Guiding Diffusion Models for Spatially Consistent Image Generation

Master Thesis

Violeta Chatalbasheva

Delft University of Technology

TU Delft

Guiding Diffusion Models for Spatially Consistent Image Generation

by

Violeta Chatalbasheva

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday June 20, 2025 at 4:00 PM.

Student number:	5080428	
Project duration:	August 8, 2024 – June 20, 2025	
Thesis committee:	Dr. ir. H. Jamali-Rad,	TU Delft, Thesis Supervisor
	Dr. ir. E. Isufi,	TU Delft, Advisor
	Dr. ir. H. Caesar,	TU Delft, External Committee Member
	Dr. ir. H. Palangi,	Google AI, External Co-supervisor
	Ir. S. Rastegar,	TU Delft, Daily Co-supervisor

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Preface

The thesis report ”Guiding Diffusion Models for Spatially Consistent Image Generation” presents the work that I have done in the past 10 month of my master’s degree, an experience that turned out to be far more enlightening than I had ever imagined. This research was conducted within the Computer Vision Lab at TU Delft, under the supervision of Dr. E. Isufi and the daily supervision of Dr. H. Jamali-Rad.

I would like to wholeheartedly thank Dr. Hadi Jamali-Rad who has given me the opportunity to work alongside brilliant minds, to learn from his experience and to constantly strive for excellence in my work. I was inspired by every conversation we had and I have tried to emulate many of your impressive and admirable qualities - how you bring the team together if there is misunderstanding and the ability to ignite enthusiasm when talking about work. I truly appreciate everything you have done for me. I want to thank Dr. Elvin Isufi for the effort he puts in designing courses so one can learn through both theory and practice. I would also like to thank Dr. Holger Caesar for his interest in evaluating my work as part of my thesis defense committee. I would further like to thank Dr. Hamid Palangi who was very patient with me and was constantly offering the highest level of professional advice possible. I would like to thank Ir. Sarah Rastegar who provided me not only with exceptional professional guidance but also became a very dear friend to me. Last but not least, I would like to thank the whole team for supporting me throughout this thesis journey.

For the past six years I have been through so many experiences which have shaped me and made me want to be the best version of myself every day. I owe my successes to my hard work and to my stubbornness in never giving up when facing hardships. But most of all, I owe all the good things, that have happened to me, to my parents. They have always believed in me, supported me in every possible way and have given me the opportunity to study abroad so that I can have a better life and build a career I can be proud of. I would also like to thank my amazing friends who care about me and have motivated me to work even harder. You made my student experience more enjoyable and very much unforgettable.

Violeta Chatalbasheva
Delft, June 2025

Contents

1 Introduction

Generative Artificial Intelligence (GenAI) has been one of the most influential technological breakthroughs of the past few years. The moment it attracted everybody's attention was the rise of ChatGPT that OpenAI released immediately after the COVID-19 pandemic started wearing away [41]. It took over the world by storm. The novelty of this technology was its ability to understand and respond, in natural language, to any question a user might ask. ChatGPT could compose rhymes, write jokes, assist with daily tasks - it felt like having your own personal assistant. Soon, it could reason better, perform correct math calculations, crawl the internet to find the most relevant information and even help programmers with code generation [42]. The next leap came with introducing multimodality in GPT-4V [43], which extended generative capabilities beyond text, enabling image generation, speech processing and even video understanding. Competitors quickly joined the AI race. Google released Gemini [11], Anthropic developed Claude [3], Microsoft partnered with OpenAI to launch Copilot [37], while Mistral [2] and most recently DeepSeek [1]. These models are revolutionizing the way we work, create and access information - from automating workflows, transforming content creation on platforms like social media and entertainment to speeding up technological breakthroughs. The ability to generate realistic images from text prompts is the driving force behind the topic of this thesis.

GenAI has trasformed the way data can be synthesized enabling the creation of highly realistic images. Generative modeling is about algorithms that can capture complex probability distributions, producing synthetic data that can easily be confused with real-world examples. The growing popularity and applicability of generative models can be attributed to significant breakthroughs in deep learning techniques [30], improvements in computational resources (e.g., GPUs and TPUs) [26] and the availability of large-scale datasets such as LAION-5B [56].

Generative models include Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs) and autoregressive transformers. However, among these, diffusion-based models began dominating particularly because of their ability for high-fidelity image synthesis. Diffusion models operate through a forward process that gradually adds noise to an image and a reverse process that learns to denoise and reconstruct the original image. Notable diffusion-based models are DDPM [21], Score SDE [57], Stable Diffusion [53], Imagen [54] and GLIDE [38], setting new benchmarks in image generation based on conditional information.

Diffusion models work by iteratively refining an initial noise distribution into a meaningful visual representation. The generation process can be guided by conditional information, such as text, bounding boxes, etc., making them suitable for tasks like text-to-image generation, inpainting and image editing.

Despite their impressive capabilities [14], some challenges remain. For instance, Stable Diffusion models [53] struggle with depicting accurately the spatial relationships within generated images. Spatial understanding in diffusion models refers to their ability to generate images that respect the spatial relationships given in the text prompt. However, multiple studies show that Text-to-Image (T2I) diffusion models produce images with incorrect placement of objects or mixing objects in one [8, 16]. Such

inaccuracies limit the practical applications of these models in areas requiring high spatial precision, such as robotic perception [9].

Recent benchmarks such as T2I-CompBench [24] and MI-TUNE [63] have quantitatively confirmed these limitations, describing a clear research gap and highlighting the need for methods that can improve spatial alignment without high computational overhead. While fine-tuning-based approaches have demonstrated potential in enhancing spatial accuracy, they require substantial retraining on annotated datasets, making them computationally intensive, less adaptable, and susceptible to dataset bias. Conversely, inference-time guidance methods aim to dynamically influence the generation process without retraining. The related works in this category mainly address missing object [8] and attribute bindings [8, 15, 64] and some that use extra inputs, such as bounding boxes, blobs, segmentation maps, to achieve spatial cognizant images [10, 44] but there are few papers that precisely target improving spatial alignment of T2I generation models [7, 18].

To bridge these critical gaps, this thesis introduces `InfSplign`, a novel inference-time spatial alignment technique explicitly designed to enhance spatial accuracy in T2I diffusion models. `InfSplign` leverages sophisticated spatial loss functions based on attention map analysis and centroid calculations, allowing for dynamic guidance of image generation towards accurate spatial layouts without the burdens associated with retraining.

The primary objectives of this thesis are:

- To analyze and identify critical limitations in current generative models regarding spatial accuracy.
- To propose, implement, and rigorously evaluate an innovative inference-time spatial alignment method.
- To empirically demonstrate the performance and effectiveness of `InfSplign` against established benchmarks and existing state-of-the-art approaches.

The structure of this thesis is as follows. Chapter 2 presents the scientific article detailing the motivations, methodology, and experimental outcomes of `InfSplign`. Chapter 3 provides an extensive background on generative AI with a particular focus on diffusion models and Vision-Language Models (VLMs). Chapter 4 discusses existing inference-time guidance methods and contextualizes `InfSplign` within the broader field of spatial understanding. Chapter 5 offers additional insights, analyses, and discussions around the results and potential improvements. Finally, Chapter 6 concludes the thesis, summarizing key contributions and outlining future research directions.

2

Scientific Article

InfSplign: Inference-Time Spatial Alignment of Diffusion Models

Violeta Chatalbasheva[1] **Sarah Rastegar**[1] **Sieger Falkena**[2]
Anuj Singh[2] **Yanbo Wang**[1] **Hamid Palangi**[3] **Hadi Jamali-Rad**[1,2]

[1]Delft University of Technology, The Netherlands,
[2]Shell Global Solutions International B.V., Amsterdam, The Netherlands,
[3]Google Research

Abstract

Text-to-image (T2I) diffusion models have achieved remarkable image quality but still struggle to produce images that align with the compositional information from the input text prompt, especially when it comes to spatial cues. We attribute this limitation to two key factors: the lack of clear fine-grained spatial supervision in common training datasets, and the inability of the CLIP text encoder, used in the pretraining of stable diffusion models, to represent spatial semantics. While recent work has addressed object omission and attribute mismatches, accurately generating objects in the spatial locations defined in the text prompt remains an open challenge. Prior solutions typically rely on fine-tuning, which introduces computational overhead and risks degrading the pretrained model's generative prior on other tasks unrelated to spatial reasoning. In this paper, we introduce `InfSplign`, a simple and training-free method that improves spatial understanding in T2I diffusion models. `InfSplign` leverages attention maps and a centroid-based loss to guide object placement during sampling at inference time without modifying the pretrained model. Our approach is modular, lightweight and compatible with any pretrained diffusion model. `InfSplign` achieves strong performance on spatial benchmarks such as VISOR, T2I-CompBench and GenEval, outperforming baselines in many scenarios.

1 Introduction

Diffusion-based text-to-image generative models have rapidly advanced, enabling the synthesis of high-quality, detailed images from arbitrary textual descriptions [1–8]. Despite these developments, precise control over spatial relationships described in textual prompts remains challenging. A particularly glaring limitation is the inadequate spatial understanding, manifesting frequently as misplacement or unintended merging of objects in generated images [9, 10]. For example, diffusion models frequently fail to distinguish between prompts such as *"object A to the left of object B"* and *"object A to the right of object B"*, often producing nearly identical outputs irrespective of spatial cues as shown in Figure 1. Such misalignments substantially reduce reliability, hindering applications that demand accurate spatial reasoning, such as generating scene layouts for robotic manipulation and visual grounding in augmented reality systems [11].

This deficiency in spatial understanding is quantitatively evident. Recent works, such as MI-TUNE [10], highlight that Text-to-Image (T2I) models exhibit notable weaknesses in spatial reasoning tasks, which are evaluated as part of compositional benchmarks (T2I-CompBench [12]). Specifically, the spatial accuracy (around 20%) is significantly lagging behind aspects like attribute binding (close

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

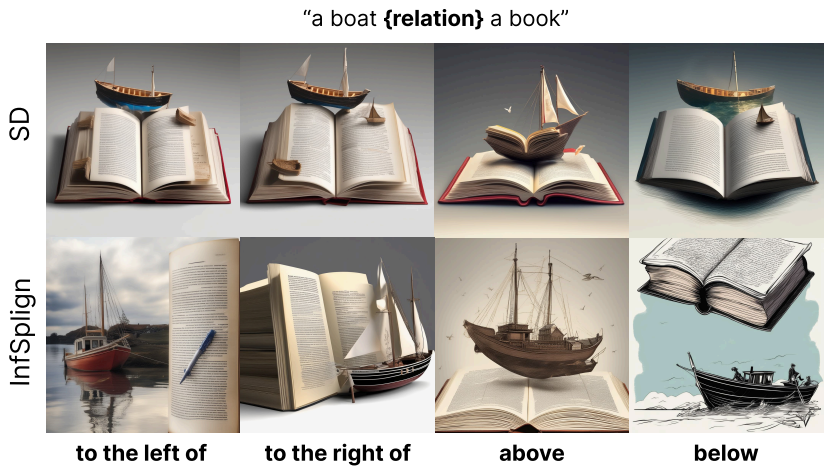


Figure 1: Comparison of Spatial Awareness of the Proposed Method vs Stable Diffusion (SD). Each column corresponds to a different spatial relationship (e.g., "to the left of," "to the right of," "above," "below"), indicated by position in the prompt. Stable Diffusion (SDXL [7]) fails to consistently reflect these spatial cues, often producing nearly identical images across spatial variants. In contrast, our method `InfSplign` enforces spatial correctness and enables diverse and compositionally valid generations even for uncommon object configurations.

to and above 50%) [10]. This performance gap underscores a critical area within T2I research, emphasizing the importance of developing solutions that effectively address spatial misalignment.

The main approaches tackling spatial accuracy broadly fall into two categories: fine-tuning-based and inference-time-based methods. Fine-tuning-based methods typically employ spatially-aware datasets, auxiliary reward models, or explicit training mechanisms to enforce spatial accuracy [9, 13], achieving relatively high spatial accuracy at the cost of considerable computational overhead and the risk of negatively impacting the carefully optimized diffusion backbone and its generalizability. In contrast, inference-time methods [14, 15] avoid expensive re-training, providing computationally efficient alternatives capable of flexible spatial adjustments during sampling. Despite their attractive efficiency, current inference-time methods often remain overly complex, relying on auxiliary inputs like layout maps [16], scene graphs [17], or external guidance from large language models (LLMs) [18, 19], thereby limiting ease of deployment and interoperability.

A natural hypothesis might be that the spatial limitations arise from the CLIP text encoder [20], commonly used in pretrained diffusion models, that fails to adequately encode spatial semantics [21]. However, our preliminary investigations indicate deeper issues. We undertook a simple contrastive guidance approach in which we extended the classifier-free guidance (CFG) formulation, following the insights gathered in [22]. The idea was to encourage the model to generate spatial layouts that align with the relationship in the prompt, e.g. "a cat *to the left of* a dog", while contrasting it to others, e.g. "a cat *to the right of / above / below* a dog". However, this contrastive signal did not lead to any significant improvements, suggesting that simply textually encoding spatial relationships does not seem to provide any meaningful guidance.

Another line of reasoning for whether inference-time guidance can achieve the task of generating images aligning with spatial information from the prompts is the amount of images it has seen during training in which one object is on the left or right of another. There is a lack of spatial supervision present in the training set of Stable Diffusion (SD) models. For example, SPRIGHT [23] re-captions the images in four of the most widely used ML datasets to provide clearer instructions to models when generating images that need to adhere to spatial information. One of the concerns is that if there are not enough examples in the prior distribution of the training data that have one object on the left of another one, guiding the model towards generation of these two objects with an accurate spatial relationship between them might not lead to an accurate result, which is the strongest motivation for using a fine-tuning approach. But since there is no evidence and fine-tuning comes with a lot of challenges, the topic of our work is to uncover meaningful guidance information that can influence

the generation towards image configurations that align with the given spatial relationships, assuming the model's prior distribution has implicitly learned to represent such compositional structures.

Furthermore, extensive ablation studies, including reversing spatial relationships or swapping object orders [9], consistently demonstrate poor alignment between generated images and the spatial semantics of prompts. Additionally, [23] reports further limitations in the semantic understanding of CLIP text encoders in an experiment where they randomly select captions, used for fine-tuning the diffusion model, and replace them by negating the spatial relationships ("a man is not to the left of a dog" substitutes "a man is to the right of a dog"). As a result, the model performs even worse when evaluated on prompts containing only negation. These findings collectively suggest that the issue lies not merely in the textual embeddings but rather within the diffusion generation process itself, particularly in how spatial information implicitly encoded in attention maps fails to propagate correctly during inference.

This has motivated us to pursue an alternative strategy. Instead of introducing additional training complexity or external knowledge bases, we explore a simpler, computationally efficient approach leveraging information already present within the diffusion process itself. Specifically, we focus on directly extracting spatial information from attention maps during the early stages of the reverse diffusion (generation) process and subsequently guiding the sampling process toward more spatially cognizant generation. While prior works fail to exploit these representations meaningfully, we show that attention maps can serve as proxies for object location. Building on this insight, we propose `InfSplign`, a method that extracts object centroids from low-resolution attention maps of the UNet backbone and computes spatial losses which measure how far off the objects are from their intended target location. These losses are differentiated with respect to the current latent variables enabling gradient-based optimization of the predicted noise at that timestep. This inference-time approach guides the sampling trajectory toward generating more spatially coherent images without modifying the model weights. This makes our method easily scalable and widely applicable. Through extensive ablations and benchmark evaluations, we show that with this minimal but targeted intervention, we are able to compete with more complex inference-time approaches and expensive fine-tuning based methods and improve the spatial cognizance in the generated outputs of T2I diffusion models. Our core contributions can be summarized as follows:

- We introduce `InfSplign`, a sampling-based guidance method that enforces the correct spatial relationship without compromising the diversity of generated samples. It ensures that the objects are moved towards the region aligning with the spatial cue from the text prompt.
- We propose a set of simple yet effective spatial loss functions, which enforce images to adhere to spatial information defined in the text prompt. Despite their minimum complexity, these loss functions outperform both fine-tuning and inference-time spatial alignment methods. Extensive evaluations on three spatial benchmarks VISOR [21], T2I-CompBench [24] and GenEval [25], demonstrates that our gradient-based guidance consistently outperforms state-of-the-art approaches.
- Through extensive ablation studies, we analyze how key hyperparameters of the spatial loss, such as gradient strength and activation range, influence the effectiveness of the guidance signal. Our results show that careful tuning of these parameters significantly improves both spatial alignment and object accuracy, even in cases where the underlying diffusion model fails to generate them.

2 Related Works

Text-to-image (T2I) generation. Aims to generate visually realistic images that align with natural language prompts. While earlier work focused on GANs [26–28] and autoregressive models [29, 30], diffusion models [1, 2, 31] have recently become the dominant approach due to their superior image fidelity, diversity, and stability [22]. The integration of vision–language models (VLMs) like CLIP [20] further improves semantic alignment [32], yet recent studies show that even state-of-the-art models struggle with accurately capturing fine-grained textual details, particularly spatial relationships [10, 24]. To address this, some methods expand network architectures or introduce new training objectives [13, 33], but these require costly retraining. An alternative line of work proposes training-free inference-time strategies [15, 34], which are more efficient and easier to integrate into existing diffusion pipelines—motivating the direction of our approach.

Inference-time guidance for Diffusion Models. Inference-time guidance methods circumvent costly retraining by directly manipulating diffusion processes during sampling. Methods like Attend

& Excite [34] address the issue of missing objects by optimizing attention maps at inference time, but do not explicitly enforce spatial accuracy. Structured Diffusion Guidance [13] manipulates attention maps for improved layout control, yet lacks explicit modeling of spatial relationships described by textual prompts. Similarly, Composable Diffusion [22] interprets diffusion models as energy-based compositions of individual concepts, improving object presence but providing minimal spatial control. More targeted spatial inference-time methods, such as Prompt-to-Prompt [35] and DIVIDE&BIND [36], demonstrate the potential of directly modifying cross-attention maps. Recent information-theoretic insights further motivate inference-time interventions: analysis of mutual information between text prompts and images [10, 37], initial diffusion noise predetermines object layout generation [38], and sometimes needs to be guided to produce a valid sample [39] according to the prompt. Diffusion Self-Guidance [15] develops a framework for image editing by controlling the appearance, shape, size and location of objects but limits the sample diversity. Unlike prior methods that modify attention maps and use internal model representations for attribute binding or editing, InfSplign guides the denoising trajectory through spatial loss optimization, enforcing spatial relationships specified in the prompt, while preserving the diversity of the generated images.

Spatial Understanding in T2I Models. Existing efforts aimed at improving spatial understanding primarily fall into fine-tuning-based or layout-conditioned methods. Fine-tuning approaches enhance spatial reasoning by training models on spatially-aware datasets or using auxiliary objectives, such as reward-based optimization [9, 40–43]. SPRIGHT [40] is the first work to present a large scale vision-language dataset for fine-tuning diffusion model for spatial data. Another example is CoMPaSS [9] which significantly advances state-of-the-art spatial accuracy on common benchmarks by explicitly incorporating spatially labeled data during training. Nevertheless, these methods involve expensive retraining processes and risk destabilizing the pretrained diffusion backbone. As an inference-time approach, our method does not suffer from these limitations as it does not update any model weights, nor does it introduce computational overhead. Another category explicitly injects spatial layout information, such as bounding boxes, depth maps, or segmentation masks, to guide generation [18, 44–49]. Although effective, these methods depend on external layout inputs, requiring additional pre-processing steps and limiting usability when explicit spatial input is unavailable. Recent works have also leveraged large language models (LLMs) to provide layout-based guidance [50, 51], further increasing inference-time complexity and computational overhead. Unlike these methods, InfSplign does not require any additional input to the system other than the text prompt.

The inference-time methods closest to our work are REVISION [52] and STORM [53]. REVISION generates spatially accurate synthetic images, used as conditional information to the diffusion model, so the problem is transformed into an image-to-image (I2I) pipeline for spatially coherent results. STORM defines a distribution-based loss function, leveraging Optimal Transport (OT) [54], to adjust attention maps to adhere to spatial composition by defining target distribution at a fixed location corresponding to the given spatial relationship. They succeed in pushing the limits in the SoTA results on two mainstream spatial benchmarks, VISOR [21] and T2I-CompBnech [24]. Unlike REVISION, our approach does not rely on additional synthetic images, and in contrast to STORM, InfSplign preserves sample diversity by avoiding fixed spatial placements, allowing spatial relationships to emerge naturally during generation.

