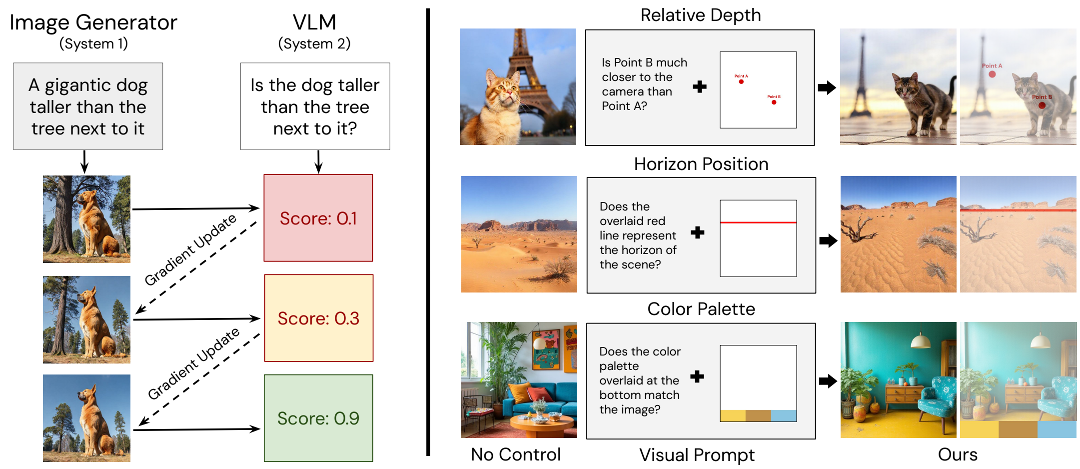
Dual-Process Image Generation

Grace Luo[1] Jonathan Granskog[2] Aleksander Holynski[1*] Trevor Darrell[1*]

[1]UC Berkeley [2]Runway



(a) Method (b) Applications

Figure 1. **Dual-Process Distillation**. Our method distills deliberation into a feed-forward image generation process. When generating an image, we ask a VLM questions about that image and backpropagate the resulting gradient to update the weights of the image generator (a). The flexibility of a VLM allows us to implement many control tasks through visual prompts (b). We construct our method such that it supports off-the-shelf VLMs and image generators without special re-training.

Abstract

Prior methods for controlling image generation are limited in their ability to be taught new tasks. In contrast, vision-language models, or VLMs, can learn tasks in-context and produce the correct outputs for a given input. We propose a dual-process distillation scheme that allows feed-forward image generators to learn new tasks from deliberative VLMs. Our scheme uses a VLM to rate the generated images and backpropagates this gradient to update the weights of the image generator. Our general framework enables a wide variety of new control tasks through the same text-and-image based interface. We showcase a handful of applications of this technique for different types of control signals, such as commonsense inferences and visual prompts. With our method, users can implement multimodal controls for properties such as color palette, line weight, horizon position, and relative depth within a matter of minutes. Project page: https://dual-process.github.io.

1. Introduction

Current large language models have demonstrated competence across many domains and the ability to learn new tasks in-context [9], yet when trained multi-modally to jointly generate image and text, either fail to achieve the fidelity of image-only generation [12, 73, 75] or are inaccessible to academic experimentation [57]. Conversely, contemporary image generation models are near photorealistic, but can be frustratingly hard to communicate with. Inspired in part by cognitive science models, we propose a dual-process architecture [34, 69], combining a knowledge-rich multimodal language model with a visually precise image generator. The former is akin to a "cognitive" or "System 2" component, and the latter plays the role of a "reflex" or "System 1" module. This dual-process idea is also present in some early works in image generation. Classifier guidance [16], which uses an external classifier to steer image generation, can also be interpreted as embedding System 2 deliberation into a System 1 process. However, using this type of guidance requires specialized classifiers for each type of signal or control, and therefore is not easily extensible to new uses.

[*]Equal advising contribution.

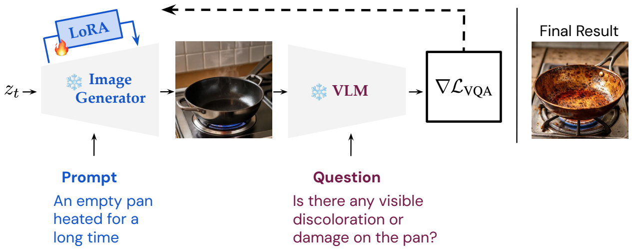


Figure 2. **Diagram of Dual-Process Distillation**. Given an input prompt and question to check, we feed the predicted clean image to the VLM to rate. Our method then backpropagates this loss to update LoRA [29] weights on the image generator.

On the other hand, vision-language models (VLMs) have proven to be generally applicable, capable of simulating many different discriminators within the same model. Simply by adjusting the input prompt, VLMs can perform different tasks such as optical character recognition [40], object detection [4], and image scoring [13, 44]. VLMs can not only be prompted with text but also multimodal inputs, or mixtures of images and text, such as visual prompts [11].

In this work, we propose a dual-process scheme that combines a deliberative VLM and feed-forward image generator, illustrated in Figure 1. We implement this scheme with gradient-based distillation and low-rank adaptation [29], where we update the weights of the image generator based on the VLM's ratings of its outputs. We demonstrate a handful of possible use cases of our method, such as visual prompting with a desired color palette, line weight, horizon position, or relative depth. We also evaluate our method on improving commonsense understanding and physical accuracy in image generation, where we find that our method significantly outperforms baselines such as prompt expansion.

2. Related Work

Classifier Guidance. Guidance methods use an external discriminator to introduce new capabilities missing in the base model. Dhariwal and Nichol [16] first demonstrated this idea with an ImageNet [15] classifier, and follow-up works explore other discriminators to support each newly proposed control task. These works investigate classic depth, segmentation, and object detection models for spatial controls [5, 35], as well as aesthetics and safety rating models for more abstract controls [66, 79]. Other works also look at re-using the image generator itself as a kind of discriminator [18, 42, 49, 76, 81]. Most similar to our work, Nguyen et al. [55] and Jiang et al. [32] investigate image captioning models [17, 41] for improved text-conditional generation. We focus on the novel use of modern VLMs as the most general form of these discriminators, given their ability to simulate many different tasks within the same model [2, 46]. Furthermore, we show that VLMs can newly support multimodal controls jointly defined with image and text, or visual prompts [11], compared with the image or text-only controls from prior work.

