LLM-grounded Diffusion: Enhancing Prompt Understanding of Text-to-Image Diffusion Models with Large Language Models

Long Lian[1] **Boyi Li**[1] **Adam Yala**[1,2] **Trevor Darrell**[1]
[1]UC Berkeley [2]UCSF

Abstract

Recent advancements in text-to-image generation with diffusion models have yielded remarkable results synthesizing highly realistic and diverse images. However, these models still encounter difficulties when generating images from prompts that demand spatial or common sense reasoning. We propose to equip diffusion models with enhanced reasoning capabilities by using off-the-shelf pretrained large language models (LLMs) in a novel two-stage generation process. First, we adapt an LLM to be a text-guided layout generator through in-context learning. When provided with an image prompt, an LLM outputs a scene layout in the form of bounding boxes along with corresponding individual descriptions. Second, we steer a diffusion model with a novel controller to generate images conditioned on the layout. Both stages utilize frozen pretrained models without any LLM or diffusion model parameter optimization. We validate the superiority of our design by demonstrating its ability to outperform the base diffusion model in accurately generating images according to prompts that necessitate both language and spatial reasoning. Additionally, our method naturally allows dialog-based scene specification and is able to handle prompts in a language that is not well-supported by the underlying diffusion model.

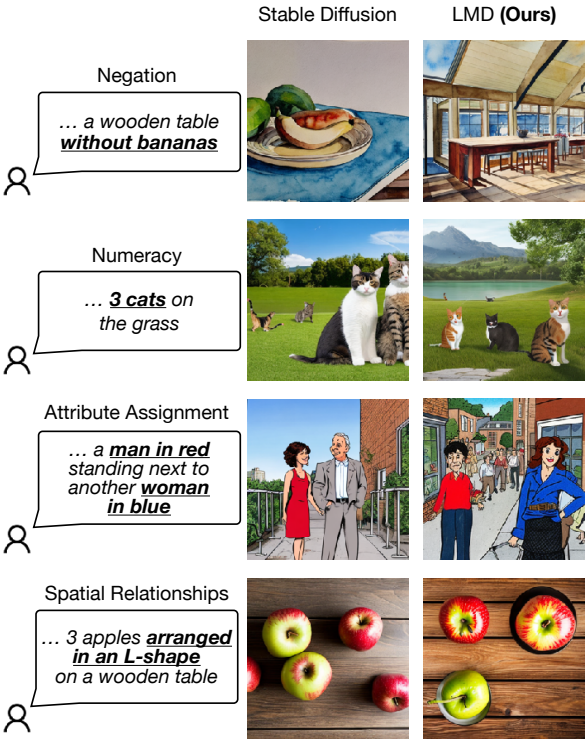


Figure 1: Our method enhances the prompt understanding ability of text-to-image diffusion models (Rombach et al., 2022).

1 Introduction

The field of text-to-image generation has witnessed notable advancements, especially with the adoption of diffusion models. These models have showcased remarkable capabilities in generating highly realistic and diverse images in response to textual prompts. However, despite their impressive capabilities, diffusion models, such as Stable Diffusion (Rombach et al., 2022), often struggle to accurately follow the prompts when spatial or common sense reasoning is required. Fig. 1 lists four scenarios in which Stable Diffusion falls short in generating images that accurately correspond to the given prompts.

One possible solution to address this issue is of course to gather a vast multi-modal dataset comprising intricate captions and train a new diffusion model. This approach comes with significant costs: It is time-consuming and expensive to train both large language models (LLMs) and diffusion models. To efficiently solve this problem with minimal cost, we instead equip diffusion models with enhanced spatial and common sense reasoning by using pretrained LLMs in a novel two-stage generation process.

In the first stage of our method, we adapt an LLM to be a text-guided layout generator through in-context learning. Given an image prompt, this

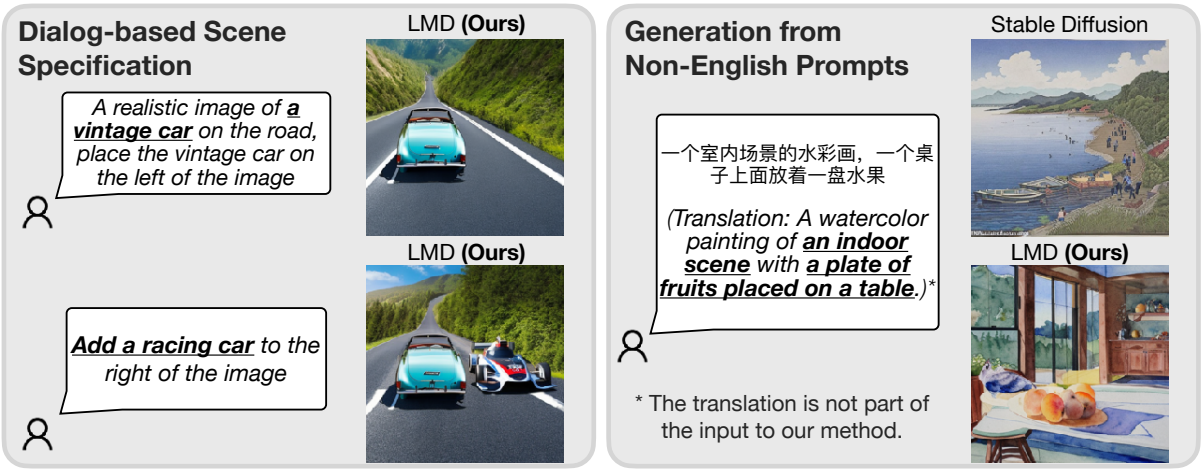


Figure 2: Incorporating an LLM for prompt understanding, our method is naturally able to perform dialog-based scene specification and generation from prompts in a language (Chinese in the example above) that the underlying diffusion model does not support.

