The choice of this seed significantly affects the outcome, as it influences the path the model takes during generation. Using multiple random seeds in the first step ensures a diverse exploration of possible outputs, avoiding premature narrowing of options into a single path based on that seed. This limits the diversity of the results and increases the chance of missing better options.

Greedy Decoding and Nucleus Sampling. In both greedy decoding and nucleus sampling, we use the same approach in the first step. Multiple random seeds are employed to generate a set of candidate images. Each candidate is then evaluated using the CLIP score mechanism, which compares the generated images with the step text. The image that best aligns with the text, according to the CLIP score, is selected to continue the process. This method ensures that both greedy decoding and nucleus sampling benefit from multiple diverse starting points, increasing the quality and variety of the generated outputs.

BeamDiffusion. In BeamDiffusion, relying on a single random seed in the first step effectively makes the process a greedy selection, as it locks the generation into a specific path. This limits diversity and reduces the chances of discovering higher-quality outputs. By using multiple random seeds, each seed generates a different decoding trajectory, helping to broaden the search space and increasing the likelihood of finding better solutions. Using multiple random seeds in the initial step enhances the exploration of the latent space, ensuring better and more diverse results across different decoding strategies.

Random Seeds for mid-sequence Generation. The sequence of scenes $S = \{s_1, s_2, \ldots, s_L\}$ may have one scene in the middle of the sequence that should be independent. To address this, for steps $s_{j>1}$, we use a combination of random seeds and past latent vectors to maintain a balance between diversity and coherence, which is particularly im-



Figure 12: Illustration of the limitation of CLIP-I: despite a high CLIP-I score, the result is not semantically correct or visually faithful. The example compares several automatic metrics (CLIP-T, CLIP-I, DINO-I, CLIP*, DINO*), highlighting discrepancies and failure cases in sequence evaluation.

portant for such scenarios.

E Metrics and Annotations

Automatic Metrics. In image sequence generation, CLIP similarity scores between visually similar frames often exceed 0.85, even for near-duplicate content. Wang et al. (Wang et al., 2024) highlight that such high CLIP-I scores can indicate semantic duplication, which poses a challenge when diversity across frames is desired. While CLIP-I and DINO-I are common proxies for semantic similarity, rewarding near-duplicates undermines meaningful visual progression. Conversely, poor alignment between images and their textual prompts is reflected in low CLIP-T scores. Lin et al. (Lin et al., 2023) address this by clipping CLIP-T values below 0.1, effectively filtering out weak text-image matches.

To balance these factors, we set CLIP-I scores above 0.9 and DINO-I scores above 0.85 to zero, penalizing visual redundancy. Similarly, we zero out CLIP-T scores below 0.1 to discourage poor alignment with the prompt. This clipping strategy encourages the generation of sequences that are both semantically relevant and visually diverse. In Figure 12, we present the automatic metric scores for a sequence across all methods. The results illustrate that very high CLIP-I values can lead to sequences that fail to follow the textual steps accurately, as observed in the case of GILL. Despite achieving strong visual similarity, the sequence diverges from the intended narrative, highlighting a key limitation of relying solely on CLIP-I . This example reinforces the importance of our clipping strategy: without penalizing excessively high CLIP-I values, the generation process may favor near-duplicates over meaningful visual progression, ultimately undermining semantic fidelity to the prompt.

Even after applying this clipping, we observe that automatic metrics remain imperfectly aligned with human and LLM-as-a-judge evaluations, underscoring the ongoing challenge of developing reliable metrics for evaluating sequences of images.

Human annotations. To facilitate the selection of the best beam search configuration, we developed a custom website specifically for human annotation. We found that existing tools lacked the flexibility necessary for our specific use case, especially when evaluating multiple configurations simultaneously. To select the best beam search configuration, 5 annotators are presented with 24 different sequences, each corresponding to a unique beam search configuration.

In total, the 24 configurations were generated by combining three key factors:

- **Steps Back** (m in Eq. 2): This refers to how many previous steps the model uses from its latent representations to explore new candidate outputs. We evaluated three different settings for the number of steps back: 2, 3, and 4, which correspond to using latents from the last two, three, or four images, respectively.

- **Latents Indexes**: We tested two sets of latent configurations, 1, 3, 5, 7 and 0, 1, 2, 3, by selecting specific indexes from the latent representations generated in previous stages. This was done to explore how varying levels of latent detail impact the quality of the generated sequences.