Inference-Time Search. Recent works in inference time search improve the output quality of language models by generating a substantial number of samples and post-hoc filtering with a verifier [8, 70, 82]. Inspired by these works, recent research in image generation investigates rejection or importance sampling during the denoising process [51, 68]. In contrast with these search methods, we compute a gradient through the verifier. This denser learning signal enables our method to discover long-tail images that would normally be extremely challenging to sample, for example variations of objects influenced by subtle physical laws, which we demonstrate on the CommonsenseT2I benchmark [22].

Model Fine-tuning. There exist fine-tuning methods that use the image generator's original objective to learn a narrower subdistribution. To condition the model on spatial controls like pose or depth images, ControlNet [87] trains on a paired dataset of real images and their extracted controls. To produce personalized outputs, Dreambooth [64] fine-tunes on a small image set representative of a given subject. Distillation methods circumvent the need for reference images by using outputs from the image generator itself as the supervision signal. Prior work has proposed such methods to speed up sampling [50, 65, 86], serve as an image prior for 3D generation [60, 78], or learn a concept direction [23]. Unlike these works, rather than optimizing with the original pixel-space objective, our method optimizes with a language-space objective derived from the VLM.

Reinforcement learning methods optimize with reward functions derived from some external model, similar to classifier guidance. Prior work explores policy optimization approaches to increase aesthetic preferences or prompt alignment of generated images [7, 21, 36, 77, 84, 85]. Most relevant to our work, Black et al. [7] uses VLM-generated captions to improve image-text faithfulness. In contrast, we use the VLM to produce numerical ratings, a more general formulation that enables novel use cases like visual prompting (see Sec. 4.3). Furthermore, our method is significantly cheaper than reinforcement learning methods, because we compute a gradient through the reward model. While prior work requires at least four hours of training on 50k samples [7], our method can enforce novel controls after optimizing for one minute on a single sample (see Sec. 4.4).

VQA Scoring. There is also a growing in interest in the usage of VLMs for scoring image-text alignment. Compared with CLIP [61], VLMs offer a more accurate and interpretable scoring framework. Recent works frame scoring as a VQA, or visual question answering, task [1], where one can ask the VLM different questions to evaluate the correctness of an image [13, 14, 38, 44, 85, 88]. Lin et al. [44] shows that answer probability, derived from the logits output by the VLM, can serve as a continuous metric, where lower and higher scores correlate with human judgement. Rather than using the VLM as an evaluation metric, we use it as a signal to improve the generated images themselves.

3. Dual-Process Distillation

3.1. Preliminaries

Many image generation models are formulated as a denoising process that transforms samples from a simple Gaussian distribution to the target data distribution. Among these, diffusion models [28, 71] have emerged as a popular theoretical framework. Rectified flows [3, 19, 45, 48] offer an alternative but equivalent theoretical formulation [25]. Specifically, the forward process is viewed as a linear interpolation between the data point x_0 and the noise ϵ, for $t \in [0, 1]$

$$z_t = (1 - t)x_0 + t\epsilon \tag{1}$$

Therefore, at sampling time, the reverse process is

$$z_{t'} = z_t + \hat{u} \cdot (t' - t) \tag{2}$$

for $t' < t$, where a neural network is often used to parameterize the estimated velocity $\hat{u} = g_\phi(z_t) = \hat{\epsilon} - \hat{x}_0$. This neural network is trained with a corresponding conditional flow matching objective [45]

$$\mathcal{L}_{\text{CFM}}(\mathbf{x}) = \mathbb{E}_{t \sim \mathcal{U}(0,1),\, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\|\hat{\mathbf{u}} - \mathbf{u}\|_2^2 \right] \tag{3}$$

While our experiments primarily focus on Flux [39], a state-of-the-art rectified flow model, our method is broadly applicable. We demonstrate its efficacy on both single-step and multi-step generators in Table 1.

3.2. Approach

Loss Formulation. In this work, we show that many visual control tasks can be formulated as visual question answering. As such, we propose a simple and general VQA loss, based on the same language modeling objective used for visual instruction tuning [46]. Our loss mainly differs in the object being optimized: visual instruction tuning optimizes the VLM, while our method optimizes the image input, through the image generator. More formally, our goal is to optimize the image generator G_ϕ such that it produces images where the VLM D_θ, when given a question S_q, is most likely to output the desired answer S_a. This can be written as the following autoregressive loss:

$$\mathcal{L}_{\text{VQA}} = -\sum_{i=1}^{L} \log D_\theta(S_{a,i} \mid G_\phi(z_t), S_q, S_{a,<i}) \tag{4}$$

We illustrate our full pipeline in Figure 2. Most importantly, we still compute gradients through the VLM, which gives a denser signal than reinforcement learning.

Clean Image Estimate. For multi-step models, the image generator may output noisy images that would otherwise be uninterpretable to an off-the-shelf VLM. In these cases, instead of directly feeding $G_\phi(z_t)$, we compute a clean image estimate, following the same insight as prior work [5, 72]. In the case of rectified flow models, this estimate can be computed by setting $t' = 0$ in Equation 2

$$\hat{x}_0 = z_t - G_\phi(z_t) \cdot t \tag{5}$$

Distillation Scheme. Here, we describe our procedure for distilling the VQA loss from the VLM into the image generator. For rectified flow models, we take a partially noised image and denoise it with the image generator in a single forward pass. However, we do not assume access to any reference images, but we also need images for the VLM to rate. We simply obtain these images by generating them with the image generator with no gradient flow. We then rate these images with the VLM, and backpropagate this loss to update LoRA [29] weights on the image generator. We optimize these weights for each (prompt, question, answer) triplet, on n seeds which correspond to unique images, where n is a hyperparameter to tune.