LLM outputs scene layouts in the form of captioned bounding boxes as well as a background prompt.

In the second stage, we propose a novel layout-conditioned image generation method, which generates images by conditioning on the layout generated in the first step. In contrast to the previous region control method (Bar-Tal et al., 2023) which steers a diffusion model towards *semantics* in certain regions, we enable precise control over object *instances* in designated regions. Our layout-conditioned image generator uses a frozen diffusion model (e.g., Stable Diffusion) under the hood and does not involve parameter gradient computation.

Notably, both stages utilize frozen pretrained models, making our method applicable to off-the-shelf LLMs and diffusion models *without any LLM or diffusion model parameter optimization*.

In addition to enhanced prompt understanding, our incorporation of LLMs also allows dialog-based multi-round scene specification and image generation from prompts in a language that the diffusion model does not support (Fig. 2).

Our overall approach, dubbed **L**arge **L**anguage **M**odel-grounded **D**iffusion (LMD), can generate high-quality images from prompts that require complex spatial and common sense reasoning. We demonstrate that a diffusion model, augmented with an LLM using LMD, outperforms its base diffusion model in terms of several tasks that require reasoning to follow the prompts.

In this work, we make three main contributions:

1. We adapt large language models for enhanced prompt understanding of diffusion models. We introduce a novel two-stage text-to-image generation process LMD that consists of text-to-layout generation and subsequent layout-to-image generation.
2. We propose a novel training-free image generation method that steers a diffusion model to generate images conditioned on bounding box layouts.
3. We provide evidence that by enhancing diffusion models with LLMs, our method not only enables image generation from prompts that demand advanced reasoning capabilities but also allows dialog-based scene specification and multilingual text-to-image generation.

2 Related Work

Diffusion Models for Image Generation. Diffusion models are a class of powerful generative models that learn the data distribution of complex datasets. During the forward process, noise is added to an image to the input data $\mathbf{x}_0$ for T steps, until the resulting vector $\mathbf{x}_T$ is almost distributed according to a standard Gaussian distribution. A neural network learns to predict the added noise that can be subtracted from a standard Gaussian distribution during inference. DDPM (Ho et al., 2020) shows high-quality image synthesis results using diffusion probabilistic models. DDIM (Song et al., 2020) proposes a sampling strategy that allows image generation from fewer denoising steps as well as a way to invert the denoised sample $\mathbf{x}_0$ to $\mathbf{x}_T$ in a special case of the sampling strategy.

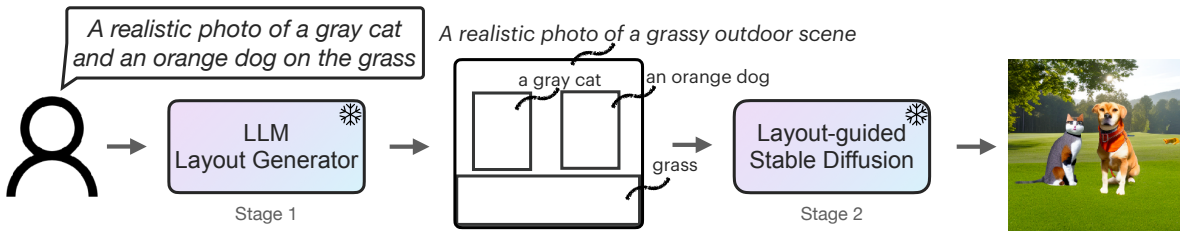


Figure 3: We propose LMD, a text-to-image generative model with a novel two-stage generation process: **1)** An LLM layout generator first takes an image prompt and outputs an image layout in the form of bounding boxes with individual descriptions. **2)** A novel layout-guided stable diffusion generates the image conditioned on the layout. Both stages use frozen pretrained models, which makes our method applicable to off-the-shelf LLMs and other diffusion models.

Classifier-free guidance (Ho and Salimans, 2022) allows conditioning the diffusion model without an externally-trained classification model. Latent diffusion and stable diffusion (Rombach et al., 2022) propose to denoise in the latent space, allowing high-quality generation in high resolution. We refer readers to the appendix for a preliminary introduction to diffusion models.