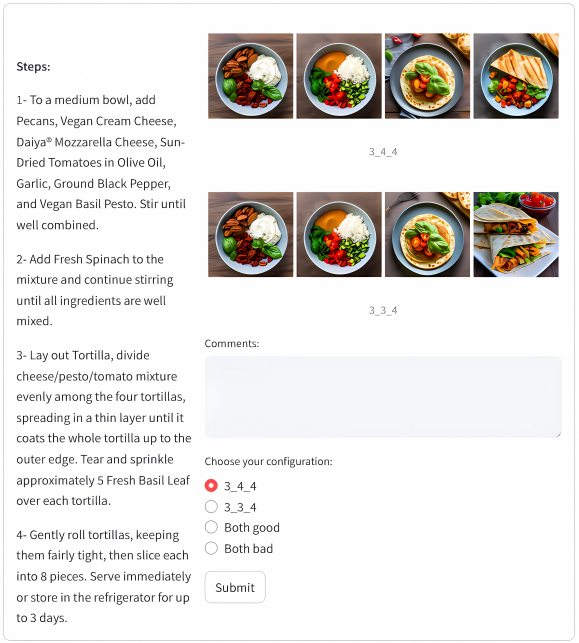


Figure 13: Example of the human annotation elimination round for selecting the best beam search configuration.

- **Beam Width** (w in Eq. 5): We explored four different beam widths, 2, 3, 4 and 6, which determine how many beams are retained at each step during the search process.

Evaluating all of these at once proved to be exhausting and difficult to manage. To simplify the evaluation process, we adopted a elimination annotation method. In this approach, 5 annotators compare two configurations at a time and select the one with better overall coherence. If neither configuration stands out, annotators can select *"both good"* or *"both bad"* (Figure 13), in which case the computationally cheaper option advances to the next round. This process is repeated until we identify the best configuration. This method focuses on direct comparisons, making it easier to pinpoint the most effective beam search settings while also considering computational efficiency.

To evaluate all models performance, annotators are presented with five generated sequences side by side. Figure 14 illustrates an example of this comparison process, where the annotators assess the sequences to determine the most effective approach.

LLM-as-a-judge Evaluation. The LLM-as-a-judge selection process for annotating the best beam search configuration and generation method closely mirrored that of human annotators, with

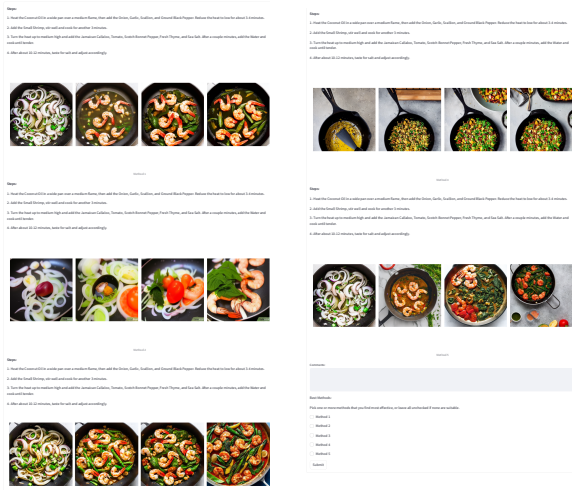


Figure 14: Example of a human annotation to choose the best method.

LLM-as-a-judge Prompt:
Evaluate two image sequences using the following criteria:

- **Visual Consistency**: Objects, ingredients, backgrounds, colors, and styles should remain consistent across frames. Frames should transition smoothly, resembling a video-like progression with no sudden changes or missing elements.
- **Semantic Consistency**: Images must logically follow the described steps, clearly and accurately.

Steps of the process: {steps}
After evaluating both sequences, select the one with the strongest overall consistency (visual and semantic). Pay close attention to even minimal differences, as the sequences are very similar. If both sequences are equally good or bad, choose the one that is slightly better. Choose exactly one of the following options:
- "First sequence"
- "Second sequence"

Figure 15: LLM-as-a-judge prompt used to select the best beam search configuration and best method.

one key adjustment: the model tended to default to *"both good"* or *"both bad"* when such options were available. To encourage more decisive outcomes, we removed these options from the prompt. We also refined our comparison methodology. Presenting all five sequences simultaneously initially led to biased evaluations, as the model disproportionately favored the last two. To address this, we adopted a pairwise evaluation strategy, comparing two sequences at a time. This adjustment yielded more balanced and reliable assessments across all comparisons.