VLM Input Formatting. Finally, we discuss how we format the input to the VLM, which is crucial for achieving good results. VLMs and LLMs are notoriously sensitive to input format [67], where it is challenging to obtain meaningful ratings if the inputs are poorly formatted. However, as open-source models improve, we expect this to become less of an issue. First, we always use the default question answering template that was used to instruction tune the VLM. We primarily word questions such that they have a clear Yes or No answer, and append “Answer with Yes or No.” to the question. We find this yields interpretable probabilities for the extent to which the control was satisfied, simply by exponentiating the VQA loss, using a similar idea as Lin et al. [44]. Our system can also be extended to more open-ended question-answer pairs, but we anticipate this requires in-context learning examples to obtain a reliable answer format, which we leave for future work. Second, we also support visual prompts where we feed not only a question but also an instructional image, described further in Sec. 4.3. Rather than feeding the image instruction as a separate image, we choose to overlay it on top of the generated image from the image generator. Much like how such overlays are easier for people to rate, where they can directly compare the generated contents with the spatial instruction, we find that this format is more helpful for the VLM. Furthermore, this means that our system is also applicable to most VLMs, as it does not require multi-image support.

4. Experiments

4.1. Experimental Details

In all experiments, we use Flux Schnell [39] as our base image generator unless otherwise specified. By default we run Schnell as a single-step model, but we also validate our method in the multi-step setting in Table 1. For each experiment, we utilize the VLM that demonstrates the strongest performance for the given task. Our method uses VLMs off-the-shelf without additional fine-tuning, which makes it easy to switch between different models. We use LoRA [29] to optimize the weights, with a rank between 8-16 and alpha that is 5x the rank. A high alpha value is crucial; if it is set too low, the VLM cannot observe the full effect of the LoRA and rate images correctly. For optimization, we use Adam [37] with a learning rate of $5e^{-5}$ for up to 100 iterations. We optimize with $n = 100$ seeds unless otherwise specified. We run all experiments on a single 80GB Nvidia A100 GPU, which takes at most three minutes. Our method can run on as little as two 24GB Nvidia RTX 4090 GPUs when implemented on smaller models, e.g., switching out Flux Schnell [39] (12B params) for Sana [83] (1.6B params).

4.2. Commonsense Inferences

Setup. Even after training on millions of images, image generators still struggle with basic associations, or commonsense. For example, if a user prompts for a pen placed in a cup of water, many models do not know that the pen should appear bent due to the law of refraction. We evaluate whether our method can fix these knowledge gaps on CommonsenseT2I [22], a benchmark for evaluating commonsense in text-to-image models. Given an underspecified prompt, the image generator should generate an image that aligns with a held-out description of the expected inference. To improve the image generator, we feed the generated images to the VLM to verify whether they satisfy the inference, and optimize the weights to maximize this score. For the VLM, we use Idefics2 [40], a late-fusion VLM that achieves superior performance on OK-VQA, a knowledge-centric question answering benchmark [52]. We automatically generate the questions used to check these inferences using GPT4o [56]. We then compare our method against the base prompt and prompt expansion. Since our method is orthogonal and complementary to prompting, we can apply the learned weights on top. For prompt expansion we use GPT4o, following the same procedure as DALL-E 3 [6]. We compare performance following the same protocol as Fu et al. [22], which uses GPT4o to provide binary (yes or no) judgements of image correctness.

Our method improves on top of prompt expansion. We report quantitative results in Table 1. Both Flux Schnell and Dev perform poorly when prompted with the base prompt, with an average accuracy of 24% or less, demonstrating that the model inherently lacks commonsense understanding. Our method, when applied on top, improves performance by at least 20%. There are also many cases where prompt expansion is insufficient; our method improves prompt expansion by at least 11%. In these cases, the model completely fails to visually represent the phenomenon, regardless of the amount of explanation. This might be because the phenomenon is uncommon and underrepresented in the training data, and explicitly teaching the model with feedback is helpful.

Commonsense needs verification over suggestion. We show qualitative examples in Figure 3. The base prompt simply generates the primary object while ignoring other information that imply subtle inferences. Sometimes the expanded prompt introduces conflicting information (top half of Figure 3). Although we use GPT4o for both expansion and question generation, the model can produce better inferences when instructed to check for correctness rather than simply adding more detail. Finally, even if the expanded prompt explicitly includes the correct inference, the image generator can still fail (bottom half of Figure 3). Since our method is verification centric, it is better at enforcing the inference rather than leaving it as a suggestion.

4.3. Visual Prompting

Setup. The default interface of image generators is restricted to text, due to its reliance on static text embeddings [62, 63]. In contrast, VLMs can process arbitrary multimodal inputs, including visual prompts [10] where the instruction is jointly defined in image and text. This multimodal instruction is more conceptual, for example pointing at a spatial location and annotating what should be there. Unlike spatial control [87], where the image generator should exactly copy the structure of the input image, in visual prompting the model needs to reason about what the instruction is asking for. To implement control with visual prompts, we overlay the image instruction on top of the generated image, which is jointly fed with the text instruction to the VLM. We use Qwen2.5-VL [4] for our VLM, as it is specifically trained on data related to spatial understanding and object grounding. Next, we will discuss some specific use cases that can be implemented as visual prompts. See Table 6 of the Appendix for the image generator prompts of each example.

Color Palette. Here, our objective is to produce an image that adheres to a given color palette. To achieve this, we overlay the palette at the bottom of the generated image and query the VLM to verify that the colors match. We display qualitative examples in Figure 4. We control with three distinct color palettes, which we call *dark academia*, *pastel*, and *retro*. Evidently, our method is able to faithfully replicate the colors on naturalistic and artistic scenes with varying lighting conditions and 3D effects.