Reasoning from Large language models. The inclusion of grounding information in large language models (Li et al., 2022) frequently enhances their reasoning capabilities and is proved advantageous across various applications. Chain-of-thought (Wei et al., 2022) introduces a simple chain-of-thought prompting method to enable reasoning abilities to emerge naturally in large language models, with a series of intermediate reasoning steps. Not limited to language-only models, many multimodal models also benefit from integrating large language models with visual models for advanced interactive performance. BLIP (Li et al., 2023) proposes a generic and efficient pretraining strategy that bootstraps vision-language pre-training from off-the-shelf frozen pre-trained image encoders and frozen large language models. Chameleon (Lu et al., 2023) synthesizes programs to compose various tools, including LLM models, off-the-shelf vision models, web search engines, Python functions, and rule-based modules tailored to user interests. Flamingo (Alayrac et al., 2022) is a family of Visual Language Models. The results show that Flamingo can achieve state-of-the-art results with few-shot learning for many visual reasoning tasks such as visual question-answering and captioning tasks. Beyond, Rozanova et al. (2021) finds that document-based models can learn a reasonable amount of spatially relevant features that make them transferable to the UI grounding task. Ghanimifard and Dobnik (2019) reveals that the language model possesses the capability to differentiate between the functional and geometric biases of spatial relations through encoding, despite lacking access to visual features of the scene.

Conditioned Image Generation. Conditioned image generation aims to generate images that follow a specific pattern or instruction. We classify these methods into visual-guided image generation and text-to-image generation. On the one hand, visual-guided image generation creates new content based on given prior visual knowledge such as pose, segmentation map or stroke (Vinker et al., 2022), etc. SPADE (Park et al., 2019) and BlobGAN (Epstein et al., 2022) synthesize photorealistic images by a given layout. ControlNet (Zhang and Agrawala, 2023) is a training-based method that controls pretrained large diffusion models with additional dense 2D input conditions. MultiDiffusion (Bar-Tal et al., 2023) allows region control for semantics in image generation and shares a similar task formulation with our layout-to-image generator. However, MultiDiffusion exhibits unsatisfying control in generating the specified instances, as multiple regions of similar semantics may be treated as one instance by the diffusion model. On the other hand, recently text-to-image generation has made significant advancements in a short period of time. DALL-E (Ramesh et al., 2021), Imagen (Saharia et al., 2022), and (Rombach et al., 2022) enable high-quality image generation with textual input descriptions, the generated content tends to exhibit subpar performance when it comes to many reasoning tasks, including generative numeracy.

Diffusion Model-based Image Editing. Prompt-to-prompt (Hertz et al., 2022) shows powerful image editing capabilities of diffusion models

given a pair of prompt descriptions. Instruct Pix2Pix (Brooks et al., 2022) distills prompt-to-prompt to edit the image given a text instruction. However, the edited instruction cannot guarantee the generated accuracy of spatial content. DiffEdit (Couairon et al., 2022) proposes to use DDIM inversion for a similar image editing task. Our text-conditioned layout generator is inspired by the intuitions for cross-attention maps in (Hertz et al., 2022) and DDIM inversion in (Couairon et al., 2022). However, we are working on text-to-image generation, a task that stands apart as it solely relies on a single provided textual prompt, distinguishing it fundamentally from previous image editing literature.

3 LLM-Grounded Diffusion

We present our method LMD in detail in this section. LMD focuses on a standard text-to-image generation setting: given text prompt $\mathbf{y}$, generate image $\mathbf{x}_0$, potentially by denoising from initial noise $\mathbf{x}_T$. Our method generates an image in two stages: text-to-layout generation (Section 3.1) and layout-to-image generation (Section 3.2).

3.1 LLM-based Layout Generation

Given a text prompt $\mathbf{y}$, the text-to-layout generator generates the layout of an image. An image layout description includes three components: 1) coordinates of the bounding box for each foreground object in *(x, y, width, height)* format, 2) a text prompt for the content of each bounding box, 3) a text prompt that describes the background in a simple and concise fashion that diffusion model's text encoder could easily understand.

Prompting. Our prompt to an LLM includes three parts:

1. **Task specification:**
   Your task is to generate the bounding boxes for the objects mentioned in the caption, along with a background prompt describing the scene.
2. **Supporting details:**
   The images are of size 512x512... Each bounding box should be in the format of ...
3. **Attitude towards guessing:**
   If needed, you can make reasonable guesses.

In-context learning. We provide the LLM with manually curated examples after the task description. Through examples, we convey the exact format of our layout generation (i.e., include the three components clearly) and provide details of the instance specification.

An example is shown as follows:
Caption: A watercolor painting of two pandas eating bamboo in a forest
Objects: [('a panda eating bambooo', [30, 133, 212, 226]), ('a panda eating bambooo', [262, 137, 222, 221])]
Background prompt: A watercolor painting of a forest

We ensure two key points in designing our examples: We list out one box for each object (e.g., if we specify four objects in an example prompt, we leave four boxes for the object in the example reference). In addition, we leave no foreground objects that are already specified in the box to the background, so that foreground objects are all kept under the control of our layout-guided image generator (Section 3.2).

After the prompt and the examples, we ask the LLM to perform completion[1]:
Caption: [user's input image prompt]
Objects: [start of LLM completion]

The LLM is supposed to generate the name of each instance along with its location and size in a bounding box format, and then it is expected to output the background prompt without the specified foreground instance. We refer the readers to the appendix for our full prompt.