3 Preliminaries

Diffusion Models. provide an effective framework for sampling from complex data distributions $q(x)$ by learning to invert a forward diffusion process. The forward process is a Markov chain iteratively adding a small amount of random noise to a "clean" data point $x_0 \in \mathcal{X}$ over T steps. The noisy sample at step t is given by $x_t{=}\sqrt{\bar{\alpha}_t}x_0{+}\sqrt{1-\bar{\alpha}_t}\varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, I)$, $\alpha_t{=}1{-}\beta_t$, $\bar{\alpha}_t{=}\prod_{t=1}^{T}\alpha_t$, $\{\beta_t\}_{t\in[T]}$ is a variance schedule [5, 31] and $\mathcal{N}(0, I)$ refers to a normal distribution. To estimate $q(x)$, the diffusion model p_θ learns the conditional probabilities $q(x_{t-1}|x_t)$ to reverse the diffusion process starting from a fully noisy sample $x_T \sim \mathcal{N}(0, I)$. Here, a neural network $\varepsilon_\theta(x_t, t)$ (typically a U-Net [55]) attempts to predict the noise added to x_{t-1} in the forward process. Furthermore, using a conditioning signal c, diffusion models can be extended to sample from $p_\theta(x|c)$ as $\hat{x}_{t-1} = \hat{x_t} - \epsilon_\theta(\hat{x_t}; t, c)$. The conditioning signal, c, can take diverse forms, from text prompts and categorical information to semantic maps [56, 57]. Our work focuses on a text-conditioned model, Stable Diffusion [58], which has been trained on a large corpus consisting of M image-text pairs $\mathcal{D} = \{(x^i, c^i)\}_{i=1}^{M}$.

Inference-Time Guidance of Diffusion Models. A notable strength of diffusion models lies in their flexibility. By manipulating the sampling process, they can be adapted to a wide range of downstream tasks at inference time, i.e., without retraining or fine-tuning. Diffusion models can be guided towards a variety of downstream tasks through both input conditioning (e.g., text prompts) and external reward models (e.g., CLIP-based scores). This process, known as guidance, influences the denoising trajectory to better align the generated samples with the desired outcomes. A common form is classifier guidance [4], which steers generation using gradients from a pretrained image classifier. More recent approaches use classifier-free guidance [59], which eliminates the need for an external classifier by training the model to denoise both with and without conditioning, and then interpolating between the two at inference time.

Diffusion models can be interpreted through their score-based formulation, where the model estimates the gradient of the log probability density, $\nabla_{x_t} \log p(x_t, t)$. This gives us intuition about the direction in which we have to move in space to increase the log likelihood of our data sample based on some conditional information. While the denoising formulation, $\epsilon_\theta(x_t, t)$, gives us a prediction of the noise that was added by the forward diffusion model at each timestep. Classifier guidance is defined for the gradient of the score function as:

$$\nabla_{x_t} \log p(x_t|y, t) = \nabla_{x_t} \log p(x_t, t) + \gamma \nabla_{x_t} \log p(y|x_t, t), \tag{1}$$

where γ is the guidance strength. Due to the equivalence between the score-based formulation and the denoising models, classifier guidance can be written in terms of denoising models using the equivalence relation defined in:

$$\nabla_{x_t} \log p(x_t|y, t) = -\frac{1}{\sigma_t} \nabla_{x_t} \epsilon_\theta(x_t, y, t), \tag{2}$$

So the final classifier guidance equation can be given by:

$$\epsilon_\theta(x_t, y, t) = \epsilon_\theta(x_t, t) - \gamma \sigma_t \nabla_{x_t} \log p(y|x_t, t). \tag{3}$$

The drawback of classifier guidance is that it relies on an additional classifier that needs to learn to predict the class of images given arbitrarily noisy images. Since this poses significant difficulties in training such a classifier that gives reliable predictions, the classifier-free guidance (CFG) mechanism avoids the need from an external classifier by using two diffusion models—unconditional and conditional. The score-based formulation of CFG is defined as:

$$\nabla_{x_t} \log p(x_t|y, t) = \nabla_{x_t} \log p(x_t, t) + \gamma(\nabla_{x_t} \log p(x_t|y, t) - \nabla_{x_t} \log p(x_t, t)). \tag{4}$$

In the next chapter, we present how our method builds on top of diffusion models to improve the spatial alignment between prompts and generated images.

4 `InfSplign`: Inference-time Spatial Alignment

Diffusion models have consistently shown poor adherence to spatial information in generated images [40, 53]. One of their limitations is the inability of the CLIP text encoder to encode spatial relationships in a meaningful way [9]. To address this problem, we add a guidance term quantifying the misalignment between the generated image and the spatial cue in the prompt. The idea is to actively nudge the generation process towards generating more spatially-aware images. Our method improves spatial understanding at inference-time, without additional retraining and computational overhead. It serves as a lightweight, plug-and-play enhancement to diffusion models.

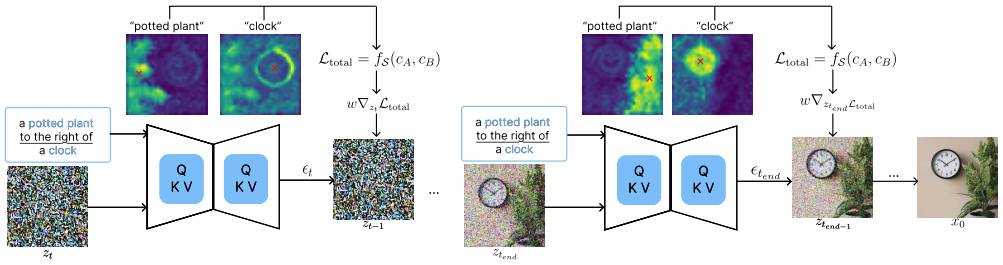


Figure 2: Overview of proposed spatial-cognizant sampling pipeline. `InfSplign` modifies the inference trajectory from timestep T to t_{end} by extracting attention maps for each object token, estimating their spatial centroids A and B, and computing a spatial loss based on their relative positions. The gradient of this loss with respect to the noise at the current timestep is used to adjust the noise vector before proceeding to the previous denoising step.

Figure 2 gives a high-level overview of our inference-time guidance framework. The input to the system consists of a user specified text prompt, e.g. *"a potted plant to the right of a clock"*, and the noisy latent embedding z_t produced at timestep t of the reverse diffusion process. We represent the prompt as a structured triplet - $\langle A, R, B\rangle$, where A and B are the object tokens and $R \in \mathcal{R}$, where $\mathcal{R}$ is the set of spatial relationships, e.g. $A =$ "potted plant", $B =$ "clock" and $R =$ "to the right of".

To guide the denoising process, we extract the cross-attention maps corresponding to the object tokens A and B. These attention maps are used to estimate the spatial location of each object by computing its centroid. The centroid is defined as the center of mass of the distribution over the image coordinates. We apply a spatial loss function, defined in Equation 5, which penalizes deviations from the spatial relationship R:

$$\mathcal{L}_{\text{total}} = f_{\mathcal{S}}(c_A, c_B), \tag{5}$$

where $c_A = (x_A, y_A)$ and $c_B = (x_B, y_B)$ are the centroids for objects A and B, respectively, and $\mathcal{S}$ is the spatial relationship. The gradient of this loss with respect to the latent z_t provides a guidance signal that modifies the predicted noise. The gradient is used to update the noise such that the latent is shifted in a direction that leads to a more spatially aligned image. The gradient $\nabla_{z_t}\mathcal{L}_{\text{total}}$ is multiplied by a weight hyperparameter w to adjust the scale of the guidance signal such that it serves as a meaningful addition to the CFG objective. Equation 6 shows how we intervene in the reverse diffusion process.

$$\epsilon_t = \epsilon_\theta(z_t; t) + \gamma(\epsilon_\theta(z_t; t, y) - \epsilon_\theta(z_t; t)) + w\nabla_{z_t}\mathcal{L}_{\text{total}}. \tag{6}$$

This updated noise prediction is used to compute the denoised latent z_{t-1} guiding the generation towards spatially cognizant images. Thus, the final update direction combines both semantic (via CFG) and geometric alignment (via spatial loss). This procedure is applied iteratively over diffusion timesteps $t \in \{T, T-1, ..., t_{end}\}$ contributing to both textually and spatially faithful outputs.

In the beginning of the reverse diffusion process the "potted plant" object from Figure 2 is generated in the leftmost part of the image which is misaligned with the target location defined in the prompt. Similarly, the "clock" object is misplaced to the right. At timestep t_{end} the attention maps clearly show the effect of the spatial loss: the objects are pushed towards the correct spatial regions - the "potted plant" is "to the right of" the "clock". After timestep t_{end}, the vanilla reverse diffusion is applied until the final image is fully denoised.

4.1 Spatial Loss for Object Positioning

To generate images that honor spatial relationships in the prompt, e.g. *"a potted plant to the right of a clock"*, we guide the diffusion model in the denoising process through an external signal, our spatial loss, defined over object centroids derived from the cross-attention maps of the UNet. The guidance loss function must (i) be differentiable with respect to the latent variable z_t, (ii) produce meaningful gradients for spatially-cognizant image generation and (iii) encode the geometric semantics of the spatial relationship.

From Attention to Centroids. Previous research [60] shows that cross-attention layers encode information about the spatial layout of objects in the generated image. Thus, we extract the attention maps from a selected set of layers from the decoder of the UNet which capture most of the objects' structure and encode their location more reliably. Equation 7 describes the estimation of each object's position as the centroid. Let $\mathcal{A}_{i,j,t}^{(k)} \in \mathbb{R}^{H_{i,j} \times W_{i,j}}$ denote the attention map at timestep t, from attention layer j in decoder block i of the UNet, corresponding to token k from the input prompt. The centroid $\mathbf{c}^{(k)} = (x_c^{(k)}, y_c^{(k)})$ of token k is computed as:

$$\mathbf{c}^{(k)} = \text{centroid}(\mathcal{A}_{i,j,t}^{(k)}) = \left(\frac{\sum_{h,w} \mathcal{A}_{i,j,t}^{(k),\text{thresh}}[h,w] \cdot x_w}{\sum_{h,w} \mathcal{A}_{i,j,t}^{(k)}[h,w]}, \quad \frac{\sum_{h,w} \mathcal{A}_{i,j,t}^{(k),\text{thresh}}[h,w] \cdot y_h}{\sum_{h,w} \mathcal{A}_{i,j,t}^{(k)}[h,w]} \right), \tag{7}$$

$\mathcal{A}_{i,j,t}^{(k),\text{thresh}}$ is obtained through mean-based thresholding applied to the attention map to enhance the region occupied by the object, highlight its silhouette more accurately and remove noise which hinders the precise estimation of the centroid. Then, a weighted average is computed over the coordinates of the latent z_t and the attention weights and normalized by the sum of the attention activations. Thus, we show that the loss function, shown in Equation 5, is differentiable with respect to the latent z_t because the centroids are extracted from the UNet cross-attention layers which are dependent on the latent.

Defining Spatial Displacement. We express the spatial relationship R between the two objects A and B as a difference between their centroids, denoted as Δ and defined in Equation 8. Depending on the relationship, Δ is computed along the appropriate axis:

$$\Delta = \begin{cases} x_B - x_A, & \text{for } \textit{"left"}, \\ x_A - x_B, & \text{for } \textit{"right"}, \\ y_B - y_A, & \text{for } \textit{"above"}, \\ y_A - y_B, & \text{for } \textit{"below"}, \end{cases} \tag{8}$$

where Δ captures the directional alignment between the two objects and signals whether this adheres to the specified spatial relation.

Spatial Loss Definition. We formulate the spatial loss function as follows:

$$\mathcal{L}_{\text{spatial}} = f_{\text{spatial}}(\alpha(m - \Delta)), \tag{9}$$

where $f_{\text{spatial}} \in \{\texttt{ReLU}(.), \texttt{Leaky ReLU}(.), \texttt{GELU}(.), \texttt{Sigmoid}(.)\}$ and α is a scaling factor, sharpening the slope around the decision boundary, which affects how strictly the model is penalized. The margin indicates the acceptable minimum distance between the objects' centroids. The spatial loss results in a small penalty if the objects are placed correctly with respect to the target spatial relation, e.g. if $\mathcal{S} =$ *"to the right of"*, $x_A - x_B > m, \mathcal{L}_{\text{spatial}} \to 0$. Whereas, the loss would increase when objects are too close to each other, $\Delta < m$, and when the objects violate the spatial relation. This definition satisfies the other two conditions for a well-crafted spatial loss guidance signal.

Orthogonal Alignment Loss. To prevent degenerate configurations where the objects align with respect to the right axis but are offset along the wrong axis, we define a helper alignment term, δ:

$$\delta = \begin{cases} y_A - y_B, & \text{for } \textit{"left"}, \textit{"right"}, \\ x_A - x_B, & \text{for } \textit{"above"}, \textit{"below"}. \end{cases} \tag{10}$$

This encourages alignment along the orthogonal axis and helps spatial instructions to be interpreted correctly. For instance, this ensures that the relationship *"A to the right of B"* does not become *"A to the right of and above B"*. Thus, we formulate an additional loss term:

$$\mathcal{L}_{\text{orthogonal}} = \|\delta\|_p^p, \tag{11}$$

where $\| \cdot \|_p$ is the p-norm and for our method, we consider $p \geq 1$. Therefore, the final spatial loss function can be summarized as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{spatial}} + \mathcal{L}_{\text{orthogonal}}. \tag{12}$$

Note that the proposed spatial loss does not predetermine object positions which preserves generation diversity. Instead, it penalizes misplaced objects through gradients computed with respect to the latent z_t during sampling. These gradients are used to steer the predicted noise $\epsilon_\theta(z_t, t, y)$ in the reverse diffusion process to a more spatially coherent output.

Gradient-based Update. The guidance mechanism applies the chain rule to backpropagate the spatial loss to the latent space. The update is written as:

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial z_t} = \frac{\partial \mathcal{L}_{\text{spatial}}}{\partial \Delta} \cdot \frac{\partial \Delta}{\partial z_t} + \frac{\partial \mathcal{L}_{\text{orthogonal}}}{\partial \Delta} \cdot \frac{\partial \Delta}{\partial z_t}. \tag{13}$$

The $\frac{\partial \mathcal{L}_{\text{spatial}}}{\partial \Delta}$ term corresponds to the slope of the spatial loss function responsible for the strength and the direction of the guidance signal. The combined gradient, $\frac{\partial \mathcal{L}_{\text{total}}}{\partial z_t}$, updates the latent z_t through the predicted noise ϵ_t the driving force towards a spatially coherent image generation.

Pseudocode. To achieve spatially aware image generation, we guide the denoising process for a fixed number of steps. Algorithm 1 shows the pseudocode for a single reverse diffusion step under spatial guidance. The spatial loss is computed using the centroids predicted from the attention maps and is used to adjust the noise prediction. This update is performed at inference-time.

Algorithm 1 A Single Denoising Step with Spatial Alignment Loss

Input: Prompt $\mathcal{P} = \langle A, R, B \rangle$, object tokens A, B and spatial relationship $R \in \mathcal{R}$, Latent z_t at timestep t, Stable Diffusion model SD, Spatial guidance function f_{spatial} with parameters (α, m), Guidance weight w

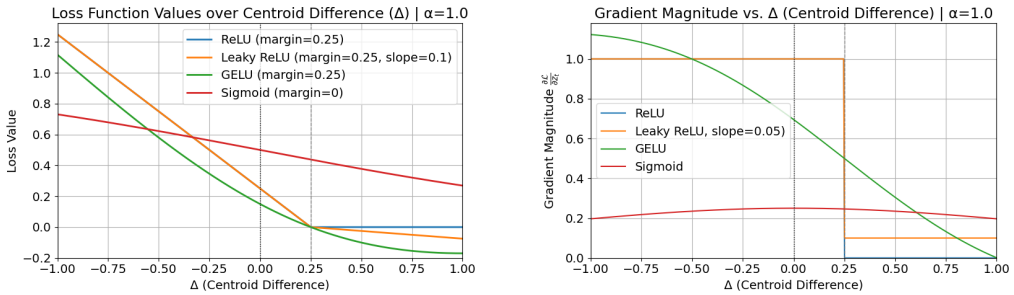
Output: A noised latent z_{t-1} for the next timestep

$\mathcal{A}_t \leftarrow SD(z_t, \mathcal{P}, t)$ ▷ Forward pass to store attention maps
$c_A \leftarrow \texttt{Centroid}(\mathcal{A}_t, A)$
$c_B \leftarrow \texttt{Centroid}(\mathcal{A}_t, B)$
$\Delta \leftarrow \texttt{Difference}(c_A, c_B, R)$
$\delta \leftarrow \texttt{Difference}(c_A, c_B, R)$
$\mathcal{L}_{\text{spatial}} \leftarrow f_{\text{spatial}}(\Delta, \alpha, m)$
$\mathcal{L}_{\text{orthogonal}} \leftarrow \|\delta\|_p^p$
$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{spatial}} + \mathcal{L}_{\text{orthogonal}}$
$\epsilon_t \leftarrow SD(z_t, \mathcal{P}, t)$ ▷ Noise prediction
$\epsilon_t \leftarrow \epsilon_t + w \cdot \nabla_{z_t} \mathcal{L}_{\text{total}}$
$z_{t-1} \leftarrow z_t - s \cdot \epsilon_t$
return z_{t-1}

4.2 Loss Function Behavior and Spatial Guidance Dynamics

This section gives an overview of the behaviour of the spatial loss functions, introduced in the previous section, and their impact on guiding the diffusion model towards more spatially-aware image generation. The gradients of these losses update the predicted noise and in turn shift the latent z_t which directly influences how the model corrects spatial misalignment. Therefore, it is of key importance to understand the shape and the slope of each spatial loss function such that we can interpret its effect on guiding the denoising trajectory.

Loss Function Comparison. Figure 3 provides insights into how the choice of a loss function determines the strength of spatial guidance. Figure 3a illustrates the behavior of the spatial loss functions for $\alpha = 1$, ReLU, Leaky ReLU, GELU and Sigmoid, defined over the centroid difference Δ between the two objects. When the object locations violate the input spatial relationship (i.e., when $\Delta{<}0$), ReLU, Leaky ReLU and GELU produce higher loss values than Sigmoid, enforcing a stronger correction. Within the interval $0{<}\Delta{<}m$, these losses penalize configurations where the spatial relation is satisfied directionally but the objects are overlapping. The ReLU loss becomes exactly zero for $\Delta{\geq}m$, providing no further gradient. Leaky ReLU extends this with a small negative slope after the margin, while GELU provides a smoother decay. Higher α values result in steeper slope around $\Delta{=}0$, producing sharper gradients and stronger updates, effectively increasing the penalty for wrong object placements. The Sigmoid loss differs fundamentally. Since we assign $m = 0$, it penalizes misalignment symmetrically and continuously. When $\alpha{=}1$, the Sigmoid function almost behaves as a linear function in the range $\Delta \in [-1, 1]$. The loss decreases very slowly and still penalizes the model compared to the remaining spatial losses. However, for large α values the Sigmoid function attains a steeper slope, increasing the range of penalties and leading to a more refined and accurate guidance strategy.



(a) Behaviour of the spatial loss functions

(b) Gradient behaviour of the spatial loss functions

Figure 3: Visualization of the spatial loss functions and their gradients. (a) compares the different spatial losses. Each function enforces different range of penalties: ReLU applies a hard cutoff, Leaky ReLU preserves a soft negative penalty after satisfying the spatial constraint, GELU enables smooth refinement and Sigmoid provides monotonously decreasing guidance signal. The gradients in (b) influence how spatial deviations are corrected during inference.

Gradient-Based Guidance. To determine which is the most informative spatial signal, we have to understand the gradients of each spatial loss as that is what controls the strength of the update applied to the latent variables z_t during inference. The full computation of the gradient is presented in Equation 13, while the gradient magnitude, irrespective of the direction of the update, is shown in Figure 3b. The key differences between the loss functions emerge in how strongly they react to misalignment. ReLU reacts with a fixed correction strength for any violation of the margin, but offers no signal once the spatial relation is satisfied, potentially halting refinement too early. Leaky ReLU retains a small gradient even after the spatial constraint is satisfied, ensuring that object positions continue to be pushed to the right spatial region. With its smoothly decaying derivative, GELU facilitates fine-grained updates. Sigmoid provides a soft continuous signal across the full domain. Higher α values allow GELU and Sigmoid to achieve stronger alignment with spatial information. These behaviours influence how effectively each loss can improve spatial understanding in generated images at inference-time.