Line Weight. We now explore using our method to control line weight in cartoons. We draw a red line at the bottom

Generator	Method	Animal Behaviors	Biological Laws	Daily Items	Human Practices	Physical Laws	Avg
Flux Schnell (single-step)	Base Prompt	41.7	4.4	23.8	28.3	24.0	24.4
	Base Prompt + Ours	59.7	48.5	34.5	41.7	38.3	**44.5**
	Expanded Prompt	70.8	57.4	29.8	53.3	42.4	50.7
	Expanded Prompt + Ours	66.7	79.4	59.5	54.4	50.0	**62.0**
Flux Dev (multi-step)	Base Prompt	29.2	4.4	14.3	24.4	16.8	17.8
	Base Prompt + Ours	47.2	44.1	31.0	56.7	28.6	**41.5**
	Expanded Prompt	55.6	48.5	33.3	46.7	36.7	44.2
	Expanded Prompt + Ours	77.8	51.5	53.6	62.2	45.9	**58.2**

Table 1. **CommonsenseT2I Evaluation.** We evaluate on all 150 pairs from the benchmark and report the percentage of correct generations, both per-category and on average. We sample Flux Schnell as a single-step model (t=1) and Flux Dev as a multi-step model (t=28).

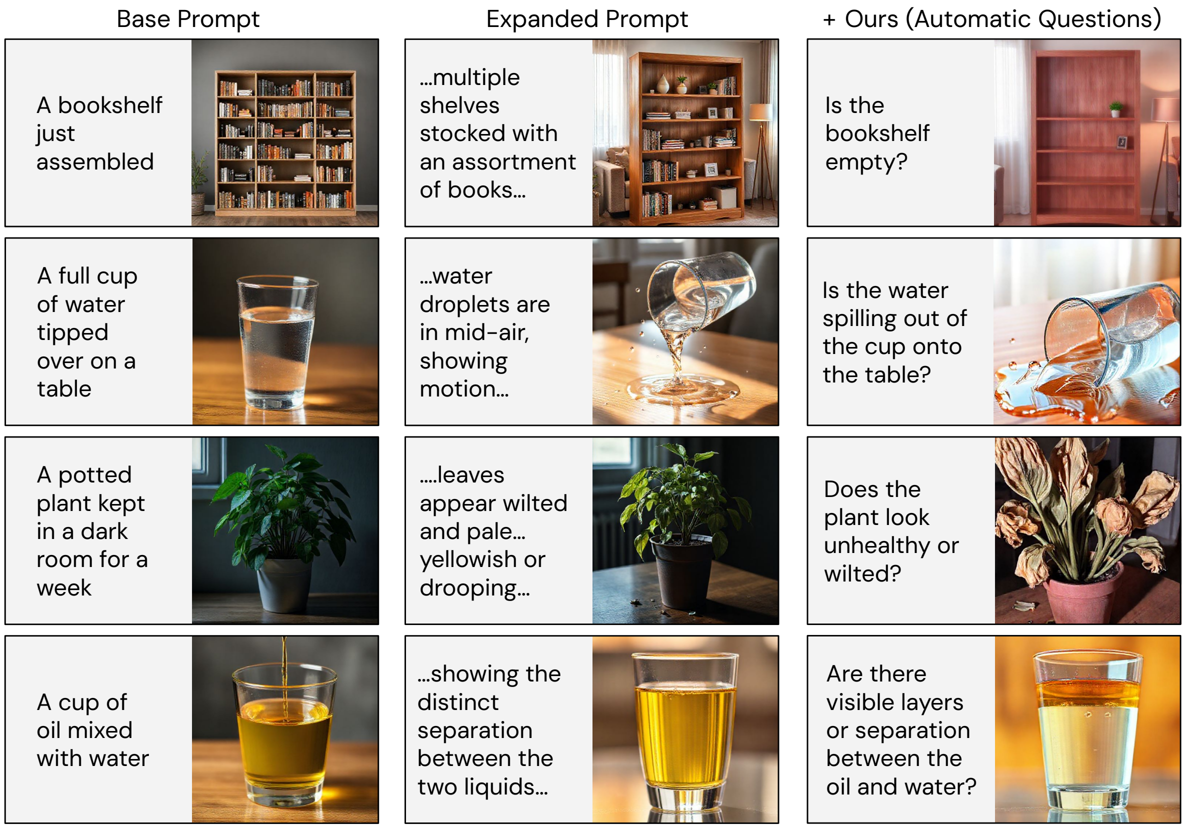


Figure 3. **Commonsense Inferences.** When the base prompt fails, prompt expansion can be unhelpful. Our method, which uses an automatically generated question to verify the inference, can be applied on top of prompting to fix these cases. The top two rows show cases where prompt expansion fails to capture implied properties given the short prompt, whereas the bottom two rows show cases where the expanded prompt contains the correct phenomenon but the generated image does not. Our method helps in both situations.

of the generated image and ask the VLM if the line is representative of the image's line weight. In Figure 5 we show qualitative examples, where our method can make the line weight thinner or thicker based on the overlaid line. Through visual prompting, we can create visual abstractions and bind different meanings based on the question. In this case we use lines to denote the desired edge thickness, whereas in the next section we use them to denote spatial position.

Horizon Position. Lines can also be used to control the position of the horizon. We overlay a red line on the generated image and ask the VLM to check if the horizon of the scene is aligned. We show qualitative examples in Figure 6, where our method is able to raise and lower the horizon according to the image instruction. These examples make it clear why overlaying the image instruction is helpful; one can check the spatial alignment simply by assessing the distance between the actual horizon and the red line.