3.2 Layout-guided Stable Diffusion

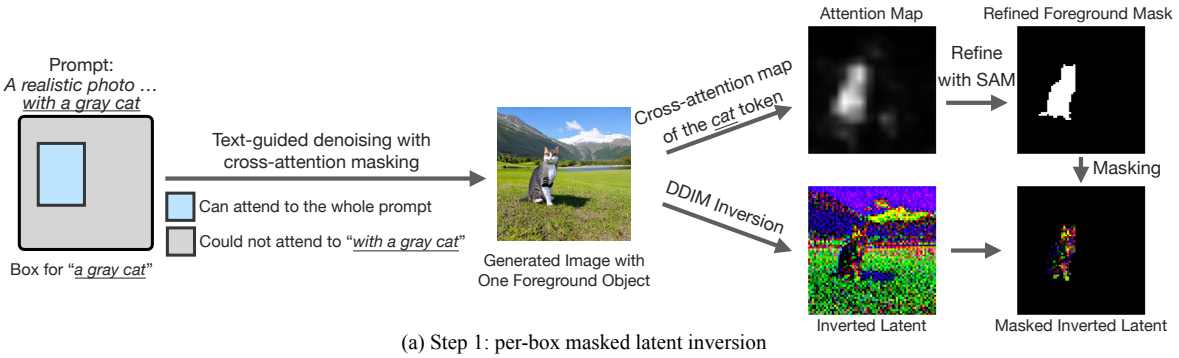
We use the layout generated by the LLM to condition the diffusion model for the overall image generation. Two key steps of our layout-guided stable diffusion are: 1) generating masked latent inversion for each box, and 2) composing the latent inversion as well as generating the corresponding background.

Per-box masked latent inversion. We process one foreground box at a time. For each of the foreground objects, we first generate a single-object image with customized text conditioning.

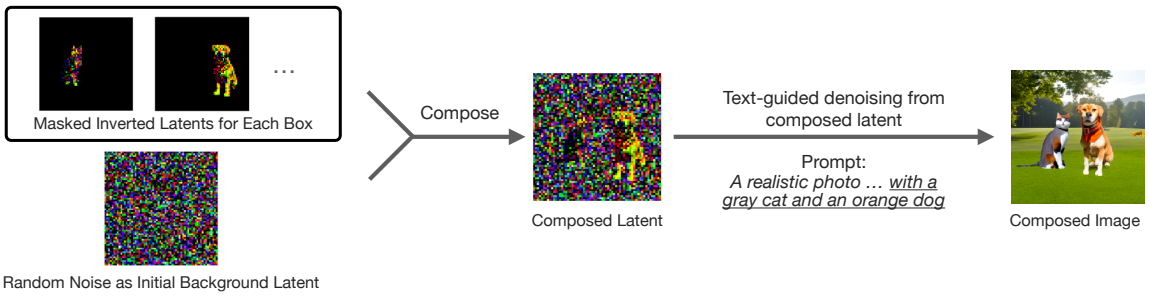
We use a composite prompt for generation: *"[background prompt] with [box content]"* (e.g., *"a realistic image of an indoor scene with a cat"*).

To ensure the object is generated at the expected location and has roughly the size of the

[1]We use Chat Completion API to query the LLM for completion.



(a) Step 1: per-box masked latent inversion



(b) Step 2: foreground-aware composed image generation

Figure 4: Our novel layout-guided stable diffusion component generates images based on the layout obtained from an LLM. Our layout-guided image generation process has two steps: masked noise inversion for each box and the subsequent composed image generation.

box, we independently control the image-to-text cross-attention maps of the diffusion model for pixels inside the box and outside the box[2]. Leaving the image-to-text cross-attention maps intact for pixels inside the box, we attenuate the cross-attention from pixels outside the box to the text tokens *"with [box content]"* so that the object is correctly placed.

Then we obtain the cross-attention map that describes the affinity from each pixel to the text tokens that correspond to the *"[box content]"*, which presents a rough saliency mask of the object in the generated image, inspired by (Hertz et al., 2022). We use segment anything model (SAM) (Kirillov et al., 2023) to further refine the quality of the mask by querying from the pixel location with the max saliency value.

Next, we apply DDIM inversion (Song et al., 2020) to obtain the latent of the generated single-object image. Denoising the inverted latent will reproduce an image that closely resembles the generated single-object image. We perform element-wise multiplication to the refined foreground mask from SAM and the inverted latent, creating a masked inverted latent that describes the foreground.

[2]The term *pixels* refers to latent values as we operate in a latent space of the diffusion model.

Foreground-aware image generation. To place all instances into an appropriate background, we first randomly generate standard Gaussian noise as the background latent. When de-noised, this background latent will generate an image that reflects the textual conditioning.

Then we place foreground masked latents onto randomly generated latent by simply replacing the background latent with foreground latent in the parts indicated by the mask and the box location. Specifically, we make sure the center of the box specification aligns with the center of the outer box of the foreground object mask. This creates a composed latent for the subsequent generation.

We then generate the output image by denoising from step T to step 0. To ensure consistency between the foreground and the background, we first generate the background with the foreground fixed and then refine the whole image in a denoising pass, controlled by a hyperparam r ranging from 0 to 1.