5 Experiments

In this section, we present a comprehensive evaluation of our method. Firstly, the benchmarks, evaluation metrics and implementation details are introduced. Then, we demonstrate the effectiveness of our approach through comparisons with state-of-the-art baselines. Next, we conduct detailed ablation studies to assess the impact of key components. Finally, we showcase qualitative results that highlight the spatial alignment and diversity achieved by our method.

5.1 Benchmarks and metrics

VISOR [21]. This metric determines if the spatial relationship information from the prompt is accurately reflected in the generated image. The benchmark consists of a set of text prompts, the corresponding objects and spatial relationship for each prompt. The object categories come from the MS-COCO dataset [61]. There are 25280 spatial prompts in total, 4 generated images per prompt and 4 spatial relationships, {*"to the left of"*, *"to the right of"*, *"above"*, *"below"*}. The benchmark defines the following set of metrics: *OA* (object accuracy) indicating the number of images in which both objects were generated and detected, $\text{VISOR}_{\text{cond}}$ estimates the conditional probability of how many of the images adhere to the spatial relationship given that both objects are generated correctly, $\text{VISOR}_{\text{uncond}}$ (or VISOR in short) gives a ratio of the number of images generated with a correct spatial relationship and the two objects generated correctly out of all generated images, and VISOR_1 to VISOR_N define the number of images out of N that have VISOR=1. VISOR chooses $N = 4$.

T2I-CompBench [24]. Constitutes a benchmark for compositional T2I generation. There are 300 spatial prompts used for evaluation purposes, 10 images generated per prompt and 7 spatial relationships, {*"on the left of"*, *"on the right of"*, *"on the top of"*, *"on the bottom of"*, *"on the side of"*,

Table 1: Summary of evaluation benchmarks for spatial understanding in T2I generation.

Benchmark	#Prompts	#Images / Prompt	#Relations
VISOR [21]	25,280	4	4 (*left, right, above, below*)
T2I-CompBench [24]	300	10	7 (*left, right, above, below, on the side of, next to, near*)
GenEval [25]	100	4	4 (*left, right, above, below*)

"next to", *"near"*}. The objects are chosen from three categories: persons, animals and objects. The spatial metric considers the difference between the x and y coordinates of the estimated centers of the objects, showing if the objects are generated consistently with the spatial relation, their absolute difference, to measure in which axis the objects are further apart, and the intersection-over-union (IoU) to prevent object overlap. The final score is determined by the sum of the 25% of the confidence score for each of the two detected objects and 50% of the positional score. If the final predicted score for the metric is below 50%, the score is set to 0, due to low confidence.

GenEval [25]. The GenEval benchmark also evaluates compositionality of T2I generation. It consists of 100 spatial prompts. The object categories used are also from the MS-COCO dataset [61], 4 images are generated per prompt for 4 spatial relationships {*"left of"*, *"right of"*, *"above"*, *"below"*}. The benchmark calculates the difference between the x and y coordinates (offset) of the two centroids and adjusts this offset by accounting for the their sizes through a threshold. If the objects are larger, the relationship should tolerate more overlap between them. Based on the normalized updated offset, one or two spatial relationships can be determined, the four given relationships and their combinations, e.g. left and above. The final spatial score is computed as the average over all images. Table 1 provides a concise comparison of the three spatial evaluation benchmarks used in our experiments, highlighting the number of prompts, images per prompt, and spatial relations. Additional technical details and evaluation procedures are available in Appendix A.

5.2 Implementation Details

We apply `InfSplign` on top of the official Stable Diffusion v1.4 and Stable Diffusion v2.1 T2I models [1] for 500 inference timesteps. The spatial loss guidance is applied in the first 25% of the denoising steps. The margin hyperparamter is set to $m = 0.25$ for ReLU, Leaky ReLU and GELU, and $m = 0$ for Sigmoid. The CFG guidance scale $\gamma = 7.5$ from Equation 4, while the guidance weight for the spatial loss is $w = 1000$. The attention layers used for SD v1.4 and SD2.1 are the first 6 from the decoder of the UNet. The experiments were run on an $A40$ GPU with 48GB VRAM with a batch size of 2, except for VISOR [21] which required multiprocessing on the $A40$ GPUs for faster generation time due to the large size of this dataset. The generation time for one image is around 23 seconds. We used the same random seeds for each benchmark as defined in the original papers, 42 for VISOR and GenEval and $\{42, 43, ..., 51\}$ for T2I-CompBench.

5.3 Baselines

To demonstrate the effectiveness of our method, we compare our work to the relevant baselines, both inference-time and fine-tuning approaches, innovating the spatial alignment domain of diffusion models. To ensure a fair comparison with previous works, we evaluate on the same diffusion backbones used by existing literature, Stable Diffusion v1.4 and v2.1 [1]. These baselines are adopted by inference-time research papers such as REVISION [52] and STORM [53], and the fine-tuning-based methods SPRIGHT [40] and CoMPaSS [9], focused on the task of improving the spatial understanding capabilities of T2I generation models.

Unlike fine-tuning methods that require further training with additional carefully curated data, our method does not change the model's weights and can be applied to any diffusion framework without any computational overhead. While fine-tuning has been shown to be powerful in learning to adapt to new tasks [62, 63], such as learning complex spatial priors in SPRIGHT and CoMPaSS through very well designed spatially aligned training dataset, it introduces an extra level of computational complexity, might lead to overfitting to the new training data distribution and a collapse of the model's generation capability with respect to the original broader training distribution. `InfSplign` eliminates these risks, not only does it avoid supervision and leave the model's weights untouched, but it also

requires no additional computational overhead. Despite its simplicity, it proves highly effective, matching or even surpassing several state-of-the-art fine-tuned methods for spatial alignment.

5.4 Comparison with State-of-The-Art

We evaluate our model's effectiveness across three spatial benchmarks against state-of-the-art methods. These datasets cover different aspects of spatial reasoning which facilitates the assessment of our module's capabilities for spatially cognizant image generation. `InfSplign` consistently achieves SOTA performance on all three benchmarks or is on-par with the best competitors, fine-tuning works, in the spatial alignment field despite being an inference-time method. This suggests that our centroid-based spatial loss provides a strong guidance signal during sampling.

Quantitative Evaluation: VISOR. VISOR [21] measures the ability of T2I generation to align the resulting images with the spatial instruction present in the text prompt. Table 2 summarizes the VISOR scores obtained using our method with the Sigmoid spatial loss and how it compares to the relevant works in the spatial reasoning domain. On SD 2.1 `InfSplign` outperforms both fine-tuning-based methods (e.g. SPRIGHT [40], CoMPaSS [9]) and inference-time methods (e.g. REVISION [52], STORM [53]). Our method achieves the highest object accuracy (OA) with a score of 63.20 leading to the highest $\text{VISOR}_{\text{cond}}$, 98.14. What this metrics tell us is that in the cases where both objects are successfully generated in the final image, the spatial relationship between them is correct. The $\text{VISOR}_{\text{uncond}}$, 62.03, is very close to the SOTA result reported by CoMPaSS [9], 62.06, showing that our method synthesizes images adhering to spatial information more than half the time, which indicates that `InfSplign` is not just guessing but it can actually captures semantic meaning of spatial phrases. On SD 1.4 `InfSplign` attains slightly lower scores than the competitors but it still shows a remarkable increase in the $\text{VISOR}_{\text{cond}}$ score, 98.65, reinforcing its effectiveness in spatially aligned image generation. On SD 2.1 VISOR_1 and VISOR_2, 89.71 and 74.60, respectively, imply higher success rate in generating at least 2 out of 4 spatially-aware images for all test prompts. Although VISOR_3 and VISOR_4 are quite challenging, our results are slightly lower than CoMPaSS [9] but are higher than the current best inference-time methods.

Table 2: Results on the VISOR benchmark using the Sigmoid spatial loss. All metrics are reported as percentages, thus, higher score is better. We compare against both fine-tuning and inference-time methods using SD1.4 and SD2.1 as base models. Best results in each column are **bolded**.

Alignment	Model	OA	$\text{VISOR}_{\text{uncond}}$	$\text{VISOR}_{\text{cond}}$	VISOR_1	VISOR_2	VISOR_3	VISOR_4
Stable Diffusion 1.4 Based								
	SD [1]	29.86	18.81	62.98	46.60	20.11	6.89	1.63
Fine-tuning	CoMPaSS [9]	–	57.41	87.58	83.23	67.53	**49.99**	**28.91**
Inference-time	REVISION [52]	53.96	52.71	97.69	77.79	61.02	44.90	27.15
	STORM [53]	**61.01**	**57.58**	94.39	**85.93**	**69.71**	49.01	25.70
	InfSplign (ours)	53.56	52.84	**98.65**	83.76	64.51	42.94	20.15
Stable Diffusion 2.1 Based								
	SD [1]	47.83	30.25	63.24	64.42	35.74	16.13	4.70
Fine-tuning	SPRIGHT [40]	60.68	43.23	71.24	71.78	51.88	33.09	16.15
	CoMPaSS [9]	–	**62.06**	90.96	85.02	71.29	**56.03**	**33.73**
Inference-time	REVISION [52]	48.26	47.11	97.61	76.07	55.75	37.10	19.53
	STORM [53]	62.55	59.35	94.88	88.34	71.75	52.03	25.42
	InfSplign (ours)	**63.20**	62.03	**98.14**	**89.71**	**74.60**	54.53	29.28

Quantitative Evaluation: T2I-CompBench. T2I-CompBench [24] evaluates the ability of generative models to adhere to compositionality. Table 3 presents the different baselines, which report it, and all the loss functions we analyzed. `InfSplign` performs best using the Sigmoid spatial loss and beats both inference time and fine-tuning-based approaches. While CoMPaSS [9] achieves the highest score on SD 1.4, 0.34, our methid gets very close with GELU and Sigmoid spatial losses. But, on SD 2.1 `InfSplign` surpasses all the baselines with a spatial score of 0.3836 achieved with the Sigmoid loss.

Quantitative Evaluation: GenEval. GenEval [25] provides a spatial framework for evaluation of generated images with the emphasis on object presence due to their observation that in most failure

cases the objects are overlapping. Table 4 presents the results obtained using `InfSplign` on GenEval using all spatial loss functions. The authors of this benchmark use a minimum distance constraint to ensure objects are not too close to each other which could hinder determining the spatial relationship correctly. Among the spatial losses, the best performing one for the GenEval benchmark is GELU, with scores 0.545 and 0.703 on SD 1.4 and 2.1, respectively. This loss is well suited for enforcing object non-overlap which aligns with how the metric is defined. Although the fine-tuning-based work CoMPaSS [9] is the only one in the literature that reports scores on GenEval, we get ahead of them by a big margin - 9% on SD 1.4 and 19% on SD 2.1, without requiring any retraining.

Table 3: Results on the spatial score of the T2I-CompBench compositional benchmark. We compare fine-tuning and inference-time methods using SD1.4 and SD2.1 backbones. The best results in each column are **bolded**.

Method	SD1.4	SD2.1
SPRIGHT [40]	–	0.2133
CoMPaSS [9]	**0.3400**	0.3200
STORM [53]	0.1613	0.1981
InfSplign (ReLU)	0.2783	0.3324
InfSplign (Leaky ReLU)	0.3221	0.3786
InfSplign (GELU)	0.3260	0.3677
InfSplign (Sigmoid)	0.3244	**0.3836**

Table 4: Results on the spatial score of the GenEval compositional benchmark. We compare our method with the only available fine-tuning baseline on SD1.4 and SD2.1. All scores represent spatial alignment accuracy. The best results in each column are **bolded**.

Method	SD1.4	SD2.1
CoMPaSS [9]	0.460	0.510
InfSplign (ReLU)	0.378	0.545
InfSplign (Leaky ReLU)	0.483	0.596
InfSplign (GELU)	**0.545**	**0.703**
InfSplign (Sigmoid)	0.520	0.643

5.5 Ablation Studies

Ablation Subset Creation. To gather more insights about the significance of the hyperparameters our method requires, we design multiple ablation studies. For this, we need to obtain an ablation dataset. We decide to focus on the VISOR dataset since it is the most challenging in term of size, it considers all combinations of object categories for all spatial relationships. We performed inference-time hyperparameter exploration and not model training. Therefore, having a subset of prompts for these ablations that is used in the final evaluation does not lead to overfitting since the hyperparameters that are being optimized are not learnable. Thus, we are not compromising the generalizability of `InfSplign`.

We performed data exploration on the failure cases, where one or both objects are not generated and when the spatial relationship is incorrect. We constructed a subset of 1000 failure prompts, 500 with one object generated and the other 500 with no object generated. The goal is to capture both failure and success cases in this subset such that it provides a good representation of the full dataset. Since VISOR prompts use the 80 COCO object categories, we sample one object from each super-category and create object pairs within this subset of objects. There are 12 super-categories, so we compose $12 * 11 = 132$ object combinations. This is done such that the model's generation ability is not biased towards any class. The way we pick each object from a super-category is by maximizing the object combinations appearing in the subset of 1000 failure cases. Furthermore, we ensure that there is an equal number of spatial relationship examples - we get exactly 44 prompts per relationship ($132/4 = 44$). The final ratio of failure success cases in this subset is 1:2.

This VISOR subset consists of 132 prompts which are used for all ablation studies discussed in this section.

Effectiveness of Orthogonal Loss Component. With this ablation study, we aim to evaluate if the inclusion of a secondary spatial penalty along the orthogonal axis to the primary axis of the spatial relation, e.g. vertical for *"left of"*, affects performance. This addition is motivated by cases where the objects are generated diagonally in opposite corners of the image. For example, if A is generated in the bottom left corner and B is generated in the top right corner for the prompt *"A to the left of B"*, the spatial relationship from the text is captured but another spatial relationship, namely *"below"*, might also be considered as correct, potentially misleading the evaluation detector. By discouraging these ambiguities, our goal is to enforce stronger spatial alignment.

In Table 5, we present the results from testing the combinations of the $\mathcal{L}_{\text{spatial}}$ loss and ℓ_1-norm and ℓ_2-norm for the $\mathcal{L}_{\text{orthogonal}}$ loss term in Equation 11, using SD 1.4 as our baseline. Across all loss functions, $\mathcal{L}_{\text{spatial}}$ and the ℓ_2-norm of $\mathcal{L}_{\text{orthogonal}}$ yield the highest object accuracy (OA) and VISOR$_{\text{uncond}}$. In fact, GELU achieves the highest OA, 36.553, VISOR$_{\text{uncond}}$ of 36.174 and VISOR$_{\text{uncond}}$ of 98.964. This shows that the combination between $\mathcal{L}_{\text{spatial}}$ and ℓ_2-norm of $\mathcal{L}_{\text{orthogonal}}$ is the most effective and indicates stronger and more reliable spatial alignment and object accuracy. These results indicate that, for SD 1.4, the addition of our orthogonal loss $\mathcal{L}_{\text{orthogonal}}$ improves spatial alignment, particularly by enhancing object presence. Most of the gains observed in the VISOR$_{\text{uncond}}$, stem from improved object accuracy, rather than purely spatial alignment. This effect arises because our primary spatial loss emphasizes alignment along a single axis, which can inadvertently displace objects along the orthogonal direction, leading to partial or complete object disappearance. To mitigate this, our orthogonal loss maintains object placement along the complementary axis, preserving recognizability.

Table 5: Ablation results of `InfSplign` on SD 1.4 for the different terms of the spatial losses, $\mathcal{L}_{\text{spatial}}$ and $\mathcal{L}_{\text{orthogonal}}$ which can be ℓ_1-norm and ℓ_2-norm. Scores are reported on the OA, VISOR$_{\text{uncond}}$, and VISOR$_{\text{cond}}$ metrics.

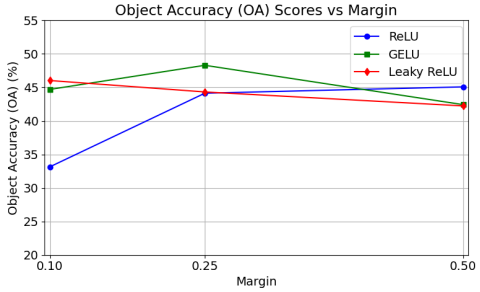
Model	**OA**	**VISOR$_{\text{uncond}}$**	**VISOR$_{\text{cond}}$**
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}}$)	33.333	32.008	96.023
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	30.114	29.924	99.371
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	35.417	34.280	96.791
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}}$)	31.439	30.682	97.590
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	31.250	30.871	98.788
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	35.038	34.470	98.378
InfSplign (GELU, $\mathcal{L}_{\text{spatial}}$)	33.144	32.765	98.857
InfSplign (GELU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	33.144	32.386	97.714
InfSplign (GELU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	36.553	36.174	98.964
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}}$)	33.333	32.955	98.864
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	32.955	32.386	98.276
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	34.280	33.712	98.343

This also explains why the improvements from our loss are less pronounced when using a sigmoid-based alignment function. Due to its saturating behavior, the sigmoid yields smaller gradients when objects are not in close proximity, resulting in less aggressive spatial adjustments and reducing the risk of object omission. We further validate these findings in Table 7 (see Appendix B), where we apply the same loss ablation to SD 2.1. As expected, the benefits of adding an orthogonal loss diminish for stronger baselines that already perform well in object preservation.

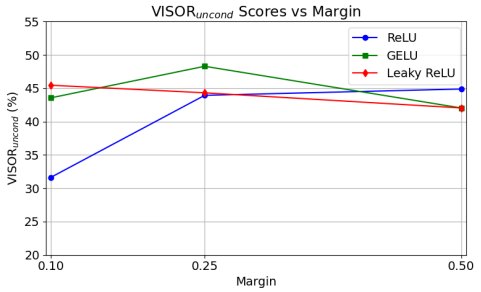
Effect of Margin. The margin hyperparameter m determines the minimum required separation between object centroids and affects the strength of the spatial alignment loss. Since object coordinates are normalized to the range $[0, 1]$, the potential margin values must lie below 0.5. As shown in Figure 4, the influence of m varies across the loss functions. For ReLU, increasing m from 0.1 to 0.5 leads to a consistent improvement in both object accuracy (Figure 4a) and VISOR scores (Figure 4b, Figure 4c). This is because the ReLU gradient becomes zero when the alignment error $\Delta{=}m$ and remains inactive for smaller deviations. A larger margin prolongs the range over which the loss remains active, yielding more informative gradients during the crucial early diffusion steps that shape object spatial positioning [64]. The GELU loss achieves its best performance at $m{=}0.25$. While it behaves similarly to ReLU near the margin, GELU allows non-zero gradients beyond it. This gradient *"leakage"* becomes problematic at larger margins, guiding objects that are already well-aligned to drift further apart. This results in decreased object accuracy, suggesting that excessive spatial enforcement may push objects outside the visible frame. A similar effect is observed with Leaky ReLU. Because it always produces gradients even for well-aligned pairs, larger margin values amplify this leakage, leading to declining object accuracy. We also observe that the conditional spatial alignment metric VISOR$_{\text{cond}}$ peaks at intermediate margin values rather than at the extremes $m{=}0$ or $m{=}0.5$ for all three loss functions. When $m{=}0$, overlapping objects incur no penalty, allowing invalid spatial alignment; conversely, at $m{=}0.5$ the loss becomes almost linear, yielding gradients that

lack spatial specificity. Intermediate margins simultaneously produce the most informative gradients and highest spatial alignment.

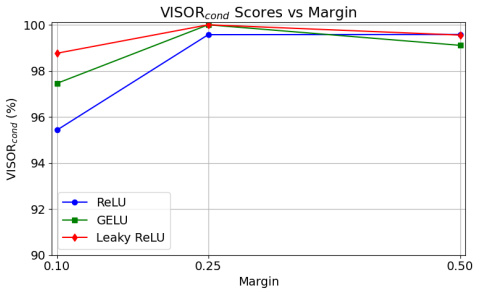
These trends collectively indicate a trade-off; larger margins provide stronger spatial correction signals but risk degrading object presence. Thus, the optimal margin depends on balancing spatial precision and object retention. Overall, across these losses, a margin of 0.25 yields a strong performance balancing well object separation and spatial flexibility. This motivates our choice for using m=0.25 in our experiments.



(a) Effect of m on OA for different losses.



(b) Effect of m on VISOR$_{\text{uncond}}$ for different losses.