Relative Depth. We now show how points can be used to

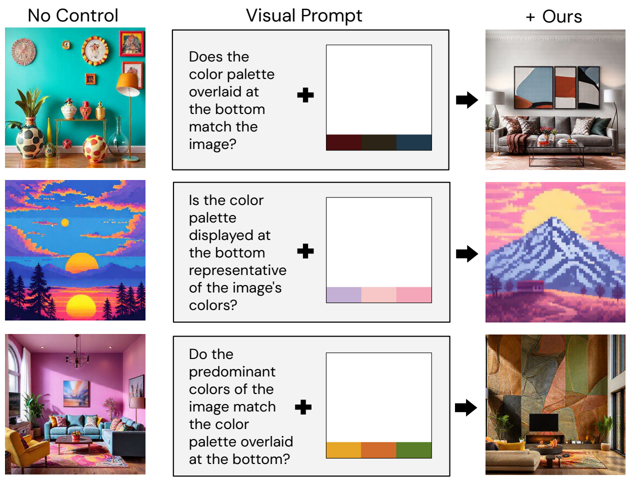


Figure 4. **Color Palette.** While the base image generators cannot be natively instructed with color palettes, our method can implement this control as a visual prompt. We simply overlay the palette at the bottom of the generated image and ask the VLM if the colors match.

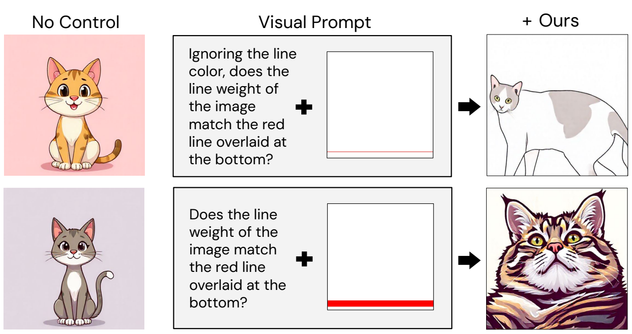


Figure 5. **Line Weight.** We use red lines to represent the desired line weight of a cartoon, through the thickness of the line.

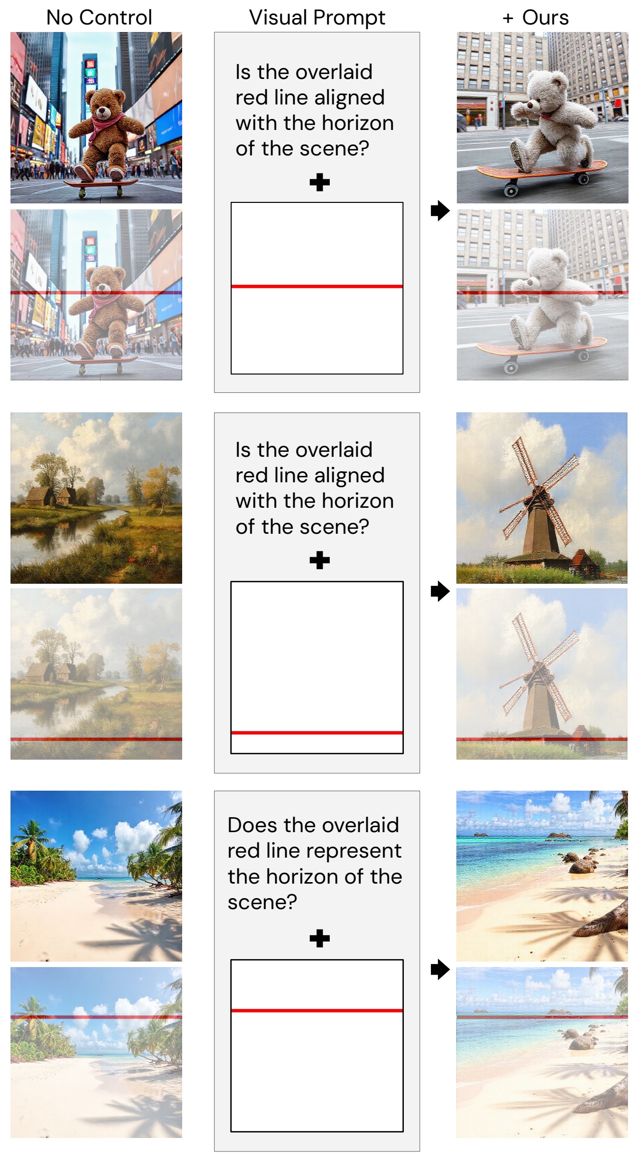


Figure 6. **Horizon Position.** We use red lines to specify the position of the horizon, or the boundary at which the earth and sky meet. The same visual abstraction can be bound with different meanings; lines can be used to specify both line weight and horizon position.

specify relative depth. We place two red points, labeled as Point A and Point B, and use the VLM to verify which point is in the foreground vs. background or which point is closer vs. further from the camera. As seen in Figure 7, our method is able to move around two subjects to enforce the control. In the first case, our method shrinks the dog to ensure that Point A lies on the background rather than the dog. The second case imposes the reverse control, where Point A should now lie on the foreground and Point B on the background. Our method enlarges the dog and creates empty space to its right to satisfy the visual prompt. Finally, in the last case our method is able to adjust the depth ordering of the two subjects, where both animals start at roughly the same distance from the camera then the dog is moved in front of the sheep. These examples also demonstrate how our method is able to follow truly multimodal instructions, where the VLM needs to distinguish between the two points by associating the red labels on the image instruction and the references to the labels in the question.

Visual Composition. We now show how visual prompting can be used to control composition with abstract art. This setting is inspired by the work of Piet Mondrian, who is best known for his abstract De Stijl paintings comprised of simple primary colors and lines. Mondrian paintings can be interpreted as highly abstracted Dutch landscapes, with elements like trees and the horizon reduced to vertical and horizontal lines [80]. In Figure 8 we show how our method can be used to perform the inverse problem and produce