From T to $(1-r)T$ steps, we only allow modifying background from the composed latent. Since the background part is originally a standard Gaussian, this allows the background to evolve according to the textual prompt and the foreground latents, i.e. be *foreground-aware*. In these steps, the foreground part of the composed latent is directly taken from the corresponding step in the latent inversion

process, inspired by (Couairon et al., 2022). From rT to 0 steps, we allow modifying the whole image for more coherent results, as the layout is already generated, inspired by the bootstrapping technique in (Bar-Tal et al., 2023). The textual prompt for the diffusion model is simply *"[background prompt] with [box 1 content], [box 2 content], etc."* r indicates how much we allow the foreground to change to get more coherent results. A large r allows more coherent results but can deviate from the instance specification. The final generation is thus expected to both adhere to the foreground object specification and come with a coherent background.

4 Visualizations

We present visualizations of our method along with the generation of Stable Diffusion 2.1, which is the base model that we use in the layout-guided image generation stage. As shown in Fig. 5, our two-stage text-to-image generation method can follow the prompts that require language and spatial reasoning much better than our base model which performs text-to-image generation in one stage. In addition, our generated images also closely align with the layouts generated by our text-to-layout generator, shown in the middle column of Fig. 5.

In addition to comparing with our base model Stable Diffusion, we also compare with MultiDiffusion (Bar-Tal et al., 2023), a method that allows image generation conditioned on semantic masks. Since MultiDiffusion is proposed to leverage masks as the input in addition to the text prompt, we use the layout generated in the first step of our method and convert the labeled boxes into semantic masks (Fig. 6(b)). We present the first four generated images from the three methods with no random seed selection. Stable Diffusion does not adhere to the number of balls in the prompt (Fig. 6(c)). MultiDiffusion generates images with semantics that match the specifications provided in the layout (Fig. 6(d)). However, it does not have fine-grained control over each instance. As shown in Fig. 6(e), our method correctly generates three plastic balls in three out of four images, showing better instance-level control in the generation process.

5 Additional Capabilities of LMD

Using an LLM as a prompt parser that interpreter, LMD naturally gains capabilities in addition to enhanced understanding and reasoning in text-to-image generation. Given an LLM that supports multi-round dialog (e.g., GPT-3.5 or GPT-4), we can naturally provide additional information or clarification to the LLM by querying the LLM after the first layout generation in the dialog and generate images with the updated layout in the subsequent response from the LLM. For example, a user could request to add an object to the scene Fig. 2 (left) or change the existing objects in location or descriptions. Furthermore, by giving an example of a non-English prompt with a layout and background description in English during in-context learning[3], LMD accepts inputs of non-English prompts and will generate layouts, with descriptions of boxes and the background in English for subsequent layout-to-image generation. As shown in Fig. 2 (right), this allows generation from prompts in a language that the underlying diffusion models do not support.

Benchmarks	Accuracy (%)
Negation	100%
Generative Numeracy	93%
Attribute Assignment	100%
Spatial Relationships	98%

Table 1: Our LLM-based layout generator is able to handle prompts that require several types of reasoning with high accuracy.

6 Evaluating Text-guided Layout Generation

Setting. We propose an evaluation method of our approach that four benchmarks: negation, generative numeracy, attribute assignment, and spatial reasoning. Negation and generative numeracy involve generating a specific number of objects. Attribute assignment involves assigning the right attribute to the right object. Spatial reasoning involves understanding words that describe the relative locations of objects. For each prompt type, we compose a prompt for LLM as input and check whether the output layout matches the LLM prompt. Specifically, we pick 10 common object types from the COCO dataset (Lin et al., 2014)[4]. We design our benchmark so that the LLM is queried 100 times in each benchmark, generating 100 layouts for evaluation. For negation and generative numeracy bench-

[3]Specifically, we simply translate the prompt of our last in-context learning example, keeping the layout intact.