Figure 15 presents the LLM-as-a-judge prompt for evaluating image sequences on visual and semantic consistency.

F Datasets

Recipes. Each manual task in the Recipes datasets includes a title, a description, a list of ingredients, resources and tools, and a sequence of step-by-step instructions, which may or may not be illustrated. The Recipes dataset comprises approximately 1,400 tasks, with an average of 4.9 steps per task, totaling 6,860 individual steps. Most tasks feature an image for each step.

VIST. We used the Descriptions of Images-in-Isolation (DII) annotations, as they allow for the generation of more precise and informative visual prompts. These annotations provide standalone descriptions of images, making it easier to capture their key visual elements without relying on surrounding context. The dataset includes 50 000 stories, offering a diverse range of visual and textual information. To ensure the most accurate and relevant annotation for each image, we leveraged CLIP. By using CLIP, we were able to select the annotation that best represented the image, improving the quality of our generated prompts and enhancing the overall alignment between text and visuals.

G Challenges

In the development and evaluation of the BeamDiffusion method, we encountered several challenges that required careful consideration and innovation. These challenges spanned multiple aspects of the model, ranging from the difficulty of generating accurate and concise captions, to identifying suitable domains for testing, and addressing the computational complexity of the generation process. While we were able to propose solutions for some of these challenges, others remain open and present ongoing areas of research. In this section, we outline the key challenges we faced and the strategies we employed, while acknowledging that not all of them have been fully addressed.

Captions. One of the challenges encountered in this process is generating captions that accurately describe the steps involved in the sequence. Often, a single step may consist of multiple sub-steps, each contributing to the overall generation. This makes it difficult to write a single concise caption that captures the full scope of each step. The challenge lies in ensuring that the caption not only reflects the specific action or transformation occurring at each step but also conveys the progression and complexity of the entire sequence. Additionally, if we divide the steps into smaller sub-steps to better describe the process, the sequence becomes very extensive, making it even harder to maintain clarity and readability while ensuring comprehensive coverage of the entire generation process.

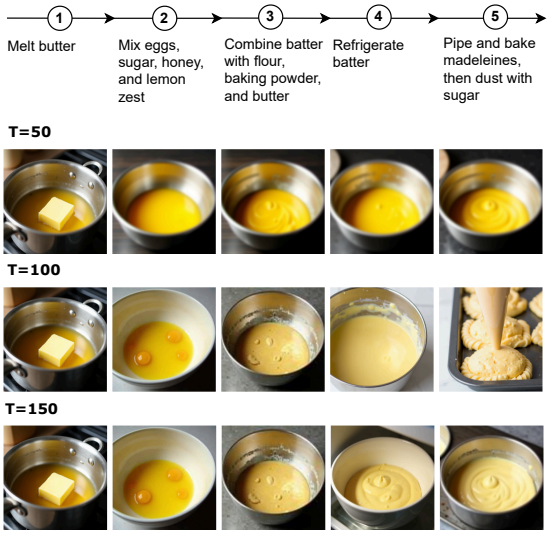
Domains. Identifying a domain that effectively highlights the advantages of the BeamDiffusion method, particularly the benefits of beam search, is still a challenge. While BeamDiffusion provides improvements in consistency and coherence, not all domains clearly showcase these advantages. Certain domains may lack the complexity or diversity needed to demonstrate the method's strengths, such as its adaptive latent selection or its ability to maintain alignment with contextual instructions. Finding a domain that not only aligns with the method's capabilities but also provides a clear and compelling demonstration of its improvements remains an ongoing challenge.

H Analytical Analysis of Inference-time Scaling Cost

The BeamDiffusion model generates multiple candidate images at each step by exploring a range of latents. We analyze the complexity of the model for a steps back of m steps and N latents explored. In the initial steps, the complexity is high because pruning begins only after the second step. For each beam, $N \cdot m$ latents are explored, and the search space grows exponentially, resulting in significant computational cost in the early stages. During the first two steps, the complexity grows with the number of candidates and latents, proportional to $\mathcal{O}(N \cdot m)^j)$ for $j \leqslant 2$. Specifically, for $j = 1$, the complexity is proportional to the number of explored random seeds. After the second step, pruning reduces the number of candidates to the most promising w beams, decreasing the complexity to $\mathcal{O}(N \cdot m \cdot w)$.