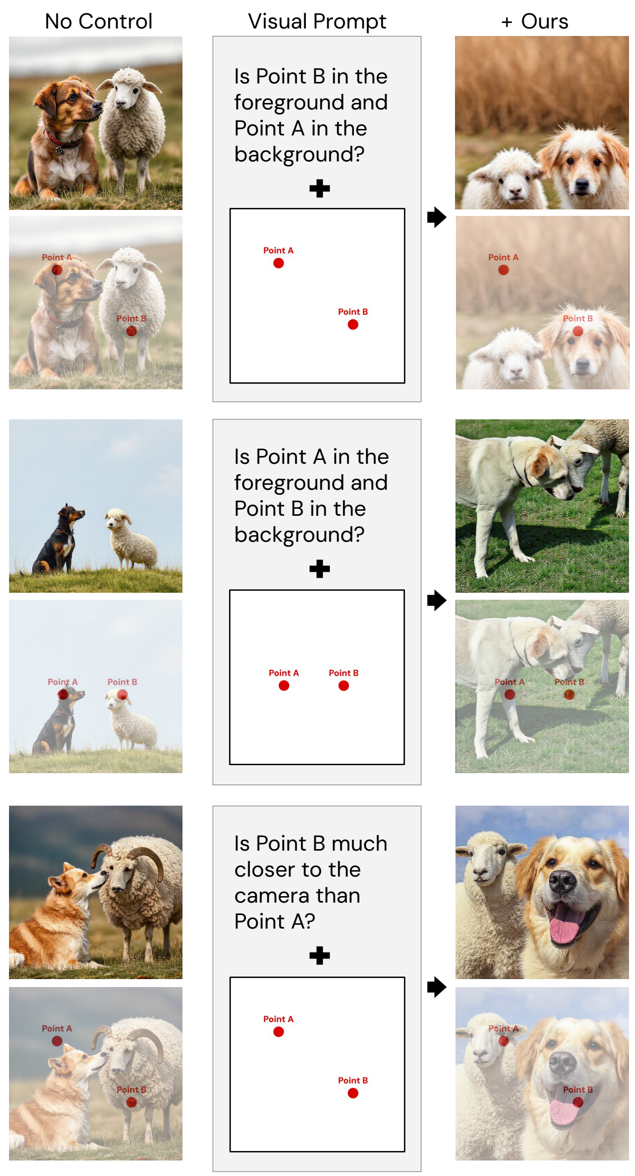


Figure 7. **Relative Depth.** We use two red points to specify relative depth ordering. We also annotate these points with text labels, which are referred to in the question.

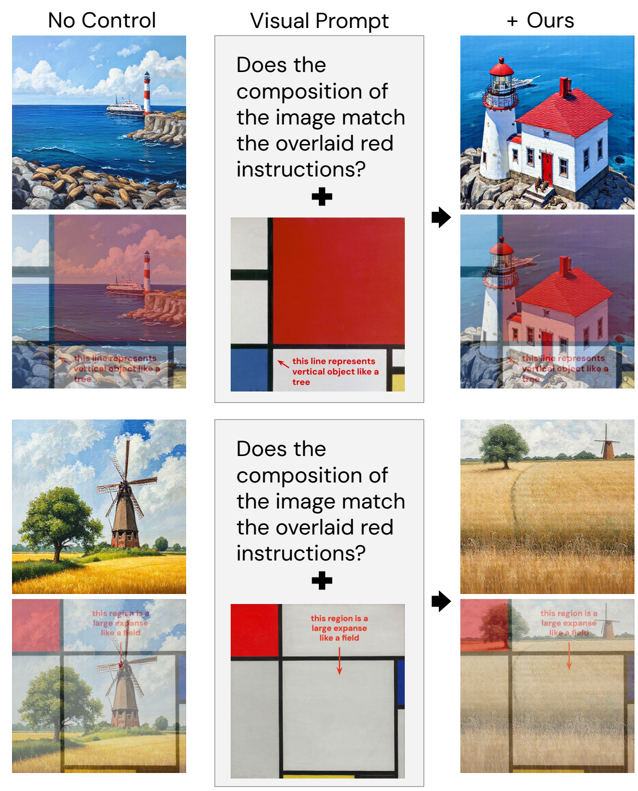


Figure 8. **Visual Composition.** We use abstract paintings from Piet Mondrian [53, 54] to control visual composition. We also annotate additional instructions for how to interpret the painting in red.

images that match the structure of these paintings. To make the task more clear, we annotate the paintings with additional instructions in red. We point to specific elements and instruct that *"this line represents vertical object like a tree"* or *"this region is a large expanse like a field."* Our method is able to follow these instructions and produce images with the desired composition, such as a wheat field in the boxed region or a lighthouse aligned with a vertical line. Note that there exists no paired data of abstract to landscape paintings, yet this transformation is possible when framed as a discriminative comparison task with our method.

4.4. Optimizing Latents vs. Weights

Setup. Although latent optimization may seem significantly cheaper than weight optimization, we show that this is not the case. We compare both approaches for guiding image generation, using our VLM loss (see Equation 4). Unlike the previous experiments, here we follow the setup of Eyring et al. [20], where the goal is to optimize a single seed such that the final image adheres to the input control. While this setup is originally designed to optimize latents, we can use the same gradients to instead optimize model weights, with near-identical implementations. For our VLM, we use Idefics2 [40].

Computational Cost Comparison. In Table 2 we compare the computational cost of latent and weight optimization. The gradient computation through the VLM, with respect to the clean image estimate $\hat{x}_0$, is identical for both methods. The cost slightly differs for the computation through the image generator, or the gradient of the clean image estimate $\hat{x}_0$ with respect to the input noise z_t. In this case, optimizing the weights increases the VRAM usage by 5%. However, weight optimization also benefits from compute amortization. After one minute of optimization upfront, every subsequent application of the weights is more than 200x faster.