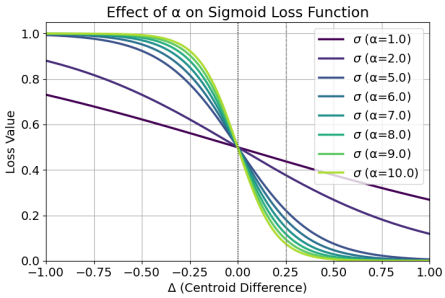


(c) Effect of m on VISOR$_{\text{cond}}$ for different losses.

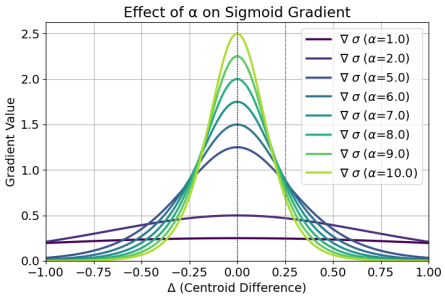
Figure 4: Impact of different margin values m on the Object Accuracy (OA), VISOR$_{\text{uncond}}$, and VISOR$_{\text{cond}}$ scores. Each figure shows the performance on one metric for the different loss function across margin values $[0.1, 0.25, 0.5]$. As m increases, ReLU losses shows more improvement. GELU benfits from a margin $m = 0.25$ for both object accuracy and spatial alignment. While, higher m reduces the object accuracy for Leaky ReLU. The results on VISOR$_{\text{cond}}$ show that all three losses benefit from an $m = 0.25$ for spatial alignment.

Role of the Scaling Parameter α. The parameter α controls the steepness of the loss and the magnitude of the gradient curves. Increasing α, sharpens the transition around $\Delta \approx m$, effectively amplifying the gradient magnitude, as illustrated in Figure 5. This enhances the model's ability to perform faster corrections when objects are near the threshold of spatial alignment. However, this effect is not beneficial for all spatial loss function in our analysis. Experimental results for ReLU, GELU and Leaky ReLU indicate that overly steep gradients may lead to overshooting in updating the predicted noise, $\alpha = 5$ in Figure 6a, Figure 6b and Figure 6c. The parameter α scales the strength of the spatial loss gradient applied during early timesteps. While a larger α provides a stronger corrective signal, benefiting spatial alignment, it can also degrade object retention by pushing centroids toward the image boundaries, causing partial or complete object omission. As shown in Figure 6a, for ReLU, GELU, and Leaky ReLU losses, object accuracy increases with α up to a critical point, after which further increases harm object presence. In contrast, spatial alignment continues to improve with α until it saturates. Thus, the optimal α lies at the trade-off between high object accuracy and strong spatial alignment. We find that in practice, α=2 yields the highest performance for GELU and Leaky ReLU, while α=0.75 gives is optimal for ReLU.

In contrast, the Sigmoid function exhibits a different trend. As shown in Figure 5a, increasing α leads to steeper slope enriching the penalty range applied to the model, while at $\alpha = 1$ Sigmoid almost



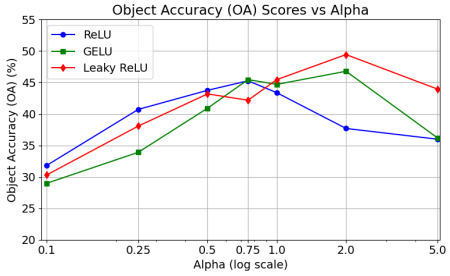
(a) Effect of α on the Sigmoid loss function.



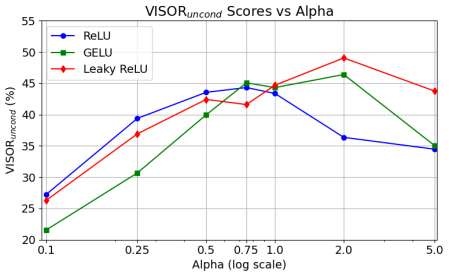
(b) Effect of α on the gradient of the Sigmoid loss.

Figure 5: Effect of scaling parameter α on the Sigmoid loss (a) and its gradient (b). Larger α increases steepness around $\Delta = 0$, resulting in stronger gradients and improved alignment. This highlights the distinct behavior of Sigmoid compared to piecewise and saturating losses.

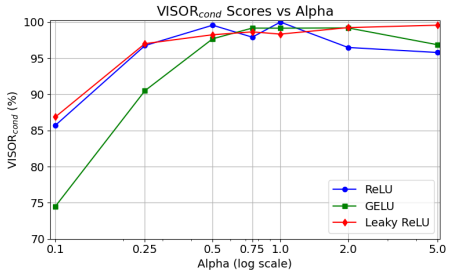
resembles a linear function in the interval $\Delta \in [-1, 1]$ limiting the correction ability of the guidance signal. That results in a very narrow range of loss values. Since the Sigmoid behaves differently than the other losses, we ablate higher α values for Sigmoid to make it better suited for the spatial alignment problem. The results displayed in Figure 6d show that α=8 is optimal. Increasing the parameter α consistently strengthens the guidance signal from the Sigmoid loss. This is expected, since even with a larger α, the Sigmoid still saturates at large deviations and produces its steepest gradients near zero. Consequently, it corrects small misalignments improving spatial alignment, while leaving well-placed objects unaffected, thereby preserving object accuracy. As a result, we observe a near-monotonic improvement in both object accuracy and spatial alignment as α increases.



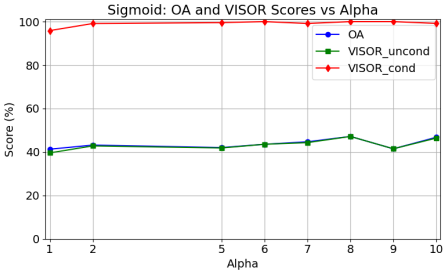
(a) Effect of α on OA for different losses.



(b) Effect of α on VISOR$_{\text{uncond}}$ for different losses.



(c) Effect of α on VISOR$_{\text{cond}}$ for different losses.



(d) Effect of α on the Sigmoid loss.

Figure 6: OA and VISOR scores for all spatial losses across different α values. Performance peaks at intermediate α for ReLU, GELU, Leaky ReLU. Extremely large α values result in diminished object accuracy and less effective spatial correction due to gradient overshooting. The results on Sigmoid are depicted in one plot as that loss function is fundamentally different from the others and requires a different range of α values for a meaningful experiment. Sigmoid consistently improves with increasing α, benefiting from a smoother gradient and more expressive penalty range.

Table 6: Average loss reduction across attention layers in the UNet decoder of SD 2.1. The loss reduction is computed as the difference between the values attained by the spatial loss function at the first and last denoising step. This is averaged over 20 prompts for each spatial loss function and normalized per loss function with 1.0 indicating maximum reduction. $\mathcal{A}_{i,j}$ denotes the j-th attention layer of the i-th decoder block of the UNet. The highlighted rows correspond to the last decoder block and show the least reduction across the losses.

Layer	ReLU	Leaky ReLU	GELU	Sigmoid
$\mathcal{A}_{1,1}$	0.989	0.870	0.739	0.658
$\mathcal{A}_{1,2}$	1.000	0.962	0.801	0.914
$\mathcal{A}_{1,3}$	1.000	0.969	0.808	0.840
$\mathcal{A}_{2,1}$	0.982	0.860	0.749	0.672
$\mathcal{A}_{2,2}$	0.971	0.898	0.782	0.703
$\mathcal{A}_{2,3}$	0.987	0.862	0.652	0.546
$\mathcal{A}_{3,1}$	0.302	0.154	0.098	0.077
$\mathcal{A}_{3,2}$	0.147	0.133	0.059	0.068
$\mathcal{A}_{3,3}$	0.200	0.103	0.057	0.038

Empirical Analysis of Attention Layer Effectiveness. We examine the impact of the attention maps on spatial guidance by measuring the average loss reduction during denoising. The loss reduction refers to the difference of the loss at the start and end of our intervention in the sampling process. This way we get a sense of whether the loss is decreasing according to its formulation, which would indicate that it produced a reliable guidance signal. To refer to the attention maps we use the notation $\mathcal{A}_{i,j}$ where j is attention layer of the i-th decoder block of the UNet. Our findings show that attention maps from earlier decode blocks, $\mathcal{A}_{1,j}$ and $\mathcal{A}_{2,j}$, provide informative gradients and reduce the spatial loss over time. Whereas, later attention layers result in noisy attention maps, $\mathcal{A}_{3,j}$, and produce unreliable centroid estimations resulting in a potentially harmful signal. We attribute this to the hypothesis that high-resolution attention layers do not focus on object-specific information but rather capture more local features such as textures and fine-details.

To quantify this, we computed the loss reduction between the first and last denoising steps of applying the method for each attention layer over 20 VISOR [21] prompts. Attention maps $\mathcal{A}_{1,j}$ and $\mathcal{A}_{2,j}$, show consistent and meaningful loss reduction which directly influences the ability of the model to generate spatially cognizant images. On the other hand, $\mathcal{A}_{3,j}$ attention maps result in negligible reductions, indicating a limited guidance ability. We further analyze this hypothesis in Appendix E.

5.6 Qualitative results

Figure 7 sheds light on the actual generation abilities of `InfSplign` through comparing with the baseline Stable Diffusion model [65].

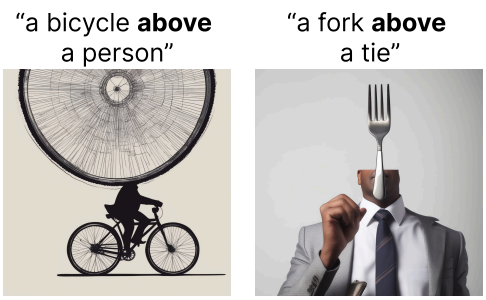
Figure 7: Qualitative comparison with Stable Diffusion across VISOR prompts. We show 2 examples for the baseline SDXL [65], top row, and for our method, bottom row. Not only does the baseline fail to recognize the meaning of the spatial relationship from the prompt, but also it fails to generate both objects in unnatural object combinations. `InfSplign` significantly improves upon these limitations and synthesizes spatially aware images even in atypical object settings.

The spatial losses successfully overcome the limitations in the baseline - incorrect spatial placement and single object generation. The base model generated results where objects vary in spatial location which implies the ignorance of the spatial relationship mentioned in the prompt. The examples with spatial relations *"left of"* and *"right of"* generate two misaligned objects, whereas with *"above"* and *"below"* the model struggles to generate both objects in one image. We attribute this mostly to the unnatural combination of the objects in the prompt "a bench **above** a cake" [34]. Hence, the best that SD can do is to only generate object combinations that it has seen during training - it is more likely to produce a "cat" together with a "motorcycle" than a "cake" and a "bench" in one image. The missing object problem can also be explained with some of the insights discussed in A&E [34], namely that it can be suppressed, mixed with the other object, entangled in the representation of the other object or subtly blended in the image.

`InfSplign` successfully addresses the constraints Stable Diffusion faces by introducing well-crafted spatial losses which produce a meaningful signal used to guide the underlying diffusion model through the denoising process. Our method successfully interpreted the spatial information and generated both object at locations in accordance with the spatial relationship given in the prompt. In the rare object combination case, "a bench **above** a cake", `InfSplign` successfully disentangled the concept of the "bench" from the attention map information and that object reappears in the generated image. The model cleverly figures out that the bench object cannot realistically be placed on top of a cake, so it generates it as a cake topper and positions it correctly above the cake.

We aim to critically assess the performance of our model, which is why we provide some insights into the failure cases and weaknesses of `InfSplign`. The limitations of our method are mostly caused by the underlying model's generation capabilities.

Semantic Bias from Training Data. Trained on real-world images from the LAION dataset [66], Stable Diffusion inherits common sense configurations of objects, making it susceptible to certain biases. Some object combinations appear more often together than individually and this poses some challenges to the model's ability to identify and generate them independently. Figure 8a shows typical examples of this pitfall. The model has often seen men wearing a tie and people riding bicycles. So, prompts requiring the "bike" to appear *"above"* the "person" or isolating "tie" from its usual context, lead to failures in concept disentanglement. The issues are linked to the cross-attention maps where text and image interact with each other, as discussed in [13, 15, 64]. In spite of these difficulties, `InfSplign` produces spatially consistent images to the best of its abilities.



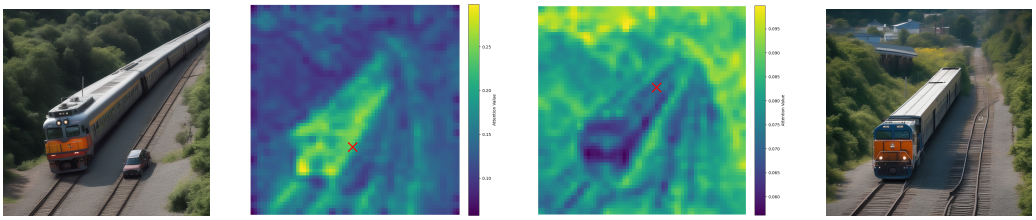
(a) SD suffers from prior distribution bias - it struggles to separate objects when it has learned that they appear together.

(b) 2D spatial relationships can be confused for 3D spatial configurations due to lack of depth precision in `InfSplign`.

Figure 8: Limitations due to model priors and spatial ambiguity.

Limitations of 2D Spatial Reasoning. While inspecting the results qualitatively, we noticed that spatial relationships like *"above"* and *"below"* are sometimes confused with *"behind"* and *"in front of"*. This arises because our method operates in 2D space and does not enforce 3D spatial constraints. Thus, there is no knowledge about depth or perspective when we apply the spatial losses to correct any misaligned object. Figure 8b illustrates this behaviour. In both examples the spatial relationship is respected in the 2D image space, the "keyboard" is *"below"* the "boat" and the "person" is *"above"* the "keyboard", but not in 3D space. The evaluation benchmarks consider such outputs spatially correct, as the metrics assume 2D spatial interactions.

Weak Attention Hinders Object Presence. A key limitation in our work is its reliance on the accurate object representation in attention maps. This affects object accuracy which is a prerequisite for spatial alignment. Figure 9 shows the result of SD on the prompt "a train **below** a car" as well as the outcome of `InfSplign` and the attention maps for the corresponding object tokens.



(a) Result from the baseline SD.

(b) `InfSplign` attention map from token "train".

(c) `InfSplign` attention map from token "car".

(d) Result from `InfSplign`.

Figure 9: `InfSplign` fails to generate the "car" object due to weak attention activations for the prompt "a train **below** a car". The estimated location of the object centroids is shown with a red cross. The scale of the values in the attention map is depicted on the right.

The baseline, Figure 9a, generates the "car" object, while `InfSplign`, Figure 9d, fails to do so. The "train" token's attention map, Figure 9b, is strongly localized which indicates confident object placement. However, Figure 9c shows a weak and dispersed attention map which fails to provide a clear spatial signal. Due to the weak attention to the "car" token, this object is omitted from the final image. This illustrates a failure case of `InfSplign` as spatial guidance is limited by weak object localization in the attention maps. It also highlights how the effectiveness of our method depends on the baseline model's ability to represent both objects in the attention maps.

Object Mixing. Figure 10 depicts another limitation of the base model, object mixing, where the two objects "zebra" and "motorcycle" are fused into one indicating the difficulty the model is experiencing in disentangling individual concepts from the compositional prompt. Figure 10a illustrates the issue of object mixing - features of the "zebra" appear on the "motorcycle" and there is no distinct "zebra" object. Although `InfSplign` suffers from the same limitation, Figure 10b shows reduced entanglement between the objects and generates a "zebra" according to the spatial relationship *"to the right of"*. This typically arises when the attention maps of the different objects highlight regions that are overlapping in the final image.

(a) Result from SD.

(b) Result from `InfSplign`.

Figure 10: Limitations of the base SD model for object mixing for the prompt "a zebra **to the right of** a motorcycle".

An overview of the effect of the spatial loss on the attention maps is presented in Figure 11. It can be seen that at timestep T, when the generation process begins, the attention maps are noisy and the centroids are estimated in the center of the image. As the denoising process is unrolled, the centroids start slowly moving towards the designated regions according to the spatial relationship specified in the prompt. It is clear that at the start the objects are mixed into one, but at the later timesteps it can be noticed that the foot of the zebra is formed which indicates some improvement in the ability to disentangle the "zebra" from the "motorcycle". The regions in the attention maps attended by

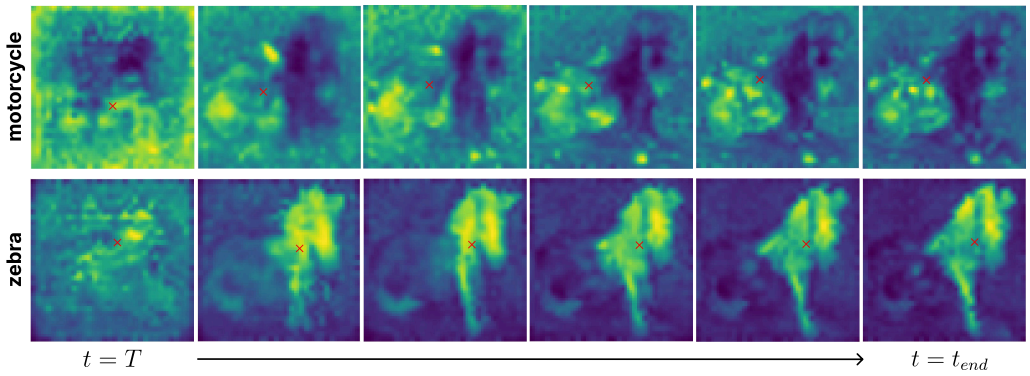


Figure 11: `InfSplign` reduces concept entanglement for the prompt "a zebra **to the right of** a motorcycle". The attention maps for both objects with the estimated centroid predictions, marked as crosses, are shown for individual timesteps from the beginning till the end of the spatial loss intervention part of the denoising process. The top row corresponds to the attention maps of the "motorcycle" token, while the bottom row shows them for the "zebra" token.

the object tokens become more focused and have less overlap towards the end of the spatial loss intervention. This progression shows that although not entirely, `InfSplign` separates the objects and improves spatial alignment by refining the attention maps over time according to the spatial guidance signal.

6 Conclusion

In this work, we address the problem of spatial understanding in text-to-image diffusion models, a key limitation that hinders accurate compositional generation. We present `InfSplign`, a simple yet effective inference-time approach that enhances spatial alignment without retraining or supervision. By applying a spatial loss during sampling and leveraging attention maps to infer object locations, `InfSplign` encourages adherence to spatial relations described in the prompt. It operates without altering model weights or relying on extra inputs such as bounding boxes or segmentation maps. `InfSplign` is fully modular and can be applied to any existing diffusion model. Despite its simplicity, it achieves competitive or superior performance compared to fine-tuning-based approaches, highlighting the potential of inference-time optimization for controllable generation. Our method also outperforms prior inference-time strategies targeting spatial consistency, demonstrating its effectiveness in improving spatial alignment.

Limitations. The results of our method vary depending on the choice of loss function and evaluation benchmark, largely due to differences in dataset size and object categories. Each benchmark presents distinct challenges, and the base diffusion model often struggles to generate uncommon object combinations. For instance, VISOR includes a wide range of object pairs, many of which rarely co-occur in natural data. While this setup is valuable for testing generalization in spatial reasoning, it also makes the task more difficult, particularly when one or both objects fail to appear in the generated image. In such cases, `InfSplign` cannot effectively correct the spatial layout, as the foundation (object presence) is missing. As a result, object accuracy becomes the limiting factor in our pipeline.

Future Works. `InfSplign` demonstrates strong performance across the three spatial benchmarks but nevertheless, STORM [53], another inference-time method, achieves higher scores on VISOR with SD 1.4. We attribute that to the higher object accuracy they report due to applying the iterative refinement trick from A&E [34] to increase the attention activations of missing object tokens. In our work, we focused solely on enforcing spatial alignment without trying to explicitly enhance the object presence. This is a potential future direction for boosting object accuracy without sacrificing the achieved spatial reasoning capabilities. Furthermore, we uncover the need for an evaluation metric for 3D spatial relationships so the scope of spatial understanding in T2I generation can expand further.

References

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

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

[3] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23 (47):1–33, 2022.

[4] Prafulla Dhariwal and Alex Nichol. Diffusion Models Beat GANs on Image Synthesis. *Advances in Neural Information Processing Systems*, 11:8780–8794, 5 2021. ISSN 10495258. URL `https://arxiv.org/abs/2105.05233v4`.

[5] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[6] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[7] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=di52zR8xgf`.