To further reduce complexity, a heuristic based on text relevance can be used to predict the top P most relevant past images. By restricting the search space to the latents of these top P candidates, the model improves efficiency by prioritizing contextually relevant images, particularly in the early steps. While the initial three steps remain computationally demanding due to the expansive search space, pruning combined with this heuristic can significantly lower the computational cost by refining the candidate set.

I Implementation Details



(a) Example 1: 50-step generations exhibit blur propagation, while 100 steps reduce blur and sharpen frames. 150 steps show no substantial gain over 100.



(b) Example 2: 50 steps produce blurry, inconsistent frames, while 100 steps improve fidelity and coherence. 150 steps do not yield further noticeable improvements.

Figure 16: Qualitative comparison of 50 vs. 100 denoising steps across two examples. Increasing steps to 150 does not produce substantial gains over 100, indicating diminishing returns.

Base Models. Our method is primarily implemented on top of FLUX.1-dev(black-forest labs, 2024), a diffusion transformer–based model released by Black Forest Labs. We extend its open-source implementation to support beam-based generation, latent trajectory tracking, and contrastive scoring mechanisms as described in the main text. Unless otherwise specified, we follow the default configuration of FLUX.1-dev with 100 denoising steps, chosen to balance generation quality and computational efficiency. While the default 50 steps generally produce globally coherent images, we observe that finer structures often remain underdeveloped, resulting in blur and reduced visual consistency across sequences. Increasing to 100 steps substantially sharpens details and improves sequence coherence, as illustrated in Figures 16a and 16b. Further increasing to 150 steps provides no additional gains, confirming that 100 steps is a reasonable trade-off. To ensure that the choice of decoding method is not dependent on a specific backbone, we also implement BeamDiffusion on Stable Diffusion v2.1(Rombach et al., 2022), a widely used U-Net–based latent diffusion model, using its standard parameters. This setup verifies that our conclusions regarding decoding strategies generalize across different generative architectures.

Latent Trajectory Reuse. In our approach, maintaining coherence across sequential generations requires access to the intermediate latent representations produced during the generative process. Fortunately, the model pipeline inherently provides the full sequence of latent representations at each generation step. This enables us to easily extract and reuse specific latents as seed for new generations.

Classifier Training. To train the contextual classifier, we employ a sequence-aware training approach that enables the model to capture dependencies between steps in the generation process. The contrastive classifier φ is trained as a *next-image prediction* task, estimating how well a candidate image aligns with the previously generated sequence of steps and images. The classifier is primarily trained on the Recipes dataset, which provides rich sequential image–text pairs of cooking steps, allowing it to learn meaningful context transitions in a structured procedural domain. Additionally, a small amount of data from the VIST dataset is included during training to introduce limited variability in narrative structure.

The classifier takes as input a candidate image I_l, its textual description s_l, and the previous sequence images and respective step text ($B_{l-1}^{(b)}$), and outputs a score reflecting how well I_l fits within the evolving semantic and visual trajectory of the sequence. During training, we use labels $l_{d,l}$ indicating whether I_l is the correct continuation for task d at step l. Negative examples are constructed by

sampling images from other sequences, resulting in a contrastive learning setup.

The model is trained to minimize the cross-entropy loss:

$$\text{argmin} \sum_{d=1}^{D} \sum_{l=1}^{L} l_{d,l} \cdot \log(\varphi(I_l, s_l, B_{l-1}^{(b)})), \quad (7)$$

which encourages the classifier to assign high scores to contextually appropriate continuations while penalizing incoherent or semantically unrelated ones. As a result, the classifier learns to distinguish subtle differences in step-to-step coherence, both visually and textually.

Hardware. All experiments were conducted on a NVIDIA A100 GPU with 40GB of memory.

J Broader Impacts

This work proposes a method to improve generative pretraining by incorporating subject-aware mutual information maximization, aiming to produce more contextually aligned and coherent sequences of images. The ability to generate visually consistent image sequences has positive applications in areas such as animation, visual storytelling, educational media, and accessibility tools (e.g., for generating illustrative content from text). However, as with many generative models, there are potential negative societal impacts. Enhanced coherence and subject fidelity could be misused to create misleading or deceptive visual content, such as more convincing visual disinformation, propaganda, or synthetic media (e.g., deepfakes).