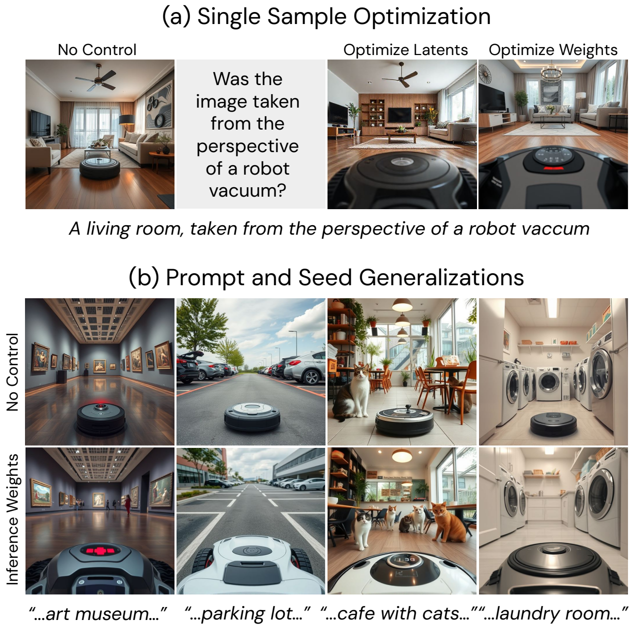


Figure 9. **Weight Generalizations.** We show that latent and weight optimization can achieve comparable results in a controlled single sample optimization setting in (a). We then show that these same weights can generalize to new seeds and prompts in a single inference pass, whereas latent optimization would need to be re-run with significant computational cost, in (b).

Metric	Latents	Weights
VLM VRAM	21GB	21GB
Image Generator VRAM	42GB	44GB
Optimization Time	56s	66s
Inference Time	56s	0.3s

Table 2. **Computational Cost Comparison.** We compare the memory and runtime cost of latent and weight optimization, averaged over 100 runs.

Weight Generalizations. In Figure 9, we show an example of controlling with an unusual visual perspective. We optimize latents and weights with the same gradients, on a single seed and single prompt, and they yield similar results (Figure 9a). However, these same weights can also generalize to unseen seeds and prompts, unlike latent optimization. The optimized weights can apply the same perspective control to new scenes in a single inference pass, making the camera appear lower to the ground and raising the horizon (Figure 9b). Note that this single-sample setting for weight optimization is unusual. Traditional fine-tuning methods typically optimize on multiple seeds and reference images [64], whereas we show it is possible to optimize the weights in an extremely data-limited setting on a single seed without any reference images.

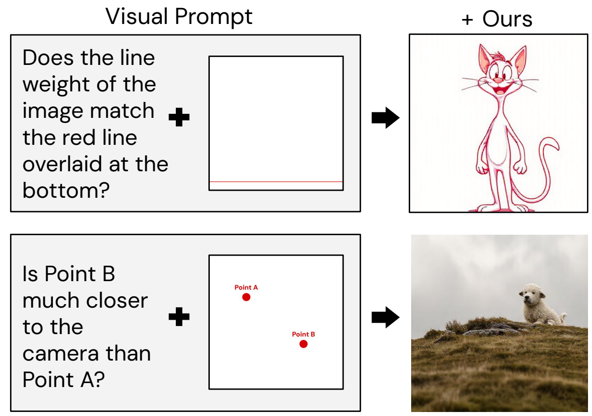


Figure 10. **Limitations.** Our method can misinterpret the visual prompt, for example changing line color when only line weight is specified or aggressively raising the ground level to satisfy a depth constraint.

5. Limitations

Since our method relies on feedback from a model, it is susceptible to reward hacking [24, 30, 58], where the VLM exploits ambiguities in the task specification to achieve a high reward. In Figure 10, we show two examples of such misinterpretations. When a red line is used to specify line width, our method may simultaneously change the line color from black to red. When the prompt is ambiguous, for example whether the points are bound to subjects or other elements in the scene, the model may produce an unintended but technically correct result. However, this same phenomenon could also be a useful tool for understanding how VLMs behave. One could use our method to mine adversarial images [26] where VLMs differ from human judgements; for example, cases where VLMs are blind to some visual property or are over-relying on certain visual cues. These automatically generated images could then be used as additional training data to improve the VLM's robustness.

6. Conclusion

We proposed a method that distills the deliberative ability of VLMs into image generators. Our method improves accuracy on challenging image generation tasks that require commonsense understanding, and provides new ways to control models through multimodal visual prompting. Our method is designed to be broadly applicable, and can utilize arbitrary off-the-shelf VLMs, selected based on their strengths for a given task. As open-source VLMs scale and improve in visual understanding and instruction following, we anticipate that our method can be used as a flexible and general framework for improving image generation.

Acknowledgements

We would like to thank Anastasis Germanidis, Yining Shi, Alexander Pan, Chung Min Kim, Boyi Li, and David Chan for helpful discussions and feedback. We would also like to thank the folks at Stochastic Labs, especially Vero Bollow, Alexander Reben, and Joel Simon, for previewing early prototypes of this work. Authors were supported in part by the NSF, DoD, and/or the Berkeley Artificial Intelligence Research (BAIR) industrial alliance program.

References

[1] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering. 2015. 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, 2022. 2

[3] Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2, 4, 1, 3

[5] Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Roni Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 3

[6] James Betker, Gabriel Goh, Li Jing, † TimBrooks, Jianfeng Wang, Linjie Li, † LongOuyang, † JuntangZhuang, † JoyceLee, † YufeiGuo, † WesamManassra, † PrafullaDhariwal, † CaseyChu, † YunxinJiao, and Aditya Ramesh. Improving image generation with better captions. 4, 3

[7] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[8] Bradley Brown, Jordan Juravsky, Ryan Saul Ehrlich, Ronald Clark, Quoc V Le, Christopher Re, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling, 2025. 2

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[10] Mu Cai, Haotian Liu, Siva Karthik Mustikovela, Gregory P. Meyer, Yuning Chai, Dennis Park, and Yong Jae Lee. Making large multimodal models understand arbitrary visual prompts. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 4

[11] Mu Cai, Haotian Liu, Siva Karthik Mustikovela, Gregory P Meyer, Yuning Chai, Dennis Park, and Yong Jae Lee. Vipllava: Making large multimodal models understand arbitrary visual prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12914–12923, 2024. 2

[12] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025. 1

[13] Jaemin Cho, Abhay Zala, and Mohit Bansal. Dall-eval: Probing the reasoning skills and social biases of text-to-image generation models. In *ICCV*, 2023. 2