[8] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

[9] Gaoyang Zhang, Bingtao Fu, Qingnan Fan, Qi Zhang, Runxing Liu, Hong Gu, Huaqi Zhang, and Xinguo Liu. Compass: Enhancing spatial understanding in text-to-image diffusion models. *arXiv preprint arXiv:2412.13195*, 2024.

[10] Chao Wang, Giulio Franzese, Alessandro Finamore, Massimo Gallo, and Pietro Michiardi. Information theoretic text-to-image alignment. In *Proceedings of the International Conference on Learning Representations*, 2025.

[11] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465, June 2024.

[12] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A Comprehensive Benchmark for Open-world Compositional Text-to-image Generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 12 2023.

[13] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=PUIqjT4rzq7`.

[14] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–11, 2023.

[15] Dave Epstein, Allan Jabri, Ben Poole, Alexei Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 16222–16239, 2023.

[16] Wenqiang Sun, Teng Li, Zehong Lin, and Jun Zhang. Spatial-Aware Latent Initialization for Controllable Image Generation. 1 2024. URL `https://arxiv.org/abs/2401.16157v1`.

[17] Azade Farshad, Yousef Yeganeh, Yu Chi, Chengzhi Shen, Böjrn Ommer, and Nassir Navab. Scenegenie: Scene graph guided diffusion models for image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 88–98, 2023.

[18] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7932–7942, 2024.

[19] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded Diffusion: Enhancing Prompt Understanding of Text-to-Image Diffusion Models with Large Language Models. URL https://llm-grounded-diffusion.github.io.

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

[21] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022.

[22] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022.

[23] Agneet Chatterjee, Gabriela Ben Stan Melech, Estelle Aflalo, Sayak Paul, Dhruba Ghosh, Tejas Gokhale, Ludwig Schmidt, Hannaneh Hajishirzi, Vasudev Lal, Chitta Baral, and Yezhou Yang. Getting it Right: Improving Spatial Consistency in Text-to-Image Models. 4 2024. URL https://arxiv.org/abs/2404.01197v2.

[24] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 78723–78747, 2023.

[25] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[26] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[27] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[28] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019.

[29] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[30] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

[31] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[33] Jiachen Li, Jitesh Jain, and Humphrey Shi. Matting anything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1775–1785, 2024.

[34] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM transactions on Graphics (TOG)*, 42(4): 1–10, 2023.

[35] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=_CDixzkzeyb.

[36] Yumeng Li, Margret Keuper, Dan Zhang, and Anna Khoreva. Divide In *BMVC*, page 366, 2023. URL http://proceedings.bmvc2023.org/366/.

[37] Xianghao Kong, Ollie Liu, Han Li, Dani Yogatama, and Greg Ver Steeg. Interpretable diffusion via information decomposition. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=X6tNkN6ate.

[38] Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Boqing Gong, Cho-Jui Hsieh, and Minhao Cheng. The crystal ball hypothesis in diffusion models: Anticipating object positions from initial noise. *arXiv preprint arXiv:2406.01970*, 2024.

[39] Xiefan Guo, Jinlin Liu, Miaomiao Cui, Jiankai Li, Hongyu Yang, and Di Huang. Initno: Boosting text-to-image diffusion models via initial noise optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9380–9389, 2024.

[40] Agneet Chatterjee, Gabriela Ben Melech Stan, Estelle Aflalo, Sayak Paul, Dhruba Ghosh, Tejas Gokhale, Ludwig Schmidt, Hannaneh Hajishirzi, Vasudev Lal, Chitta Baral, et al. Getting it right: Improving spatial consistency in text-to-image models. In *European Conference on Computer Vision*, pages 204–222. Springer, 2024.

[41] Yinan Zhang, Eric Tzeng, Yilun Du, and Dmitry Kislyuk. Large-scale reinforcement learning for diffusion models. In *European Conference on Computer Vision*, pages 1–17. Springer, 2024.

[42] Xinyan Chen, Jiaxin Ge, Tianjun Zhang, Jiaming Liu, and Shanghang Zhang. Learning from mistakes: Iterative prompt relabeling for text-to-image diffusion model training. *arXiv preprint arXiv:2312.16204*, 2023.

[43] Xinchen Zhang, Ling Yang, Guohao Li, Yaqi Cai, Jiake Xie, Yong Tang, Yujiu Yang, Mengdi Wang, and Bin Cui. Itercomp: Iterative composition-aware feedback learning from model gallery for text-to-image generation. *arXiv preprint arXiv:2410.07171*, 2024.

[44] Wenqiang Sun, Teng Li, Zehong Lin, and Jun Zhang. Spatial-aware latent initialization for controllable image generation. *arXiv preprint arXiv:2401.16157*, 2024.

[45] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 5343–5353, 2024.

[46] Biao Gong, Siteng Huang, Yutong Feng, Shiwei Zhang, Yuyuan Li, and Yu Liu. Check locate rectify: A training-free layout calibration system for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6624–6634, 2024.

[47] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023.

[48] Phillip Y Lee and Minhyuk Sung. Reground: Improving textual and spatial grounding at no cost. In *European Conference on Computer Vision*, pages 275–292. Springer, 2024.

[49] Weili Nie, Sifei Liu, Morteza Mardani, Chao Liu, Benjamin Eckart, and Arash Vahdat. Compositional text-to-image generation with dense blob representations. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=dMOhgHNYAf`.

[50] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL `https://openreview.net/forum?id=hFALpTb4fR`. Featured Certification.

[51] Mohammad Mahdi Derakhshani, Menglin Xia, Harkirat Behl, Cees GM Snoek, and Victor Rühle. Unlocking spatial comprehension in text-to-image diffusion models. *arXiv preprint arXiv:2311.17937*, 2023.

[52] Agneet Chatterjee, Yiran Luo, Tejas Gokhale, Yezhou Yang, and Chitta Baral. Revision: Rendering tools enable spatial fidelity in vision-language models. In *European Conference on Computer Vision*, pages 339–357. Springer, 2024.

[53] Woojung Han, Yeonkyung Lee, Chanyoung Kim, Kwanghyun Park, and Seong Jae Hwang. Spatial transport optimization by repositioning attention map for training-free text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[54] Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2008.

[55] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 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, 2015.

[56] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding Conditional Control to Text-to-Image Diffusion Models. *Proceedings of the IEEE International Conference on Computer Vision*, pages 3813–3824, 2 2023. ISSN 15505499. doi: 10.1109/ICCV51070.2023.00355. URL `https://arxiv.org/abs/2302.05543v3`.

[57] Sicheng Mo, Fangzhou Mu, Kuan Heng Lin, Yanli Liu, Bochen Guan, Yin Li, and Bolei Zhou. FreeControl: Training-Free Spatial Control of Any Text-to-Image Diffusion Model with Any Condition. 12 2023. URL `https://arxiv.org/abs/2312.07536v1`.

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2022-June:10674–10685, 12 2021. ISSN 10636919. doi: 10.1109/CVPR52688.2022.01042. URL `https://arxiv.org/abs/2112.10752v2`.

[59] Jonathan Ho, Google Research, and Tim Salimans. Classifier-Free Diffusion Guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 12 2021.

[60] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[61] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014.

[62] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

[63] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[64] Qiucheng Wu, Yujian Liu, Handong Zhao, Ajinkya Kale, Trung Bui, Tong Yu, Zhe Lin, Yang Zhang, and Shiyu Chang. Uncovering the disentanglement capability in text-to-image diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1900–1910, 2023.

[65] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[66] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

[67] Qiucheng Wu, Yujian Liu, Handong Zhao, Trung Bui, Zhe Lin, Yang Zhang, and Shiyu Chang. Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7766–7776, 2023.

A Benchmarks and metrics

A.1 VISOR

The authors use the 80 object categories from the MS-COCO dataset and define unordered pairs of all objects, resulting in 6320 object combinations. Each object pair is matching with all 4 spatial relationships - "to the left of", "to the right of", "above" and "below". This gives us a total of 25280 spatial prompts. The VISOR dataset includes 6320 more prompts for each of the object pairs and the concept conjunction "and". Finally, it includes 80 more one-object prompts for all 80 COCO categories. In total, this results in 31680 text prompts. Since the benchmark defines visor 4, it requires 4 generated images per prompt. All VISOR images are generated with the seed 42, ensuring that the images across prompts start from the same initial noise. We followed these guidelines in the evaluation of InfSplign.

The VISOR benchmark uses the OWL-ViT object detector to localize objects classes which are matched to the objects from the prompt. Then using a simple set of rules it predicts the spatial relationship and compares it with the ground truth relationship to determine if the image aligns with the spatial information in the prompt. The benchmark defines the following set of metrics: OA (object accuracy) indicating the number of images in which both objects were generated and detected, $\text{VISOR}_{\text{cond}}$ estimates the conditional probability of how many of the images adhere to the spatial relationship given that both objects are generated correctly, VISOR (or $\text{VISOR}_{\text{uncond}}$) gives a ratio of the number of images generated with a correct spatial relationship and the two objects generated correctly out of all generated images and VISOR_1 to VISOR_N define the number of images out of N that have VISOR=1. VISOR chooses N=4. All VISOR metrics are reported as a percentage.

A.2 T2I-CompBench

T2I-CompBench [24] is a benchmark for compositional T2I generation. It looks at attribute binding, object relationships and complex compositions. For our research, we are interested only in the 2D spatial relationships subcategory. The benchmark offers 1000 spatial prompts, but only 300 of them are used for testing since they present a fine-tuning approach and use the remaining 700 prompts for training. T2I-CompBench includes 7 spatial relationships. The first 4 relationships are directional and consistent with VISOR, while the last 3 are relative, requiring close proximity of objects - *"on the left of"*, "on the right of", "on the top of", "on the bottom of", "on the side of", "next to", "near". The objects are chosen from 3 categories: persons, animals and objects. They generate 10 images per prompt with 10 different random seeds, 42 to 51, again ensuring that across prompts, the images start with the same initial noise.

T2I-CompBench uses the UniDet object detector to predict the location of each generated object. The centers of the objects are calculated from the detected bounding boxed. The metric considers the difference between the x and y coordinates of the estimated centers of the objects, which estimates is the spatial relationship is respected, their absolute difference, to measure in which axis the objects are further apart, and the intersection-over-union (IoU), to prevent object overlap. To address the correctness of the 3 relative spatial relationships, the metric compares the difference between the two objects' centroids both in the x and y axis to a threshold. If the difference in either axis is smaller than the fixed threshold, the output is 1. Otherwise, the positional score is set to the ratio of the threshold over the highest coordinate difference. For the 4 directional spatial relationships, the metric considers all rules mentioned above. If all of them are satisfied, the positional score is set to 1, otherwise, - to the ratio of the IoU threshold over the estimated IoU (which is higher). The final score is determined by the sum of the 25% of the confidence score for each of the two detected objects and 50% of the positional score. If just one object is detected, the score is 25% of that object's confidence score from the object detector. In the end, if the final predicted score for the metric is below 50%, the score is set to 0, as the metric did not have a high confidence for the score.

GenEval The GenEval benchmark [25] evaluates compositionality of T2I generation, similar to T2I-CompBench. The benchmark consists of 100 spatial prompts. The object categories used are also from the MS-COCO dataset. The spatial relationships are the following: "left of", "right of", "above", "below". The prompts are in the form "a photo of obj_A rel obj_B". The benchmark generates 4 images per prompt, with seed 42 ensuring each image across the prompts starts with the same initial noise.

The object detector used for detecting the center of the objects is Mask2Former. The benchmark calculates the difference between the x and y coordinates (offset) of the two centroids and the width and height of each of the objects. Furthermore, it adjusts the offset between the two objects by accounting for the their sizes through a threshold. This threshold, can be interpreted as the acceptable IoU ratio, defines how much overlap between the objects can be ignored. If the objects are larger, the relationship should tolerate more overlap between them. Then, the offset between the centroids is updated and normalized in the range $[-1, 1]$ for x and y. If the normalized updated offset $\in [-0.5, 0.5]$, it means that the objects are rather aligned by the two axis and no relationship can be inferred from their placement. Otherwise, one or two spatial relationships can be determined - the four given relationships and their combinations, e.g. left and above. The predicted relationships are compared to the ground truth relationship and if there is a match, the image is considered as successfully spatially aligned. The final spatial score is computed as the average over all test images.

B Ablation: Effectiveness of Additional Loss Terms.

Table 7 evaluates the impact of adding orthogonal loss terms to the primary spatial loss across different activation functions using SD 2.1. Compared to the results on SD 1.4, shown in Table 5, the improvements from the orthogonal losses on SD 2.1 are less pronounced. This can be attributed to the stronger baseline performance of SD 2.1 which already achieves high object presence and spatial consistency.

The $\text{VISOR}_{\text{cond}}$ of ReLU, Leaky ReLU and GELU benefit slightly from the ℓ_2-norm variant of $\mathcal{L}_{\text{orthogonal}}$ but there no gains in the $\text{VISOR}_{\text{uncond}}$ and OA metrics. These findings confirm that the

orthogonal term is most beneficial in cases where object preservation is weaker (e.g., SD 1.4) compared to when the model already generates well separated objects.

Table 7: Ablation results of InfSplign on SD 2.1 for the different terms of the spatial losses, $\mathcal{L}_{\text{spatial}}$ and $\mathcal{L}_{\text{orthogonal}}$. Scores are reported on the OA, VISOR$_{\text{uncond}}$, and VISOR$_{\text{cond}}$ metrics.

Model	**OA**	**VISOR$_{\text{uncond}}$**	**VISOR$_{\text{cond}}$**
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}}$)	45.265	44.318	97.908
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	40.152	39.962	99.528
InfSplign (ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	43.750	42.803	97.835
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}}$)	49.432	49.053	99.234
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	45.455	45.265	99.583
InfSplign (Leaky ReLU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	46.212	45.644	98.770
InfSplign (GELU, $\mathcal{L}_{\text{spatial}}$)	46.780	46.402	99.190
InfSplign (GELU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	43.750	43.561	99.567
InfSplign (GELU, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	44.318	43.939	99.145
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}}$)	47.159	47.159	100.000
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}} + \|\delta\|_1^1$)	46.212	46.023	99.590
InfSplign (Sigmoid, $\mathcal{L}_{\text{spatial}} + \|\delta\|_2^2$)	47.159	46.970	99.598

C Ablation: Loss Intervention

Figure 12 illustrates how the timing of spatial loss application affects the model's ability to control object placement. t_{start} indicats at which timestep we begin intervening in the denoising process. When the spatial loss is introduced at the beginning of the denoising process, the model successfully arranges objects according to the spatial relationship in the prompt. However, when the guidance starts at later timesteps (e.g., $t_{\text{start}} = 50$ or 100), the object positions remain fixed and cannot be changed. This behavior highlights that diffusion models establish coarse spatial structure early in the generation process, and later steps mainly focus on refining texture and appearance [60, 67]. Consequently, spatial interventions must be applied early to influence object layout effectively.

We perform an ablation using the visor ablation subset, described in subsection 5.5, and the SD 2.1 base model. The results are presented in Table 8 We choose the interval $t_{start} = 0, t_{end} = 125$ for our main experiments, despite $t_{end} = 250$ yielding marginally higher scores. This decision balances performance with efficiency and interpretability. First, the gains from extending to $t = 250$ are small compared to the computational overhead of a longer guidance window. Second, most spatial structure in diffusion models is determined early in the denoising process and intervening later could harm the performance. Finally, restricting the spatial loss to earlier steps reduces the risk of interfering with image quality refinement that occurs in later stages, thus, preserving visual fidelity while still improving spatial alignment.

D Ablation: Slope of Leaky ReLU.

Applying the spatial loss function Leaky ReLU comes to address the limitation of ReLU. Our empirical analysis showed that ReLU reaches a spatial loss of 0 quite quickly. Once $\mathcal{L}_{\text{ReLU}} = 0$, the gradient (Equation 13) also becomes 0, thus no further updates are applied in the fixed number of timesteps of the spatial loss intervention. On the contrary, the GELU spatial loss shows promising improvements over ReLU. That we owe to the smooth curvature of GELU compared to the linearity of ReLU and to the fact that GELU keeps decreasing below 0, so the small negative values this function attains provide some guidance signal. Therefore, we see that the nature of Leaky ReLU might better fit the spatial problem setting since it introduces a negative slope instead of directly dropping to 0 for inputs $x < 0$. However, the choice of Leaky ReLU comes with setting the negative slope which is another hyperparameter we need to tune for. Since the function is "leaky", it is natural to explore small slopes but we also ablate over higher values of the negative slope. Table Table 9

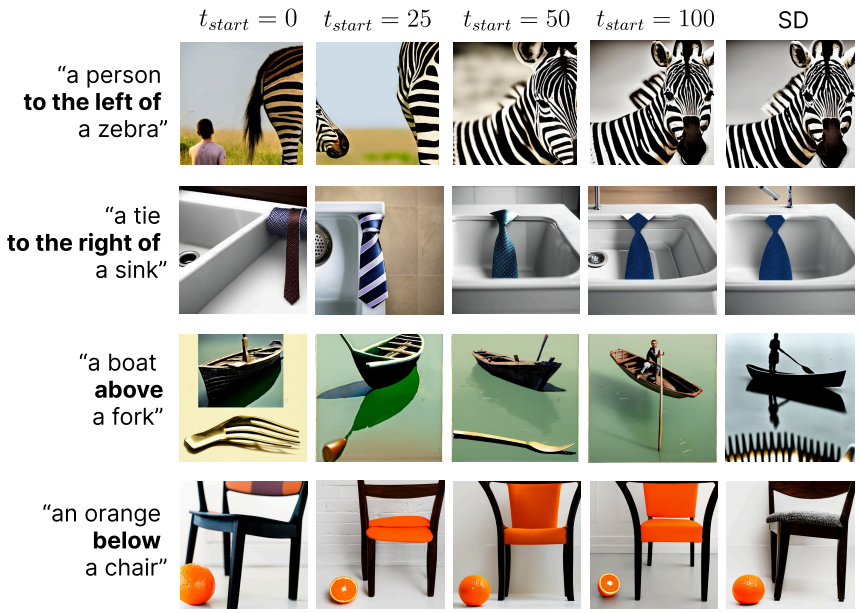


Figure 12: Effect of varying the spatial loss start time. Spatial guidance is most effective when applied early, confirming that object layout is determined in the early denoising steps. Late interventions fail to correct spatial misalignment.

shows that smaller negative slopes give better results on the VISOR ablation dataset. The ablation is performed using SD 2.1 using the 6 attention layers and $\alpha = 2$ that resulted in the best performing configuration.

E Ablation: Effectiveness of Attention Layers

The effectiveness of the spatial loss can be visualized by examining its behavior over the 125 timesteps during which InfSplign is applied. Figure 13, Figure 14, Figure 15, and Figure 16 show the average spatial loss values across timesteps for each of the four loss functions: ReLU, Leaky ReLU, GELU, and Sigmoid, respectively. The plots reveal that attention layers $\mathcal{A}_{1,\_}$ and $\mathcal{A}_{2,\_}$, corresponding to `up_blocks.1` and `up_blocks.2` from the UNet architecture and shown in the legend of the plots, decrease consistently. This suggests that these layers contain more structured and informative signals, making them better suited for spatial supervision. In contrast, attention layers $\mathcal{A}_{3,\_}$ show flatter or noisier behavior and do not contribute towards loss reduction. This indicates that the guidance signal coming from these layers is not informative and might harm the spatial generation ability. These insights motivate our design choice to restrict the guidance loss to a subset of layers where only spatial gradients can be extracted reliably.

Table 8: Performance of InfSplign on SD 2.1 with varying intervals of spatial loss application. Each row shows results for a different pair of starting and ending timesteps. The best performance is achieved when the spatial loss is applied early in the denoising process.