K Application Examples

This section presents additional visual examples and qualitative results that complement the main paper. Images and other visual aids are essential tools that significantly improve comprehension across a variety of domains. Whether breaking down complex tasks, illustrating a sequence of events, or clarifying intricate concepts, visual representations provide clear, step-by-step guidance that enhances understanding (Mayer, 2002). From technical procedures and instructional design to storytelling and educational content, multi-scene image sequences help readers grasp information more intuitively, transforming even the most complex or dense topics into something more accessible. However, generating a sequence of images that not only aligns with each textual instruction but also maintains overall coherence remains a major challenge (Lu et al., 2024). In this appendix, we include expanded examples to further illustrate how our model addresses this issue. These include additional visualizations of the beam search process and extended qualitative comparisons across domains such as recipes and visual storytelling. Together, these examples highlight the strengths and limitations of current approaches in achieving textual alignment and visual consistency in sequential image generation.

K.1 Sequence Examples

To provide a comprehensive comparison between all models, we present several examples across different domains in Figures 17-20, highlighting how each method maintains sequence coherence in sequential image generation.

For the Recipes domain (*"Buttermilk chicken wings"*, Figure 17), most models incorrectly depict the chicken as already cooked in step 2, instead of starting raw and gradually appearing cooked. BeamDiffusion correctly captures this progression, faithfully reflecting the cooking process across all steps. Similarly, in the Recipes task *"Sweet potato and orange soup"* (Figure 18), GILL, StackedDiffusion, 1P1S, and StoryDiffusion misrepresent the ingredient being cut, while BeamDiffusion consistently depicts a person cutting a sweet potato and maintains stepwise visual consistency without hallucinations.

For the VIST stories, BeamDiffusion demonstrates strong identity and style preservation. In *"4th of July 2006"* (Figure 19), it is the only method to maintain the same child between step 2 and step 4. In *"Yosemite Weekend"* (Figure 20), both BeamDiffusion variants maintain coherent tree styles across steps, whereas other models show a noticeable break between step 1 and step 2.

Overall, while baseline methods may partially adhere to textual instructions or maintain visual similarity, BeamDiffusion consistently balances stepwise textual alignment and visual coherence, producing sequences that are both contextually accurate and visually stabl



Figure 17: Example of the Recipes domain task *"Buttermilk chicken wings"*, with a sequence of 4 steps.



Figure 18: Example of the Recipes domain task *"Sweet potato and orange soup"*, with a sequence of 4 steps.



Figure 19: Example of the VIST story domain *"4th of July 2006"*, with a sequence of 5 steps.

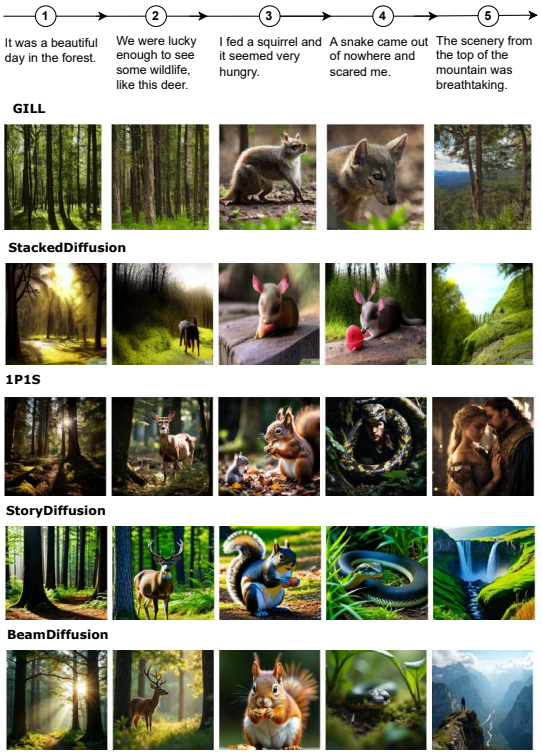


Figure 20: Example of the VIST story domain *"Yosemite Weekend"*, with a sequence of 5 steps.