[14] Jaemin Cho, Yushi Hu, Roopal Garg, Peter Anderson, Ranjay Krishna, Jason Baldridge, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian Scene Graph: Improving Reliability in Fine-Grained Evaluation for Text-to-Image Generation. In *ICLR*, 2024. 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 2

[16] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2021. Curran Associates Inc. 1, 2

[17] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):677–691, 2017. 2

[18] Dave Epstein, Allan Jabri, Ben Poole, Alexei A. Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. 2023. 2

[19] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 3

[20] Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. Reno: Enhancing one-step text-to-image models through reward-based noise optimization. *Neural Information Processing Systems (NeurIPS)*, 2024. 7

[21] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok:

reinforcement learning for fine-tuning text-to-image diffusion models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2023. Curran Associates Inc. 2

[22] Xingyu Fu, Muyu He, Yujie Lu, William Yang Wang, and Dan Roth. Commonsense-t2i challenge: Can text-to-image generation models understand commonsense? In *First Conference on Language Modeling*, 2024. 2, 4, 1, 7

[23] Rohit Gandikota, Joanna Materzyńska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. *arXiv preprint arXiv:2311.12092*, 2023. 2

[24] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, pages 10835–10866. PMLR, 2023. 8

[25] Ruiqi Gao, Emiel Hoogeboom, Jonathan Heek, Valentin De Bortoli, Kevin P. Murphy, and Tim Salimans. Diffusion meets flow matching: Two sides of the same coin. 2024. 3

[26] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021. 8

[27] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 2

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, pages 6840–6851, 2020. 3

[29] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 2, 3, 4

[30] Dario Amodei Jack Clark. Faulty reward functions in the wild. `https://openai.com/index/faulty-reward-functions`, 2016. 8

[31] Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9307–9315, 2024. 1, 2

[32] Dongzhi Jiang, Guanglu Song, Xiaoshi Wu, Renrui Zhang, Dazhong Shen, Zhuofan Zong, Yu Liu, and Hongsheng Li. Comat: Aligning text-to-image diffusion model with image-to-text concept matching. In *NeurIPS*, 2024. 2

[33] Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. *CVPR*, 2023. 2

[34] Daniel Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011. 1

[35] Gyeongnyeon Kim, Wooseok Jang, Gyuseong Lee, Susung Hong, Junyoung Seo, and Seungryong Kim. Dag: Depth-aware guidance with denoising diffusion probabilistic models. *arXiv preprint arXiv: Arxiv-2212.08861*, 2022. 2

[36] Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[37] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 4

[38] Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. Viescore: Towards explainable metrics for conditional image synthesis evaluation, 2024. 2

[39] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024. 3, 4

[40] Hugo Laurencon, Leo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2, 4, 7, 1, 3

[41] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2

[42] 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. Featured Certification. 2

[43] 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. 1, 2

[44] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part IX*, page 366–384, Berlin, Heidelberg, 2024. Springer-Verlag. 2, 3

[45] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[46] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2, 3, 1

[47] Tianyang Liu. System prompt of chatgpt. `https://leoii22.com/blog/2024/system-prompt-of-gpt4/`, 2024. 3

[48] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[49] Grace Luo, Trevor Darrell, Oliver Wang, Dan B Goldman, and Aleksander Holynski. Readout guidance: Learning control from diffusion features. 2024. 2

[50] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2023. 2

[51] Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Inference-time scaling for diffusion models beyond scaling denoising steps, 2025. 2

[52] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 4

[53] Piet Mondrian. Composition, 1929. 7

[54] Piet Mondrian. Composition with red, blue, and yellow, 1930. 7

[55] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug and play generative networks: Conditional iterative generation of images in latent space. In *CVPR*, 2017. 2

[56] OpenAI. Gpt-4o system card, 2024. 4, 1

[57] OpenAI. Introducing 4o image generation, 2025. 1

[58] Alexander Pan, Kush Bhatia, and Jacob Steinhardt. The effects of reward misspecification: Mapping and mitigating misaligned models. In *International Conference on Learning Representations*, 2022. 8

[59] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022. 2

[60] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[62] 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. 4

[63] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. 4

[64] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 2, 8

[65] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. 2024. 2

[66] Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[67] Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *ICLR*, 2024. 3

[68] Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models, 2025. 2

[69] Steven A. Sloman. The empirical case for two systems of reasoning. *Psychological Bulletin*, 119:3–22, 1996. 1

[70] Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[71] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2256–2265, Lille, France, 2015. PMLR. 3

[72] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 3

[73] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2024. 1

[74] Gemma Team. Gemma 3. 2025. 1

[75] Shengbang Tong, David Fan, Jiachen Zhu, Yunyang Xiong, Xinlei Chen, Koustuv Sinha, Michael Rabbat, Yann LeCun, Saining Xie, and Zhuang Liu. Metamorph: Multimodal understanding and generation via instruction tuning, 2024. 1

[76] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. New York, NY, USA, 2023. Association for Computing Machinery. 2

[77] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8228–8238, 2024. 2

[78] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *CVPR*, 2023. 2

[79] Jonathan Whitaker, 2023. 2

[80] Wikipedia contributors. Piet mondrian — Wikipedia, the free encyclopedia, 2025. [Online; accessed 7-March-2025]. 6

[81] Tsung-Han Wu, Long Lian, Joseph E Gonzalez, Boyi Li, and Trevor Darrell. Self-correcting llm-controlled diffusion models. 2024. 2

[82] Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[83] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformers, 2024. 4

[84] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: learning and evaluating human preferences for text-to-image generation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 15903–15935, 2023. 2

[85] Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, Jiayan Teng, Zhuoyi Yang, Wendi Zheng, Xiao Liu, Ming Ding, Xiaohan Zhang, Xiaotao Gu, Shiyu Huang, Minlie Huang, Jie Tang, and Yuxiao Dong. Visionreward: Fine-grained multi-dimensional human preference learning for image and video generation, 2024. 2

[86] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024. 2

[87] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2, 4

[88] Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. Gpt-4v(ision) as a generalist evaluator for vision-language tasks, 2023. 2