[4]backpack, book, bottle, bowl, car, cat, chair, cup, dog, and laptop

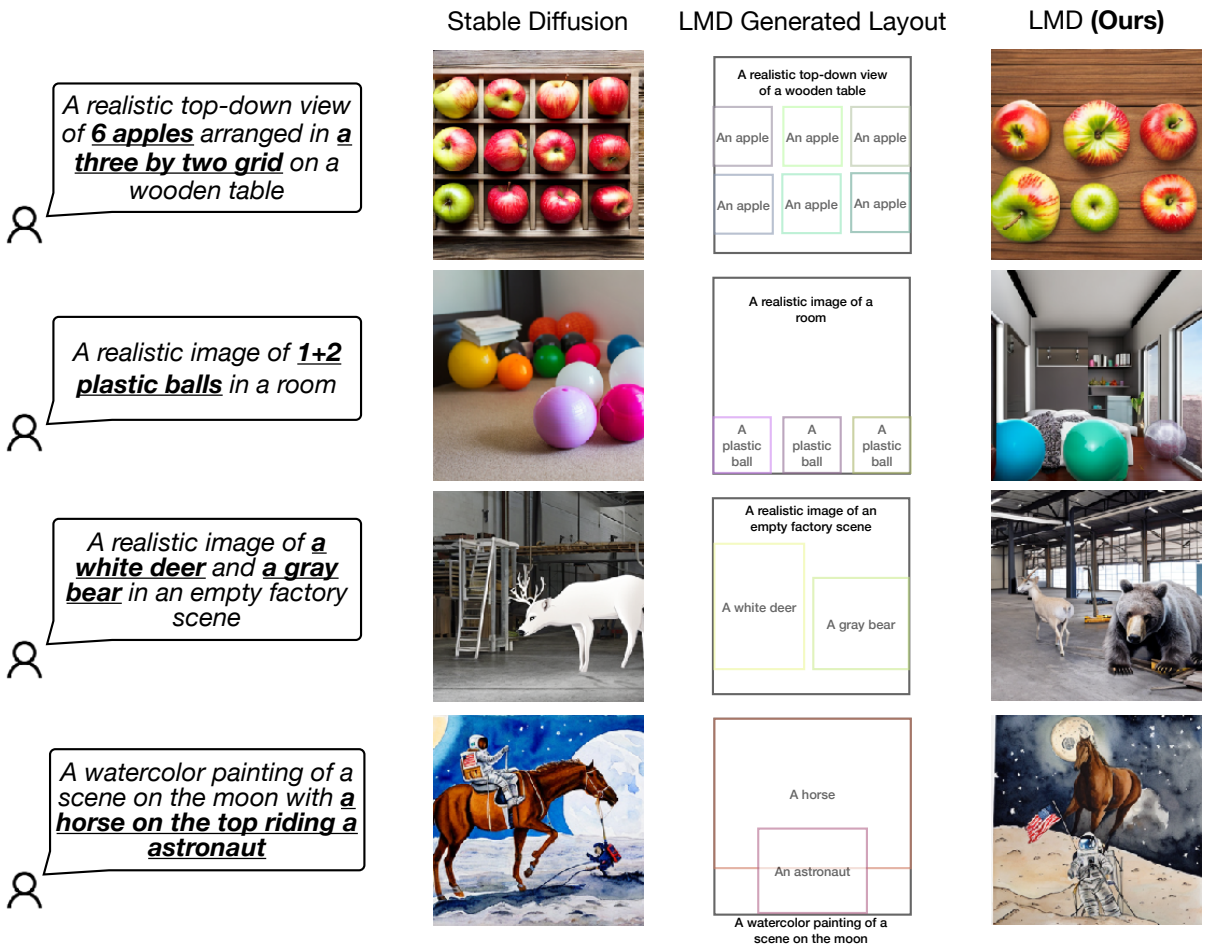


Figure 5: Our method outperforms the base text-to-image diffusion model (Rombach et al., 2022) in terms of correctly following the prompts that require spatial and language reasoning. Best viewed in color and zoom in.

mark, we prompt the LLM to generate a layout of a scene with some number of a certain object or without a certain object. Then we count the number of objects and consider the layout to be correct if the number of the object of that particular type matches the one in the prompt, with the number ranging from 1 to 5. For attribute assignment, we prompt the LLM to generate a object of a color and another object of another color, with a similar type of evaluation. For the spatial relationship benchmark, we generate an object at a certain location and another object at an opposite location (left/right and top/bottom). We then check the spatial coordinates of the boxes to ensure the layout exactly matches the prompt. All benchmarks are evaluated on `gpt-3.5-turbo`. We refer readers to the appendix for the prompts in the setting.

Results. Table 1 shows the results of our LLM-based layout generator which is designed to handle prompts requiring reasoning. The model was able to achieve high accuracy in generating the layouts that match the requirements of the prompts, reaching 100% when it comes to handling negation and attribute assignment prompts. This demonstrates the excellent ability of the model in understanding the absence of an object and assigning attributes to the correct object. Moreover, the model performed remarkably well when dealing with generative numeracy prompts, reaching an accuracy of 93% for ranges 1-5. The failure cases are mostly because the model outputs a plural form of the object (e.g., chairs) as a box rather than individual boxes, which we expect to be able to fix with better prompting. With a 98% accuracy on spatial relationship benchmark, our model also achieves highly accurate generation when prompted with keywords that describe the locations of objects.

7 Limitations

Since we use an off-the-shelf large language model for text-to-layout generation without any fine-tuning, the LLM only has limited knowledge of