Interval of Spatial Loss	**OA**	**VISOR$_{\text{uncond}}$**	**VISOR$_{\text{cond}}$**
t_{start}=0, t_{end}=50	38.826	37.879	97.561
t_{start}=0, t_{end}=100	45.455	45.076	99.167
t_{start}=0, t_{end}=125	49.432	49.242	99.617
t_{start}=0, t_{end}=250	49.621	49.432	99.618
t_{start}=0, t_{end}=500	49.053	49.053	100.000
t_{start}=25, t_{end}=50	36.174	34.280	94.764
t_{start}=25, t_{end}=100	41.667	41.098	98.636
t_{start}=25, t_{end}=125	45.833	44.886	97.934
t_{start}=25, t_{end}=250	49.621	48.864	98.473
t_{start}=25, t_{end}=500	49.242	49.053	99.615
t_{start}=50, t_{end}=100	36.174	33.712	93.194
t_{start}=50, t_{end}=125	43.561	41.856	96.087
t_{start}=50, t_{end}=250	44.886	44.318	98.734
t_{start}=50, t_{end}=500	46.780	46.023	98.381
t_{start}=100, t_{end}=125	29.924	23.106	77.215
t_{start}=100, t_{end}=250	39.773	37.121	93.333
t_{start}=100, t_{end}=500	43.182	40.152	92.982

Table 9: Object Accuracy (OA) and VISOR scores for different slope values of the leaky ReLU spatial loss

Slope	**OA**	**VISOR_uncond**	**VISOR_cond**
0.05	45.455	45.265	99.583
0.10	49.432	49.053	99.234
0.25	45.076	44.886	99.580
0.50	39.583	39.394	99.522

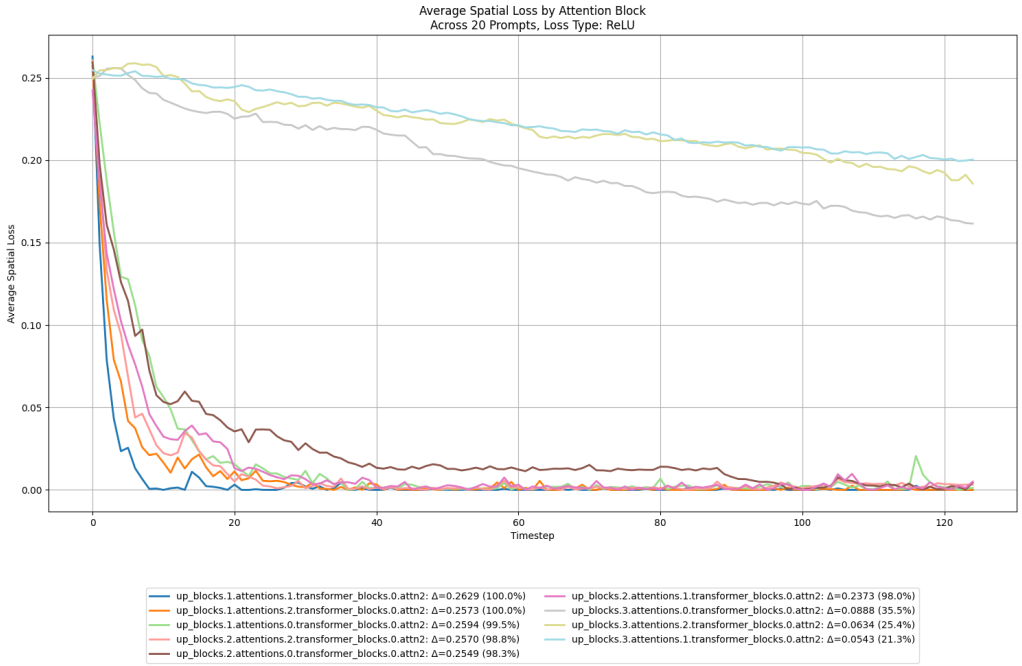


Figure 13: Average spatial ReLU loss by attention layer.

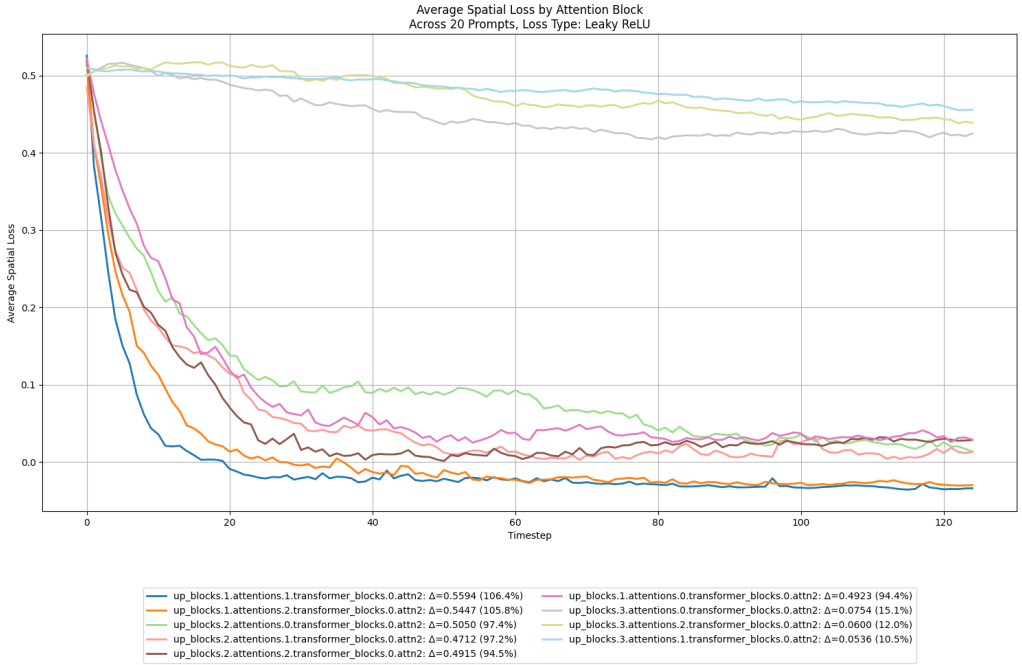


Figure 14: Average spatial Leaky ReLU loss by attention layer.

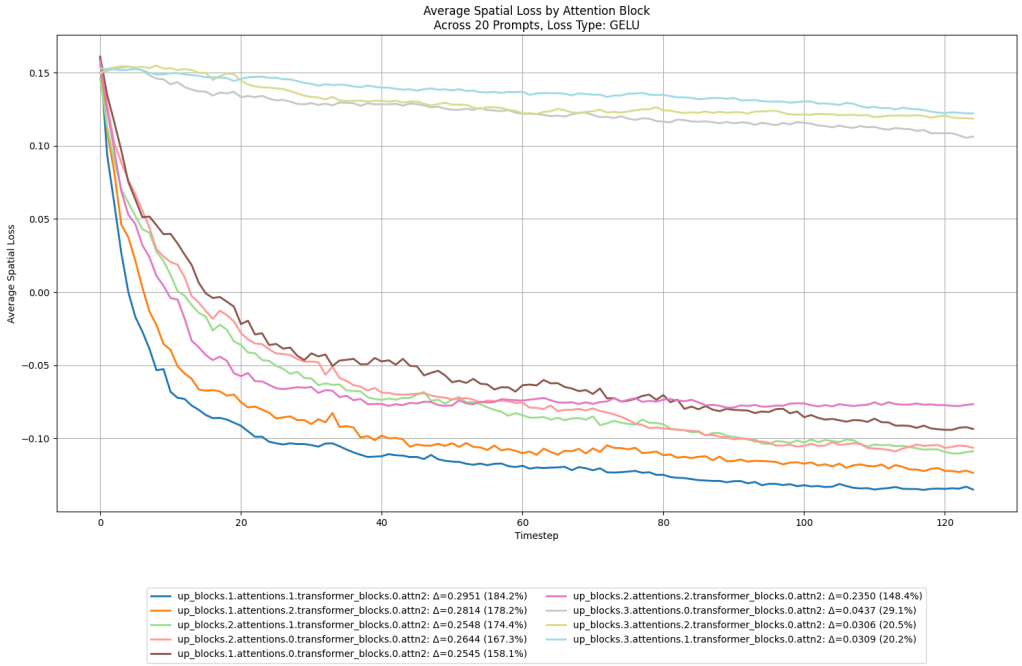


Figure 15: Average spatial GELU loss by attention layer.

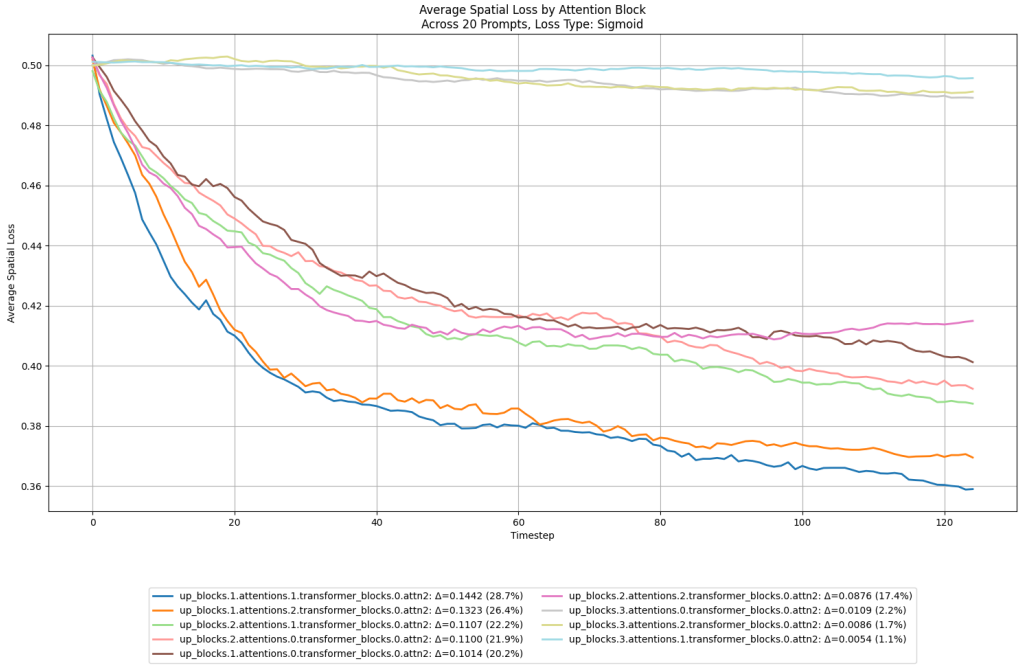


Figure 16: Average spatial Sigmoid loss by attention layer.

3

Generative AI and Diffusion Models

3.1. Overview of Generative Modeling

Generative modeling refers to the task of learning the underlying distribution of data such that one can sample novel data points that resemble the training distribution. This section will discuss the strengths and limitations of different classes of generative models - Generative Adversarial Networks (GANs) [17], Variational Autoencoders (VAEs) [28], and autoregressive models [65]. Each model takes a different approach to modeling high-dimensional probability distributions and learning effective mappings between latent and data spaces.

3.1.1. Generative Adversarial Networks (GANs)

Introduced by Goodfellow et al. [17], GANs model the data distribution implicitly using a two-player minimax game between a generator and a discriminator. The generator creates synthetic data, while the discriminator is tasked to distinguish between real and generated samples. The adversarial training objective encourages the generator to produce increasingly realistic outputs, ideally fooling the discriminator.

GANs achieve high-fidelity image synthesis, super-resolution and style transfer [25, 31]. Extensions such as DCGAN [47] and StyleGAN [27] have improved the architectural stability and sample diversity of this class of models. However, GANs are notoriously difficult to train due to issues such as vanishing gradients and non-convergence. Additionally, they do not provide a tractable likelihood function, limiting their applicability in tasks requiring probabilistic reasoning. Mode collapse, where the generator learns to produce only a subset of the data distribution, remains a significant challenge [55, 4].

3.1.2. Variational Autoencoders (VAEs)

VAEs [28] are latent variable generative models that learn a compressed representation of the data and can generate new data points from the learned distribution. A typical VAE architecture consists of an encoder network that maps input data x to a latent distribution $q(z \mid x)$ and a decoder which samples a data point from the latent space and reconstructs the input data $x \sim p(x \mid z)$. This process is illustrated in Figure 3.1.

The learning objective is to maximize the Evidence Lower Bound (ELBO) Equation 3.14, which balances reconstruction fidelity and regularization via a Kullback-Leibler (KL) divergence term. VAEs offer stable training and a probabilistic framework but often produce blurry images due to limitations in their decoder expressiveness and reliance on pixel-wise losses such as mean squared error [51].

Various extensions, such as beta-VAE [20], VQ-VAE [39], and hierarchical VAEs [60], have been proposed to improve sample quality and representation learning. However, VAEs generally lag behind GANs in visual fidelity, motivating hybrid models and alternative frameworks [29].

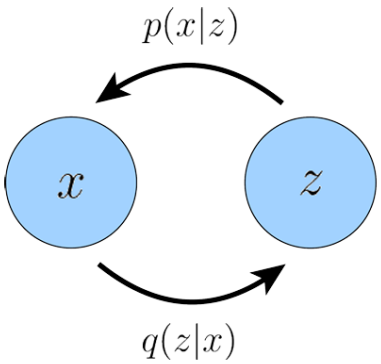


Figure 3.1: Illustration of a standard Variational Autoencoder (VAE) architecture. The encoder maps input data x to a distribution over latent variables z, and the decoder reconstructs the input from samples drawn from this latent distribution.

3.1.3. Autoregressive Models

Autoregressive models, such as PixelCNN [61] and Transformer-based architectures like GPT [48], model the joint distribution of data as a product of conditional probabilities:

$$p(x) = \prod_{i=1}^{D} p(x_i \mid x_{<i}) \tag{3.1}$$

These models generate one element at a time (e.g., pixel or token), conditioned on all previously generated elements.

Autoregressive models are simple to train using maximum likelihood estimation (MLE) and are commonly used in language modeling, such as text generation, translation and question answering [62]. Moreover, transformer-based architectures have improved upon the need for extensive computational resources when modeling long-range dependencies. Autoregressive models have also demonstrated state-of-the-art performance in image generation [61] and speech synthesis [40]. However, their primary limitation lies in the sequential sampling process, which results in slow generation, especially for high-resolution data [52].

3.1.4. Towards Diffusion Models

The limitations of GANs (instability and lack of likelihood), VAEs (low perceptual quality), and autoregressive models (slow sampling) have led to the emergence of diffusion models. These models aim to combine the strengths of likelihood-based training with the ability of generating high-quality samples. They gradually corrupt data samples with noise, making them blurry until reaching pure noise and then learning to recover the data from the noisy samples through iterative denoising. This approach allows for both stable optimization and detailed image synthesis.

Diffusion models provide explicit likelihoods, enable flexible conditioning and yield samples that often rival or surpass GANs in quality. Their foundations are closely tied to concepts from score matching, variational inference and hierarchical latent variable modeling. In the following sections, we provide a mathematical and algorithmic overview of diffusion models, heavily relying on the comprehensive overview in [36].

3.2. Foundations of Diffusion Models

Diffusion models have become widely popular and used by the community due to their strong performance in image synthesis, probabilistic interpretability and stability during training [53]. At a high level, diffusion models define a Markovian noising process that gradually ”destroys” data and learn to undo this process - to generate samples by denoising. The way this works is that the model learns the structure of real data by learning how it differs from noise.

Let $x_0 \sim p_{\text{data}}(x)$ denote a sample from the data distribution. The forward diffusion process defines a

sequence of latent variables $x_1, x_2, \ldots, x_T$ such that:

$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}) \tag{3.2}$$

where β_t is a small variance term that controls the noise added at each step t. Over many steps, the distribution $q(x_{1:T} \mid x_0)$ converges to pure Gaussian noise.

The reverse process is modeled using a neural network $\epsilon_\theta(x_t, t)$ that learns to predict the noise added at each timestep. The learned model approximates $p_\theta(x_{t-1} \mid x_t)$ using Equation 3.3:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_t) \tag{3.3}$$

where μ_θ is a function of x_t and the predicted noise.

3.2.1. Deriving the Evidence Lower Bound

As previously stated, the goal of any generative model is to learn the distribution underlying the training data such that it can synthesize new samples from the approximated distribution. The assumption is that the observed data is coming from an unseen latent distribution. That is why, we define the joint distribution modelling the latent variable and the observed data as $p(x, z)$. That is what we have to manipulate to recover the likelihood of the observed data $p(x)$. To do that, we will marginalize out the latent variable z:

$$p(x) = \int p(x, z) dz, \tag{3.4}$$

or we could apply the chain rule of probability:

$$p(x) = \frac{p(x, z)}{p(z \mid x)}. \tag{3.5}$$

We cannot simply compute and maximize the likelihood $p(x)$ because integrating over all possible latent variables z is intractable for complex models and we do not have access to a latent encoder $p(z \mid x)$. Since one of the approaches of generative modelling is to learn a model to maximize the likelihood $p(x)$ of the observed data x, instead we will do our analysis for the log-likelihood $\log p(x)$. The term **evidence** refers to the log-likelihood $\log p(x)$ of the observed data x. So, let us express the evidence using Equation 3.4.

$$\begin{aligned}
\log p(x) &= \log \int p(x, z)\, dz && (3.6)\\
&= \log \int \frac{p(x, z)\, q_\phi(z \mid x)}{q_\phi(z \mid x)}\, dz && (3.7)\\
&= \log \mathbb{E}_{q_\phi(z|x)} \left[\frac{p(x, z)}{q_\phi(z \mid x)} \right] && (3.8)\\
&\geq \mathbb{E}_{q_\phi(z|x)} \left[\log \frac{p(x, z)}{q_\phi(z \mid x)} \right], && (3.9)
\end{aligned}$$

where $q_\phi(z \mid x)$ is a model learned to estimate the true distribution over the latents z for each data point x.

In step Equation 3.8 we apply the definition of expectation, while in step Equation 3.9 we can apply Jensen's inequality. The result in step Equation 3.9 is equivalent to maximizing the lower bound of the evidence, aka. maximizing the Evidence Lower Bound (ELBO).

Maximizing the ELBO becomes the means for learning the latent distribution $p(z \mid x)$.

3.2.2. The Math behind Diffusion Models

Diffusion models can be viewed through the lens of hierarchical probabilistic modeling, specifically as a form of Markovian Variational Autoencoders (VAEs) with a fixed forward inference process and a

learned reverse generative model. The interpretation proposed in [36] provides a principled connection between the denoising process in diffusion models and classical variational inference.

In this view, the forward diffusion process $q(x_{1:T} \mid x_0)$ acts as a fixed inference model, gradually injecting Gaussian noise into the clean data sample x_0 across T timesteps. This forms a sequence of increasingly noisy latent variables:

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}), \tag{3.10}$$

where each transition is a Gaussian with time-dependent variance as described in Equation 3.2.

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}). \tag{3.11}$$

Rearranging Equation (3.11), we can express x_0 in terms of x_t and the added noise ϵ as:

$$x_0 = \frac{x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon}{\sqrt{\bar{\alpha}_t}}. \tag{3.12}$$

The reverse process $p_\theta(x_{0:T})$ serves as a generative decoder that reconstructs x_0 from pure noise $x_T \sim \mathcal{N}(0, \mathbf{I})$. This process is also modeled as a Markov chain, parameterized by a neural network:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} \mid x_t). \tag{3.13}$$

The training objective is maximizing the ELBO as explained in subsection 3.2.1:

$$\log p(x) \geq \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T} \mid x_0)} \right], \tag{3.14}$$

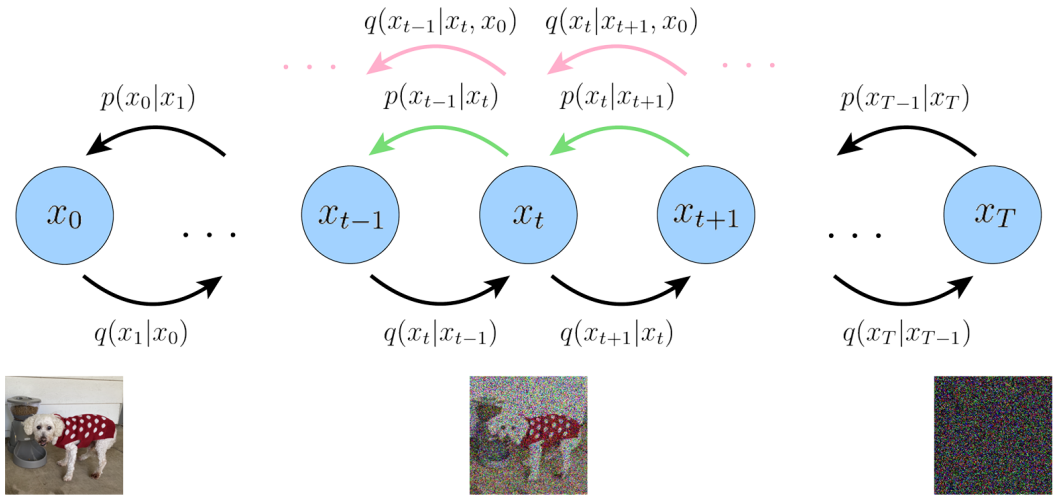


Figure 3.2: Illustration of the hierarchical VAE interpretation of diffusion models. The black arrows denote the forward process $q(x_t \mid x_{t-1})$, which is fixed and Gaussian. The green arrows indicate the learned reverse transitions $p_\theta(x_{t-1} \mid x_t)$, while the pink bidirectional arrows represent variational approximations of the reverse posterior $q(x_{t-1} \mid x_t, x_0)$. Each latent x_t becomes progressively noisier as t increases, and generation proceeds by denoising from x_T back to x_0.