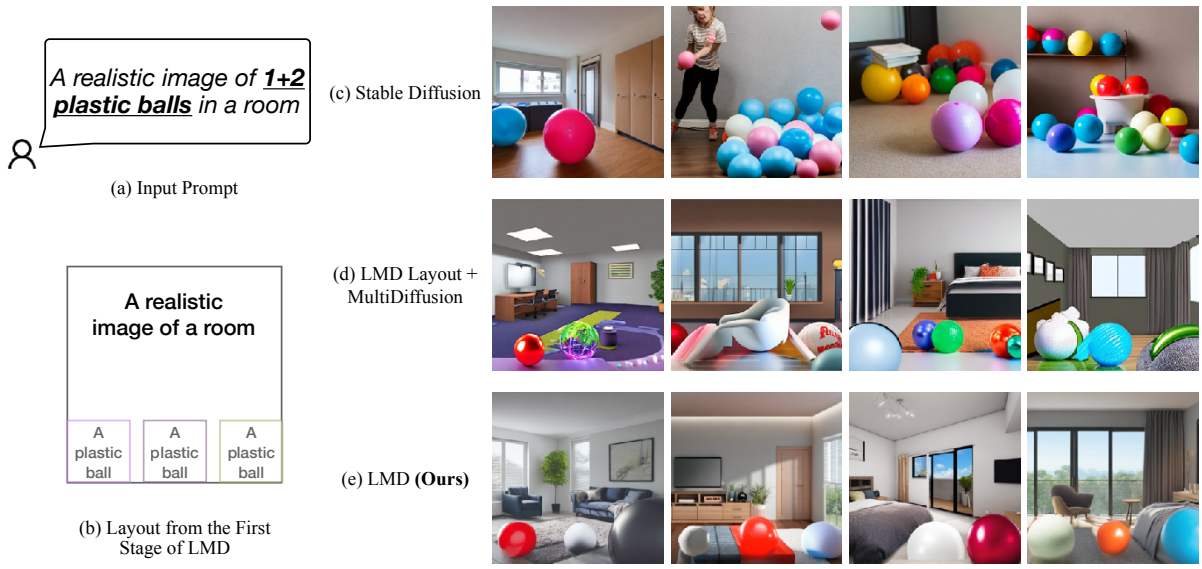


Figure 6: Our layout-guided image generator has better instance-level control compared to MultiDiffusion (Bar-Tal et al., 2023). While MultiDiffusion only specifies the semantic regions, our layout-guided image generator specifies one instance at the location of each box. Our method correctly generates exactly one ball for a box in three out of four attempts.

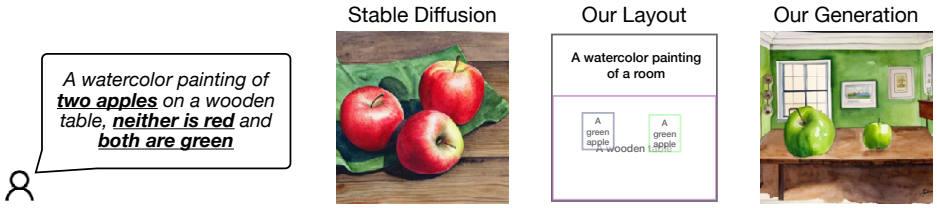


Figure 7: A failure case of our method is generating disproportional objects. Our method sometimes generates objects that are disproportional to the background due to ambiguity in the layout specification. The layout generated by our text-to-layout generator is feasible for a close-up image, but the layout-to-image model interprets it as a scene viewed from far away.

the preferences of the diffusion model from the provided examples and thus may output a layout that is hard to generate by the diffusion model. Furthermore, our layout is expressed in the format of a set of bounding boxes with a background prompt, which does not explicitly convey the viewpoint information and may confuse the diffusion model. For example, the generated layout in Fig. 7 is feasible for a close-up image, but the diffusion model generates an image viewing from far away, causing the objects and background to be disproportionate. We believe a better format for expressing the layout and specifically fine-tuning an LLM to perform layout generation from layouts obtained by running bounding box detectors on images will alleviate this problem.

Since our layout-guided image generator generates one instance at a time, there is no coordination between instances for style coherency except that the text-to-layout generator will ensure boxes of similar sizes and coherent placements. This may cause unintended style variations between instances in one image. Furthermore, partially occluded instances in the single-instance generation stage may lead to incomplete foreground objects in the composed generation.

Our method also inherits other issues, such as biases in the generation, from Stable Diffusion, which is also presented in previous works such as (Luccioni et al., 2023).

8 Summary

We enhance the capabilities of text-to-image diffusion models to understand textual prompts that require language and spatial reasoning. We propose a novel two-stage generation process that involves

text-guided layout generation through in-context learning and the subsequent layout-to-image generation. We justify the superiority of our proposed method by demonstrating its ability to outperform the base diffusion model in accurate image generation from prompts that demand language and spatial reasoning.

A Appendix

A.1 Preliminaries on diffusion models

Diffusion models are a class of powerful generative models that learn the data distribution of complex datasets. During the forward process, noise is added to an image to the input data $\mathbf{x}_0$ for T steps, until the resulting vector $\mathbf{x}_T$ is almost distributed according to a standard Gaussian distribution. In the reverse process, a diffusion model iteratively subtracts a predicted noise vector from $\mathbf{x}_T$ to transform it into a sample that resembles the real data in the training dataset. The reverse process is often referred to as "denoising". We refer readers to (Luo, 2022) for a more in-depth introduction to diffusion models.

DDPM. (Ho et al., 2020). The denoising process of denoising diffusion probabilistic models starts with the initial noise vector sampled from a standard Gaussian noise vector $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$. During training, a neural network with parameter θ learns to predict the added noise for the forward process by minimizing the training objective:

$$\mathcal{L} = ||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\theta(\mathbf{x}_t, t)}||^2 \tag{1}$$

At inference time, for each of the T denoising steps, DDPM predicts the noise $\boldsymbol{\epsilon}$ and then obtains $\mathbf{x}_{t-1}$ from $\mathbf{x}_t$:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\prod_{i=1}^{t}\alpha_i}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\Big) + \sigma_t \mathbf{z} \tag{2}$$

where $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$, α_t and σ_t are parameterized by a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$ that controls the size of the denoising step.