Figure 3.2 provides a visual depiction of the diffusion model as a hierarchical latent variable model. The sequence of latent variables $\{x_t\}$ evolves through a forward noising process (black arrows), while the model learns to reverse this trajectory through the denoising process (green arrows). The variational posterior $q(x_{t-1} \mid x_t, x_0)$ used in ELBO derivation is shown in pink. It is computed by using the Markov property to $q(x_t \mid x_{t-1})$ so it becomes $q(x_t \mid x_{t-1}, x_0)$ and then applying Bayes rule on it.

Equation 3.14 expands into a sum of Kullback-Leibler divergences and a final log-likelihood term:

$$\begin{aligned}\log p_\theta(x) \geq \mathbb{E}_{q(x_1|x_0)}\left[\log p_\theta(x_0 \mid x_1)\right] + \mathbb{E}_{q(x_{T-1},x_T|x_0)}\left[\log \frac{p(x_T)}{q(x_T \mid x_{T-1})}\right] \\ + \sum_{t=1}^{T-1} \mathbb{E}_{q(x_{t-1},x_t,x_{t+1}|x_0)}\left[\log \frac{p_\theta(x_t \mid x_{t+1})}{q(x_t \mid x_{t-1})}\right] \qquad (3.15)\end{aligned}$$

$$\begin{aligned}= \underbrace{\mathbb{E}_{q(x_1|x_0)}\left[\log p_\theta(x_0 \mid x_1)\right]}_{\text{reconstruction term}} - \underbrace{\mathbb{E}_{q(x_{T-1}|x_0)}\left[D_{\mathsf{KL}}\left(q(x_T \mid x_{T-1}) \parallel p(x_T)\right)\right]}_{\text{prior matching term}} \\ - \sum_{t=1}^{T-1} \underbrace{\mathbb{E}_{q(x_{t-1},x_{t+1}|x_0)}\left[D_{\mathsf{KL}}\left(q(x_t \mid x_{t-1}) \parallel p_\theta(x_t \mid x_{t+1})\right)\right]}_{\text{denoising matching term}} \qquad (3.16)\end{aligned}$$

Equation 3.16 consists of the following three terms: reconstruction, prior matching and denoising matching term, in order or appearance. The first term predicts the probability of the original data sample given the latent at the first timestep. The second term, expressed through the KL-divergence, is minimized when the distribution at the last timestep is a pure Gaussian distribution. It has no trainable parameters which simply means that it is 0. The third term aims to align the two distributions from the forward and the reverse process. This corresponds to predicting the noise that was added at timestep t. These two noises should be as close as possible which is achieved by minimizing the KL-divergence.

Unlike standard VAEs, where the inference model is learned, in diffusion models the forward process is fixed and analytically tractable while only the reverse model is trained.

One of the key benefits of this formulation is the parameter sharing across timesteps. The same neural network ϵ_θ is reused to approximate the reverse transitions $p_\theta(x_{t-1} \mid x_t)$ for all t, making the model scalable and efficient to train. This view emphasizes the theoretical grounding and probabilistic interpretability of diffusion models.

3.2.3. Denoising Diffusion Probabilistic Models (DDPM)

Denoising Diffusion Probabilistic Models (DDPM) [21] provide a powerful and stable generative modeling framework based on a Markovian noising and denoising process. Inspired by nonequilibrium thermodynamics, the forward process gradually corrupts a data sample by adding Gaussian noise over T timesteps as in Equation 3.10. The cumulative forward process allows direct sampling from x_t at any timestep t via:

$$q(x_t \mid x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}), \qquad (3.17)$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$.

The goal of the reverse process is to learn a model $p_\theta(x_{t-1} \mid x_t)$ that reconstructs the original data distribution by gradually removing noise. This is modeled using a neural network $\epsilon_\theta(x_t, t)$ that predicts the noise added at each timestep. The reverse distribution is approximated as:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2\mathbf{I}), \qquad (3.18)$$

where μ_θ is derived from the predicted noise and the known variance schedule.

Figure 3.3 presents the generative modeling framework of DDPM as a Markov chain that consists of two key processes. The forward process $q(x_t \mid x_{t-1})$ adds small amounts of Gaussian noise at each step, gradually transforming a clean data sample x_0 into pure noise x_T. The reverse process $p_\theta(x_{t-1} \mid x_t)$ is learned to gradually denoise the sample back toward the data distribution. The figure shows how the model learns to approximate this reverse trajectory to recover structured data from noise.

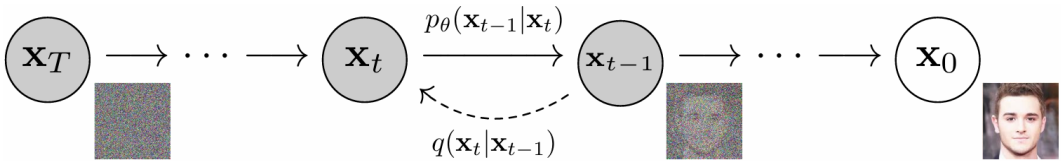


Figure 3.3: The directed graphical model used in DDPM [21]. The forward process $q(x_t \mid x_{t-1})$ incrementally adds noise to a clean image x_0, resulting in $x_T \sim \mathcal{N}(0, \mathbf{I})$. The reverse process $p_\theta(x_{t-1} \mid x_t)$ is learned to iteratively denoise the sample.

Instead of directly optimizing the complex variational lower bound (ELBO), as described in subsection 3.2.2, DDPM simplifies training by minimizing a denoising score-matching loss which reduces to the mean squared error between the predicted and true noise:

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right], \tag{3.19}$$

with $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Sampling begins from pure Gaussian noise $x_T \sim \mathcal{N}(0, \mathbf{I})$ and proceeds iteratively through the learned reverse process to produce a final sample x_0. Although sampling is slow due to the large number of timesteps, DDPM generates highly realistic and diverse outputs without adversarial training.

In practice, DDPM have been successfully applied to a wide range of generative tasks, including unconditional image generation, inpainting, super-resolution and text-to-image synthesis. Their stability and likelihood-based training objective make them particularly attractive compared to adversarial models and they serve as the foundation for many modern diffusion-based systems such as GLIDE [38] and Stable Diffusion [53].

3.2.4. Score-Based Diffusion Formulation

An alternative perspective on diffusion models arises from score-based generative modeling, which centers around estimating the gradient of the log-density $\nabla_{x_t} \log p(x_t)$ at various noise levels. This formulation connects closely with Tweedie's formula, which provides the posterior mean of an exponential family distribution given its score function.

For a Gaussian variable $z \sim \mathcal{N}(z; \mu_z, \Sigma_z)$, Tweedie's Formula states that:

$$\mathbb{E}[\mu_z \mid z] = z + \Sigma_z \nabla_z \log p(z) \tag{3.20}$$

where $p(z)$ is the marginal density and the correction term involves the score function $\nabla_z \log p(z)$. In diffusion models, this insight helps reformulate the reverse process in terms of score-based objectives.

Let x_t be a noisy sample from the forward process. Using the known closed-form expression for $q(x_t \mid x_0)$:

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{3.21}$$

we can express the posterior mean $\mu_\theta(x_t, t)$ as:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\bar{\alpha}_t}}x_t + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\nabla_{x_t} \log p(x_t), \tag{3.22}$$

which is derived by plugging Tweedie's formula into the denoising mean. In DDPM, the neural network $\epsilon_\theta(x_t, t)$ is trained to predict the Gaussian noise added during the forward process. In contrast, score-based diffusion models train a network $s_\theta(x_t, t)$ to directly estimate the score function $\nabla_{x_t} \log p(x_t)$, which represents the direction in data space that increases the log-probability. Under Gaussian noise assumptions, these two approaches are mathematically equivalent up to a known scaling factor, allowing DDPM training to be interpreted as a form of score matching.

Let $s_\theta(x_t, t)$ denote the learned score:

$$s_\theta(x_t, t) \approx \nabla_{x_t} \log p(x_t). \tag{3.23}$$

Thus, we can rewrite the denoising mean as:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\bar{\alpha}_t}} x_t + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} s_\theta(x_t, t). \tag{3.24}$$

Thus, the learning objective becomes the following denoising score-matching loss:

$$\mathbb{E}_{x_0,\epsilon,t}\left[\|s_\theta(x_t, t) - \nabla \log p(x_t)\|_2^2\right]. \tag{3.25}$$

The score function $\nabla_{x_t} \log p(x_t)$ tells us the direction in which we should move in space to increase the log-probability of the sample, i.e. making that data point more probable. Intuitively, if Gaussian noise is added to corrupt a data point, then moving in the opposite direction of the noise would best denoise it. Therefore, learning the score function corresponds to modeling the negative gradient of the source noise. This perspective tightly connects score matching with denoising and provides a solid theoretical foundation for training diffusion models using score-based objectives.

In score-based diffusion models, there exists a fundamental equivalence between the predicted noise $\epsilon_\theta(x_t, t)$ and the score function $\nabla_{x_t} \log p(x_t)$. This equivalence can be derived from Tweedie's formula and Equation 3.12, as shown in [36], and takes the form:

$$\nabla_{x_t} \log p(x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \epsilon_0. \tag{3.26}$$

Substituting into the score-based parameterization:

$$\epsilon_\theta(x_t, t) = -\sigma_t \nabla_{x_t} \log p_t(x_t), \tag{3.27}$$

we recover that ϵ_θ and the score function are proportional under Gaussian noise assumptions, justifying the use of noise prediction networks for training diffusion models.

This equivalence justifies the effectiveness of the DDPM training loss and provides a unified theoretical foundation for discrete-time and continuous-time diffusion models. Overall, the score-based view enriches the theoretical understanding of diffusion models and offers flexibility in both model training and inference-time sampling.

3.2.5. Sampling and Inference-Time Denoising

Once the diffusion model is trained to predict noise, it can be used to generate new data samples. This process is called *inference* and it involves running the learned reverse diffusion process starting from pure Gaussian noise until complete denoising of the latent z. This enables the model to generate new images by denoising random noise samples.

This is possible through the learned approximation of the reverse transitions $p_\theta(x_{t-1} \mid x_t)$. These transitions allow us to iteratively denoise a sample starting from $x_T \sim \mathcal{N}(0, \mathbf{I})$, following the estimated reverse trajectory down to x_0, which represents a clean image.

Practically, sampling in DDPM [21] is performed through a sequence of transformations applied in discrete time steps from $t = T$ down to $t = 1$. At each step, the model uses a neural network $\epsilon_\theta(x_t, t)$ to predict the noise component in x_t which is then used to reconstruct x_{t-1}. Equation 3.28 illustrates sampling at inference-time.

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t)\right) + \sigma_t z, \tag{3.28}$$

where $z \sim \mathcal{N}(0, \mathbf{I})$ is sampled independently at each step, and σ_t is a noise scaling term related to the variance schedule.

This iterative denoising process progressively removes noise from the initial random sample, guided by the predictions of the learned model. The quality of the final output depends on the accuracy of the noise predictions.

While the original DDPM framework requires hundreds to thousands of sampling steps for high fidelity, there are techniques that reduce the number of denoising steps without significantly compromising generation quality.

Inference-time denoising not only enables fast and flexible sample generation but also leads to guidance-based interventions, where external objectives or conditioning signals can be used to steer the reverse process. This is key for controlling text-to-image generation, which we explore further in the next sections.

3.2.6. Diffusion Model Guidance

What makes T2I diffusion models stand out is their ability to incorporate different conditional information signals into the denoising process to achieve controlled image generation. Usually, this is achieved by fine-tuning methods, such as [23], which allow the model to adapt to new tasks. Instead, diffusion models can be *guided*, i.e. steer the generation process, towards desired outcomes without retraining. The key advantages are computational efficiency and preservation of the generation ability of T2I models.

In the context of conditional generation, such as generating an image aligned with a given text prompt, guidance adjusts the denoising trajectory at inference time so that the model more strongly aligns with the conditioning signal. This is possible because of the probabilistic structure of diffusion models, which allows gradient-based manipulation of the generation trajectory.

There are two major classes of guidance: **classifier-based guidance** and **classifier-free guidance**. Both methods aim to guide the reverse diffusion process so that the final generated sample better reflects the desired conditional properties.

Classifier-Based Guidance. Originally proposed in the score-based formulation of diffusion [57], classifier guidance relies on an external classifier trained to predict the probability of a condition y (e.g., a class label) given a noisy image x_t at timestep t. The conditional gradient is defined as:

$$\nabla_{x_t} \log p(x_t \mid y, t) = \nabla_{x_t} \log p(x_t \mid t) + \gamma \nabla_{x_t} \log p(y \mid x_t, t), \tag{3.29}$$

where γ is a scalar parameter controlling the strength of the guidance. This formulation shows that to generate a sample more likely to correspond to condition y, we can move in the direction that increases both the unconditional likelihood of the image x_t and the conditional likelihood of the label y.

Due to the connection between diffusion models and score-based modeling, this gradient can be directly incorporated into the reverse sampling steps. However, in practice, training a reliable classifier on noisy images x_t is challenging, especially at early timesteps where noise dominates and the signal is weak. This limits the effectiveness and scalability of classifier-based methods.

Classifier-Free Guidance (CFG). To address these limitations, [22] introduced classifier-free guidance, a simpler and more practical alternative that eliminates the need for a separate classifier. Instead, the diffusion model is trained jointly on both conditional and unconditional data. During training, the conditioning signal (e.g., a text prompt) is randomly dropped, enabling the model to learn both conditional and unconditional denoising capabilities.

At inference, two forward passes are made through the network: one with the condition y and one without. The final guided prediction is then interpolated between the two:

$$\nabla_{x_t} \log\, p(x_t|y,t) = \nabla_{x_t} \log\, p(x_t,t) + \gamma(\nabla_{x_t} \log\, p(x_t|y,t) - \nabla_{x_t} \log\, p(x_t,t)). \tag{3.30}$$

where γ again controls the guidance strength. A higher γ pushes the generation closer to the conditional direction, at the risk of reduced sample diversity.

Classifier-free guidance is simple to implement, scales well with high-dimensional conditions like language, and enables precise control over the influence of the conditioning signal. Hence, it allows to add control to the diffusion generation process.

3.3. Latent Diffusion Models

Latent Diffusion Models (LDM) [53] train an autoencoder that encodes high-dimensional images into a lower-dimensional latent representation. Then, a diffusion model is trained and sampled in this latent space which substantially reduces memory and compute requirements during training and inference. This directly correlates with the wide adoption of the Stable Diffusion (SD) models [53] by the community.

Instead of learning a diffusion process over images x_0, SD applies the forward and reverse diffusion to latent representations $z_0 = \mathcal{E}(x_0)$ obtained via a pretrained encoder $\mathcal{E}$. The generative process operates in latent space by denoising from z_T to z_0, after which a decoder $\mathcal{D}$ reconstructs the image $x_0 = \mathcal{D}(z_0)$. Figure 3.4 illustrates the architecture of these models. The SD pipeline enables high-resolution generation (e.g., 512×512 or 1024×1024) with significantly lower computational overhead than pixel-space diffusion models.

To condition generation on text, LDM leverages the CLIP-ViT-L/14 [50] text encoder and a cross-attention mechanism that injects the encoded text into the U-Net at each denoising step, shown in Figure 3.4. This design allows fine-grained control over the image semantics using natural language prompts.

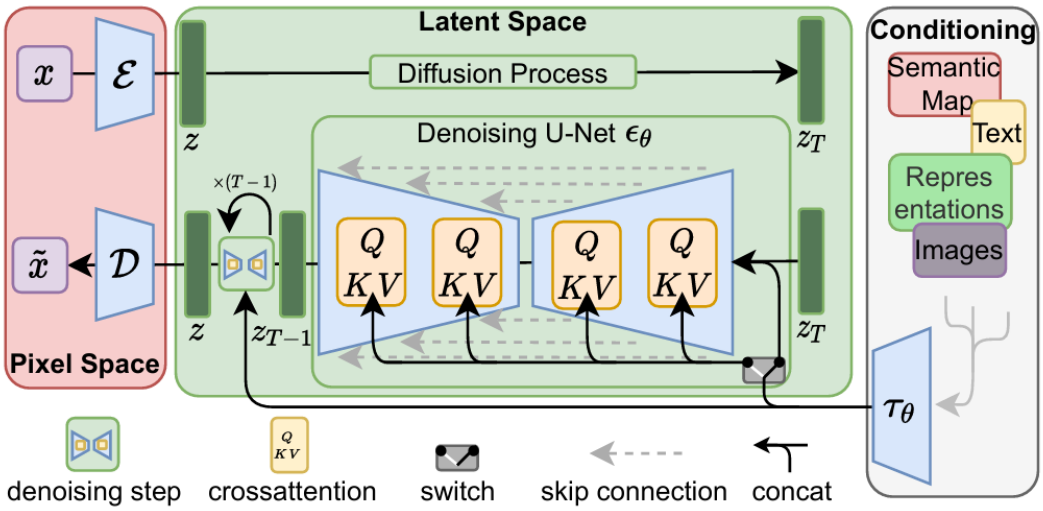


Figure 3.4: Architecture of Latent Diffusion Models (Figure 3 from [53]). The top panel shows the image-to-latent encoder $\mathcal{E}$, the middle panel is the core U-Net-based diffusion model operating in latent space z, and the bottom panel is the decoder $\mathcal{D}$ used to reconstruct the image from the latent. Conditioning is provided via CLIP embeddings processed by a transformer.

3.4. Vision-Language Models and Spatial Understanding

Text-to-image generation models such as Stable Diffusion [53] rely on vision-language models (VLM) to ground visual synthesis in natural language. These models aim to align the semantic representations of text and images in a shared latent space, enabling meaningful cross-modal interactions. Among these, CLIP (Contrastive Language–Image Pretraining) [50] has emerged as a central component in modern generative architectures.

3.4.1. CLIP as the Foundation of Stable Diffusion

CLIP is trained using a contrastive objective over a large-scale dataset of 400 million image–text pairs. It consists of text and image encoders, a Transformer-based architecture for text and a Vision Transformer (ViT) for images. They are jointly trained to maximize cosine similarity between corresponding text-image pairs while minimizing cosine similarity between the incorrect pairs. The resulting model learns an embedding space where semantically aligned images and texts lie close together, while the ones

that are semantically different lie further apart.

Figure 3.5 illustrates the training and inference pipeline of CLIP. During pretraining (Step 1), CLIP jointly optimizes a text encoder and an image encoder using a contrastive loss over a dataset of image–text pairs. The objective encourages the cosine similarity of matching pairs to be high while pushing apart mismatched ones. This ensures that well aligned images and texts lie close together in the embedding space. In Step 2, the learned embedding space is used to construct a zero-shot classifier by encoding textual descriptions of the prompts (e.g., “a photo of a dog”) and comparing them to the embedding of a test image. In Step 3, during inference the model chooses the prompt corresponding to the text embedding with the highest similarity to the image embedding.

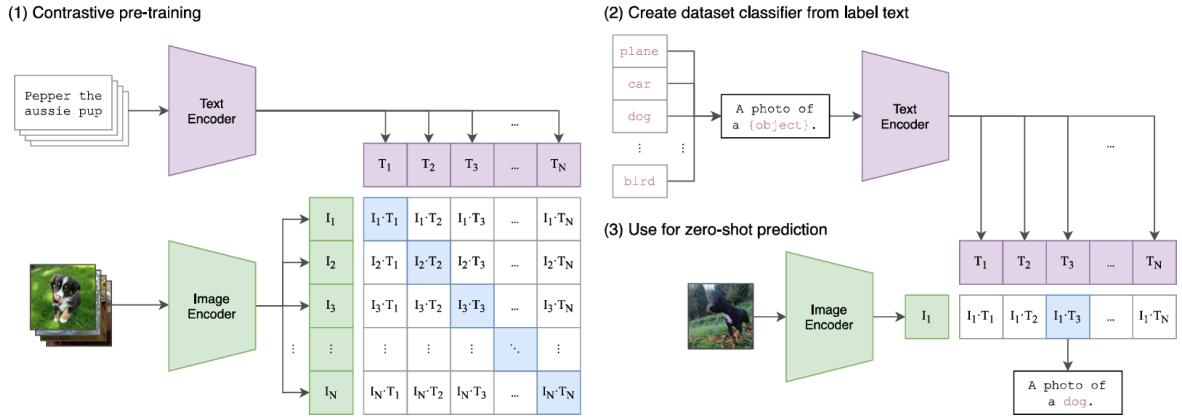


Figure 3.5: Illustration of the CLIP training and inference pipeline. (1) During contrastive pretraining a joint embedding space is learned by aligning image–text pairs. (2) Text embeddings are generated from class names using natural language templates. (3) Zero-shot classification is performed by measuring similarity between the image embedding and candidate text embeddings.

The CLIP text encoder is fundamental for Stable Diffusion [53] as it is used in their pretraining. It embeds text information into a fixed sized tensor which is injected into the denoising UNet as cross-attention keys and values during each denoising step. Thus, the alignment between text and generated images is highly dependent on the degree to which the CLIP text encoder manages to encode semantic understanding.

3.4.2. Impact on Text-to-Image Generation

Since Stable Diffusion [53] conditions its generation on CLIP embeddings, its outputs inherit both the strengths and limitations of CLIP. While the model excels at rendering semantically appropriate and photorealistic images, it frequently misaligns objects when prompts specify spatial relationships.

Recent empirical analyses [66, 63] confirm that diffusion models based on CLIP perform poorly on spatial compositional benchmarks. These failures arise from the inability of the CLIP text encoder to fully capture the variability in natural language with its rich semantics.

3.4.3. Limitations of CLIP in Spatial Reasoning

Despite its strong generalization capabilities, CLIP has a known deficiency: it is not trained to reason about spatial relationships. The contrastive objective rewards global semantic alignment between images and texts but does not enforce fine-grained grounding of object positions or spatial configurations. As a result, relational prompts such as “a dog to the left of a cat” are often encoded similarly to “a cat to the left of a dog,” leading to positional ambiguity in downstream models [53].

Moreover, CLIP lacks explicit supervision for spatial prepositions and relational phrases. Its pretraining data contains some noisy and vague captions which further weakening its ability to capture structured relationships. This is reflected in downstream text-to-image diffusion models that often fail to produce spatially cognizant images.

4

Inference-Time Guidance

4.1. Motivation

Diffusion models are dominating in the field of high-fidelity and realistic image generation [46, 53]. This stems from their iterative denoising capability which allows them to achieve fine-grained detail levels towards the generation of high quality samples. However, diffusion models were not designed to allow for controlling the generation process. **Guidance** influences the diffusion models denoising trajectory. The generation process can be steered towards a particular region in the learned prior distribution by applying a carefully chosen and well tuned external signal. If the guidance signal is designed in such a way that it provides a meaningful input to the model, the output will be a sample that aligns with the specified conditional information, e.g. text prompts, images, bounding boxes, etc.

Guidance is applied at inference-time, so it does not require any additional training and it does not incur intensive computational cost. It can be performed using a classifier, as discussed in subsection 3.2.6, or through classifier-free guidance, explained in Equation 3.2.6. However, guidance is not the solution to every problem. Despite inference-time guidance, diffusion models are still limited in their generational power in adhering to compositionality [24]. Compositionality is when the text specifies properties or relations between objects, e.g. attributes such as color, shape, texture or spatial/non-spatial relationships. For example, Stable Diffusion frequently violates simple positional constraints, such as “a dog to the left of a cat,” by swapping object positions or producing ambiguous layouts [53].

A common approach to address such failures is to fine-tune the model on spatially curated datasets [6, 66]. However, this strategy incurs high computational cost, demands large amounts of annotated data and often sacrifices generality in favor of specialization. In many real-world applications, retraining or fine-tuning is impractical due to time and resources.

Inference-time guidance methods offer an attractive alternative. These approaches intervene in the sampling process of a pretrained diffusion model without modifying its weights. By altering intermediate representations, such as attention maps, latents or noise predictions, they steer the generation process toward satisfying external constraints, such as object presence, layout and alignment with text.

In recent years several inference-time guidance methods have emerged, each targeting different aspects of controllability. Some techniques modify attention mechanisms to enhance object binding [15], while others optimize noise trajectories to enforce layout constraints [59]. More recent approaches combine these ideas with guidance signals to better reflect the prompt’s structure [33, 44].

These methods are particularly attractive because of their plug-and-play nature: they require no retraining, retain the generalization ability of the base model and allow for dynamic customization at inference. However, their effectiveness varies significantly depending on the target control objective, the quality of the underlying attention maps and the granularity of the intervention.

The challenges in T2I generation addressed by recent works using inference-time guidance are discussed in the next sections.

4.2. Missing objects

A recurring challenge in diffusion-based T2I generation is the failure to render all objects mentioned in the prompt, a problem known as catastrophic neglect. Inference-time methods such as Attend&Excite [8] directly target this issue by optimizing the attention maps to ensure that each subject token receives sufficient focus. The method defines a loss based on the maximum attention response for each token and adjusts the latent representation to excite under-attended tokens. While Attend&Excite successfully improves object presence, it does not account for spatial relationships between objects.

Similarly, Composable Diffusion [35] ensures object presence by decomposing the prompt into individual concepts and generating classifier-free guidance terms for each. These terms are linearly combined to influence the denoising trajectory, enabling generation conditioned on logical relations such as ”object A and object B” or ”object A and not object B”. However, as with Attend&Excite, the approach ensures inclusion rather than spatial configuration. These inference-time strategies demonstrate that the absence of objects can often be attributed to weak attention signals, and that careful manipulation of the latent and attention spaces can partially mitigate this issue without retraining the model.

4.3. Image editing

Inference-time guidance methods can also be adapted for the task of image editing. These approaches repurpose the generation process by injecting spatial constraints into the reverse denoising trajectory to modify the layout of existing objects.

Self-Diffusion Guidance [13] is one such method that operates directly on the latent variables during the reverse process of the diffusion model. It leverages cross-attention maps to guide object placement by applying a MSE-based loss function at selected timesteps to control for the appearance, size, shape and position of objects. The moving of objects is done through fixed locations. This work shows a lot of potential for extracting useful information from the attention maps of objects but provides only qualitative evaluation and no quantitative analysis.

DiffUHaul [5] extends this idea in the context of object dragging. The method introduces a training-free pipeline for image editing based on blob layout information. By extracting attention blobs corresponding to object locations, DiffUHaul manipulates them to guide the diffusion process toward new spatial configurations. This enables object relocation similar to dragging elements in a scene. Unlike layout-conditioned methods, which require explicit spatial annotations (e.g., bounding boxes), DiffUHaul derives positional information directly from attention maps.

Another notable inference-time editing method is Prompt-to-Prompt [19], which enables fine-grained image manipulation by directly modifying the cross-attention maps of a pre-trained diffusion model. This method allows semantic edits, such as replacing, adding or removing objects, by interpolating or swapping attention maps between source and target prompts. The key insight is that cross-attention layers encode spatial and semantic information tied to the prompt structure and by controlling these layers, the model can be steered to produce images that reflect the desired modifications while preserving style and composition.

4.4. Inference-time Guidance Methods for Spatial Alignment

The approaches for introducing spatial understanding in T2I generation during inference-time are listed in the next sections.

4.4.1. Layout-based Guidance for Spatial Alignment

Layout-based guidance methods approach spatial alignment by conditioning image generation on structured spatial representations such as bounding boxes, segmentation masks or depth maps. These techniques ensure precise object placement by explicitly encoding spatial priors into the generation process.

Spatial-Aware Latent Initialization [58] modifies the initial latent noise to reflect a predefined object layout, allowing the diffusion model to preserve spatial structure throughout the denoising process. Training-Free Layout Control[10] manipulates cross-attention maps at inference time to align object features with user-specified bounding boxes, enabling layout control without retraining. Grounded

Text-to-Image Synthesis with Attention Refocusing [45] steers attention maps toward grounded object regions during generation, improving spatial grounding through a refocusing module. GLIGEN [33] supports open-set grounded generation by injecting grounding tokens and bounding box coordinates during sampling, generalizing across unseen objects and layouts. ReGround [32] improves spatial and textual alignment by refining grounding quality at no additional training cost, using a plug-and-play refiner.

Together, these works highlight the effectiveness of leveraging explicit spatial inputs to control object positioning at the cost of requiring structured annotations or layout specifications.

4.4.2. LLM-grounded T2I generation

LLM-grounded T2I generation enhances spatial reasoning by leveraging large language models (LLMs) to extract structured layouts from natural language prompts. In LLM-grounded Diffusion[34], an LLM first generates a scene layout composed of captioned bounding boxes which are then used by a controller to guide a standard diffusion model, enabling layout-aware image synthesis. Similarly, Unlocking Spatial Comprehension in T2I Diffusion Models[12] uses an LLM to generate a structured layout, followed by a two-stage generation process. An initial image is created with one object, then image editing is performed to insert a second object at the correct spatial location. This pipeline relies on synthetic datasets and LLM-generated spatial instructions, demonstrating the potential of combining linguistic priors with layout-conditioned diffusion models for improved spatial control.

4.5. Strengths and Limitations of Inference-Time Guidance in Spatial Alignment

Inference-time guidance offers a flexible and training-free approach to improving spatial alignment in T2I diffusion models. By intervening during the iterative denoising process, these methods can steer generation toward better spatial configurations without modifying the model's original weights and without significant computational overhead. This makes them especially attractive since fine-tuning methods require both supervision and a lot of computational resources. Due to their modularity, inference-time methods can be applied on any diffusion-based backbone. Moreover, inference-time methods retain the model's generalization ability across diverse prompts and domains, avoiding overfitting to spatially annotated datasets.

Despite these advantages, inference-time methods face some limitations. They often struggle with enforcing the correct spatial relationship due to concept entanglement in the attention maps. These internal representations are not always precise which makes them unreliable for accurate layout control. Since they are applied on top of SD model, they inherit their intrinsic limitations.

5

Conclusion and Future Directions

Diffusion models have become prevalent in T2I generation due to their high-quality outputs and training stability. Throughout this thesis, their underlying principles were explored, such as text conditioning using CLIP [49] and manipulation of attention maps [19]. Despite their impressive generative capabilities, diffusion models often fail to accurately reflect spatial relationships described in textual prompts.

To address this limitation, the focus was on inference-time guidance strategies that operate without retraining the base model. These methods take advantage of the iterative denoising process to inject constraints or corrections during sampling. Compared to fine-tuning, inference-time methods offer flexibility, generality and low computational overhead, but they also face some limitations. These include reliance on noisy or entangled internal representations and the inability to guarantee spatial correctness.

In this context, the proposed `InfSplign` method demonstrated the effectiveness of applying targeted spatial losses over attention maps to guide generation toward spatially coherent outputs. It offers a training-free, efficient solution that bridges the gap between spatial alignment and object presence. Future research on `InfSplign` may focus on strategies that jointly optimize spatial positioning and visual fidelity, such as attention refinement and introducing multiple loss function covering other alignment aspects, to further improve the controllability and reliability of generative models. As inference-time approach `InfSplign` achieves a competitive performance to fine-tuning-based methods and as being faster and more efficient at T2I generation, it demonstrates its great potential in synthesizing spatially cognizant images and provides an easy way for experimentation through its plug-and-play nature.

References

[1] DeepSeek AI. *DeepSeek: Advancing Open-Source Language Models*. https://deepseek.com. 2023.

[2] Mistral AI. *Introducing Mistral 7B and Mixtral*. https://mistral.ai/news/. 2023.

[3] Anthropic. *Claude: An AI Assistant by Anthropic*. https://www.anthropic.com/index/introducing-claude. 2023.

[4] Martin Arjovsky and Léon Bottou. "Towards Principled Methods for Training Generative Adversarial Networks". In: *arXiv preprint arXiv:1701.04862* (2017).

[5] Omri Avrahami et al. "Diffuhaul: A training-free method for object dragging in images". In: *SIGGRAPH Asia 2024 Conference Papers*. 2024, pp. 1–12.

[6] Agneet Chatterjee et al. "Getting it right: Improving spatial consistency in text-to-image models". In: *European Conference on Computer Vision*. Springer. 2024, pp. 204–222.

[7] Agneet Chatterjee et al. "Revision: Rendering tools enable spatial fidelity in vision-language models". In: *European Conference on Computer Vision*. Springer. 2024, pp. 339–357.

[8] Hila Chefer et al. "Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models". In: *ACM transactions on Graphics (TOG)* 42.4 (2023), pp. 1–10.

[9] Boyuan Chen et al. "SpatialVLM: Endowing Vision-Language Models with Spatial Reasoning Capabilities". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 14455–14465.

[10] Minghao Chen, Iro Laina, and Andrea Vedaldi. "Training-free layout control with cross-attention guidance". In: *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2024, pp. 5343–5353.

[11] Google DeepMind. *Gemini: Google's Multimodal Foundation Model*. https://deepmind.google/technologies/gemini. 2023.

[12] Mohammad Mahdi Derakhshani et al. "Unlocking spatial comprehension in text-to-image diffusion models". In: *arXiv preprint arXiv:2311.17937* (2023).

[13] Dave Epstein et al. "Diffusion Self-Guidance for Controllable Image Generation". In: (June 2023). URL: https://arxiv.org/abs/2306.00986v3.

[14] Patrick Esser et al. "Scaling rectified flow transformers for high-resolution image synthesis". In: *Forty-first international conference on machine learning*. 2024.

[15] Weixi Feng et al. "Training-Free Structured Diffusion Guidance for Compositional Text-to-Image Synthesis". In: (Dec. 2022). URL: https://arxiv.org/abs/2212.05032v3.

[16] Weixi Feng et al. "Training-Free Structured Diffusion Guidance for Compositional Text-to-Image Synthesis". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=PUIqjT4rzq7.

[17] Ian Goodfellow et al. "Generative adversarial networks". In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[18] Woojung Han et al. "Spatial Transport Optimization by Repositioning Attention Map for Training-Free Text-to-Image Synthesis". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2025.

[19] Amir Hertz et al. "Prompt-to-prompt image editing with cross attention control". In: *arXiv preprint arXiv:2208.01626* (2022).

[20] Irina Higgins et al. "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *International Conference on Learning Representations (ICLR)*. 2017.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. “Denoising diffusion probabilistic models”. In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[22] Jonathan Ho, Google Research, and Tim Salimans. “Classifier-Free Diffusion Guidance”. In: *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*. Dec. 2021.

[23] Edward J Hu et al. “Lora: Low-rank adaptation of large language models.” In: *ICLR* 1.2 (2022), p. 3.

[24] Kaiyi Huang et al. “T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation”. In: *Advances in Neural Information Processing Systems*. Vol. 36. 2023, pp. 78723–78747.

[25] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. “Perceptual losses for real-time style transfer and super-resolution”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 694–711.

[26] Norman P Jouppi et al. “In-datacenter performance analysis of a tensor processing unit”. In: *Proceedings of the 44th annual international symposium on computer architecture*. 2017, pp. 1–12.

[27] Tero Karras, Samuli Laine, and Timo Aila. “A Style-Based Generator Architecture for Generative Adversarial Networks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 4401–4410.

[28] Diederik P Kingma, Max Welling, et al. *Auto-encoding variational bayes*. 2013.

[29] Anders Boesen Lindbo Larsen et al. “Autoencoding beyond pixels using a learned similarity metric”. In: *International Conference on Machine Learning (ICML)*. 2016, pp. 1558–1566.

[30] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444.

[31] Christian Ledig et al. “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 4681–4690.

[32] Phillip Y Lee and Minhyuk Sung. “ReGround: Improving Textual and Spatial Grounding at No Cost”. In: *European Conference on Computer Vision*. Springer. 2024, pp. 275–292.

[33] Yuheng Li et al. “Gligen: Open-set grounded text-to-image generation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 22511–22521.

[34] Long Lian et al. “LLM-grounded Diffusion: Enhancing Prompt Understanding of Text-to-Image Diffusion Models with Large Language Models”. In: (). URL: `https://llm-grounded-diffusion.github.io`..

[35] Nan Liu et al. “Compositional visual generation with composable diffusion models”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 423–439.

[36] Calvin Luo. “Understanding Diffusion Models: A Unified Perspective”. In: (Aug. 2022). URL: `https://arxiv.org/abs/2208.11970v1`.

[37] Microsoft and OpenAI. *Microsoft Copilot*. `https://copilot.microsoft.com`. 2023.

[38] Alex Nichol et al. “Glide: Towards photorealistic image generation and editing with text-guided diffusion models”. In: *arXiv preprint arXiv:2112.10741* (2021).

[39] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. “Neural Discrete Representation Learning”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2017, pp. 6306–6315.

[40] Aaron van den Oord et al. “WaveNet: A Generative Model for Raw Audio”. In: *arXiv preprint arXiv:1609.03499* (2016).

[41] OpenAI. *ChatGPT: Optimizing Language Models for Dialogue*. `https://openai.com/blog/chatgpt`. Accessed: 2025-06-09. 2022.

[42] OpenAI. *GPT-4 Technical Report*. `https://openai.com/research/gpt-4`. Accessed: 2025-06-09. 2023.

[43] OpenAI. *Introducing GPT-4V(ision)*. `https://openai.com/blog/gpt-4v-system-card`. 2023.

[44] Quynh Phung, Songwei Ge, and Jia-Bin Huang. “Grounded Text-to-Image Synthesis with Attention Refocusing”. In: (). URL: https://attention-refocusing.github.io/.

[45] Quynh Phung, Songwei Ge, and Jia-Bin Huang. “Grounded text-to-image synthesis with attention refocusing”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 7932–7942.

[46] Dustin Podell et al. “Sdxl: Improving latent diffusion models for high-resolution image synthesis”. In: *arXiv preprint arXiv:2307.01952* (2023).

[47] Alec Radford, Luke Metz, and Soumith Chintala. “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”. In: *International Conference on Learning Representations (ICLR)*. 2016.

[48] Alec Radford et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9.

[49] Alec Radford et al. “Learning Transferable Visual Models From Natural Language Supervision”. In: *Proceedings of Machine Learning Research* 139 (Feb. 2021), pp. 8748–8763. ISSN: 26403498. URL: https://arxiv.org/abs/2103.00020v1.

[50] Alec Radford et al. “Learning transferable visual models from natural language supervision”. In: *International conference on machine learning*. PmLR. 2021, pp. 8748–8763.

[51] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. “Generating Diverse High-Fidelity Images with VQ-VAE-2”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019, pp. 14866–14876.

[52] Yi Ren et al. “FastSpeech: Fast, Robust and Controllable Text to Speech”. In: *Advances in Neural Information Processing Systems*. 2019.

[53] Robin Rombach et al. “High-resolution image synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 10684–10695.

[54] Chitwan Saharia et al. “Photorealistic text-to-image diffusion models with deep language understanding”. In: *Advances in neural information processing systems* 35 (2022), pp. 36479–36494.

[55] Tim Salimans et al. “Improved Techniques for Training GANs”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2016, pp. 2234–2242.

[56] Christoph Schuhmann et al. “Laion-5b: An open large-scale dataset for training next generation image-text models”. In: *Advances in neural information processing systems* 35 (2022), pp. 25278–25294.

[57] Yang Song et al. “Score-based generative modeling through stochastic differential equations”. In: *arXiv preprint arXiv:2011.13456* (2020).

[58] Wenqiang Sun et al. “Spatial-Aware Latent Initialization for Controllable Image Generation”. In: (Jan. 2024). URL: https://arxiv.org/abs/2401.16157v1.

[59] Wenqiang Sun et al. “Spatial-aware latent initialization for controllable image generation”. In: *arXiv preprint arXiv:2401.16157* (2024).

[60] Arash Vahdat and Jan Kautz. “NVAE: A Deep Hierarchical Variational Autoencoder”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020, pp. 19667–19679.

[61] Aaron Van den Oord et al. “Conditional image generation with pixelcnn decoders”. In: *Advances in neural information processing systems* 29 (2016).

[62] Ashish Vaswani et al. “Attention is All You Need”. In: *Advances in Neural Information Processing Systems*. 2017.

[63] Chao Wang et al. “Information Theoretic Text-to-Image Alignment”. In: (May 2024). URL: https://arxiv.org/abs/2405.20759v1.

[64] Qiucheng Wu et al. “Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 7766–7776.

[65] Jiahui Yu et al. “Scaling autoregressive models for content-rich text-to-image generation”. In: *arXiv preprint arXiv:2206.10789* 2.3 (2022), p. 5.

[66] Gaoyang Zhang et al. “CoMPaSS: Enhancing Spatial Understanding in Text-to-Image Diffusion Models”. In: *arXiv preprint arXiv:2412.13195* (2024).