DDIM (Song et al., 2020). Denoising diffusion implicit models are a generalization to DDPM which allows sampling with fewer iterations. DDIM applies the following update rule:

$$\mathbf{x}_{t-1} = \sqrt{\alpha_{t-1}}\Big(\frac{\mathbf{x}_t - \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\alpha_t}}\Big) + \sigma_t \boldsymbol{\epsilon}_t \tag{3}$$

Note that DDIM shares the same training procedure with DDIM, which means we can take a model trained with DDPM objective and perform faster sampling using DDIM. When σ_t is set to 0, which is the case for our setting, the denoising becomes deterministic given $\mathbf{x}_T$. Assuming each step only performs small changes to $\mathbf{x}_t$, DDIM sampling can also be conducted in the reverse order, as proposed in (Song et al., 2020; Dhariwal and Nichol, 2021)

to obtain a vector $\mathbf{x}'_T$ from $\mathbf{x}_0$ so that $\mathbf{x}'_T$, when de-noised, gives back $\mathbf{x}'_0$ that is close to $\mathbf{x}_0$. The process of inverting the $\mathbf{x}_0$ to get $\mathbf{x}'_T$ is called DDIM inversion (Mokady et al., 2022).

Latent Diffusion and Stable Diffusion (Rombach et al., 2022). While DDIM and DDPM are denoising images from raw pixel space, latent diffusion proposes to denoise in the latent space for high-quality generation in high resolution. Specifically, before the denoising process, the image is encoded by an encoder, and a decoder decodes the generated $\mathbf{x}_0$ in the latent space to an output image.

Latent diffusion also proposes a conditioning scheme that allows generating samples with other modalities (e.g., text) as the condition. The condition is realized through cross-attention layers (Vaswani et al., 2017) that attend from latent locations in U-Net (Ronneberger et al., 2015) feature maps to the encoded condition (e.g., text features from a CLIP text encoder).

Stable diffusion models are large text-to-image models trained on large multi-modal datasets using the techniques proposed for latent diffusion.

A.2 Prompts for Evaluating Text-guided Layout Generation

For the negation benchmark, we use the prompt *A realistic photo of a scene without [object name].*

For generative numeracy, we use the prompt *A realistic photo of a scene with [number] [object name].*

For attribute assignment, we use the prompt *A realistic photo of a scene with [modifier 1] [object name 1] and [modifier 2] [object name2]*, where the two modifiers are randomly chosen from colors (red, orange, yellow, green, blue, purple, pink, brown, black, white, gray).

For the spatial relationship benchmark, we use the prompt *A realistic photo of a scene with [object name 1] on the [location] and [modifier 2] [object name2] on the [opposite location]*, where the location is chosen from left, right, top, and bottom.

A.3 Our full prompt

Our full prompt is listed in Table 2.

```
You are an intelligent bounding box generator. I will provide you with a caption for a photo, image, or painting. Your task is to generate the bounding boxes for the objects mentioned in the caption, along with a background prompt describing the scene. The images are of size 512x512, and the bounding boxes should not overlap or go beyond the image boundaries. Each bounding box should be in the format of (object name, [top-left x coordinate, top-left y coordinate, box width, box height]) and include exactly one object. Do not put objects that are already provided in the bounding boxes into the background prompt. If needed, you can make reasonable guesses. Please refer to the example below for the desired format.

Caption: A realistic image of four skiers standing in a line on the snow near a palm tree
Objects: [('a skier', [5, 152, 139, 168]), ('a skier', [278, 192, 121, 158]), ('a skier', [148, 173, 124, 155]), ('a palm tree', [404, 180, 103, 180])]
Background prompt: A realistic image of an outdoor scene with snow

[Additional Examples]

Caption: [User Prompt]
Objects:
```

Table 2: Our full prompt to the LLM for layout generation. LLM starts completion from "`Objects:`."

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736.

Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. 2023. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2022. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2022. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*.

Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794.

Dave Epstein, Taesung Park, Richard Zhang, Eli Shechtman, and Alexei A Efros. 2022. Blobgan: Spatially disentangled scene representations. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, pages 616–635. Springer.

Mehdi Ghanimifard and Simon Dobnik. 2019. What a neural language model tells us about spatial relations. In *Proceedings of the Combined Workshop on Spatial Language Understanding (SpLU) and Grounded Communication for Robotics (RoboNLP)*, pages 71–81.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. Segment anything. *arXiv:2304.02643*.

Boyi Li, Rodolfo Corona, Karttikeya Mangalam, Catherine Chen, Daniel Flaherty, Serge Belongie, Kilian Q Weinberger, Jitendra Malik, Trevor Darrell, and Dan Klein. 2022. Does unsupervised grammar induction need pixels? *arXiv preprint arXiv:2212.10564*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. 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.

Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. 2023. Chameleon: Plug-and-play compositional reasoning with large language models. *arXiv preprint arXiv:2304.09842*.

Alexandra Sasha Luccioni, Christopher Akiki, Margaret Mitchell, and Yacine Jernite. 2023. Stable bias: Analyzing societal representations in diffusion models. *arXiv preprint arXiv:2303.11408*.

Calvin Luo. 2022. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2022. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*.

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

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer.

Julia Rozanova, Deborah Ferreira, Krishna Dubba, Weiwei Cheng, Dell Zhang, and Andre Freitas. 2021. Grounding natural language instructions: Can large language models capture spatial information? *arXiv preprint arXiv:2109.08634*.

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. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yael Vinker, Ehsan Pajouheshgar, Jessica Y Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. 2022. Clipasso: Semantically-aware object sketching. *ACM Transactions on Graphics (TOG)*, 41(4):1–11.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.