Compositional 3D Scene Generation using Locally Conditioned Diffusion

Ryan Po
Stanford University
rlpo@stanford.edu

Gordon Wetzstein
Stanford University
gordon.wetzstein@stanford.edu

Abstract

Designing complex 3D scenes has been a tedious, manual process requiring domain expertise. Emerging text-to-3D generative models show great promise for making this task more intuitive, but existing approaches are limited to object-level generation. We introduce ***locally conditioned diffusion*** *as an approach to compositional scene diffusion, providing control over semantic parts using text prompts and bounding boxes while ensuring seamless transitions between these parts. We demonstrate a score distillation sampling–based text-to-3D synthesis pipeline that enables compositional 3D scene generation at a higher fidelity than relevant baselines.*

1. Introduction

Traditionally, 3D scene modelling has been a time-consuming process exclusive to those with domain expertise. While a large bank of 3D assets exists in the public domain, it is quite rare to find a 3D scene that fits the user's exact specifications. For this reason, 3D designers often spend hours to days modelling individual 3D assets and composing them together into a scene. To bridge the gap between expert 3D designers and the average person, 3D generation should be made simple and intuitive while maintaining control over its elements (e.g., size and position of individual objects).

Recent work on 3D generative models has made progress towards making 3D scene modelling more accessible. 3D-aware generative adversarial networks (GANs) [21, 50, 29, 10, 31, 23, 5, 44, 27, 19, 38, 4, 13, 54, 33, 53] have shown promising results for 3D object synthesis, demonstrating elementary progress towards composing generated objects into scene [32, 51]. However, GANs are specific to an object category, limiting the diversity of results and making scene-level text-to-3D generation challenging. In contrast, text-to-3D generation [35, 22, 48] using diffusion models can generate 3D assets from a wide variety of categories via text prompts [1, 39, 40]. Existing work leverages strong 2D image diffusion priors trained on internet-scale data, using a single text prompt to apply a global conditioning on rendered views of a differentiable scene representation. Such methods can generate high-quality object-centric generations but struggle to generate scenes with multiple distinct elements. Global conditioning also limits controllability, as user input is constrained to a single text prompt, providing no control over the layout of the generated scene.

We introduce locally conditioned diffusion, a method for compositional text-to-image generation using diffusion models. Taking an input segmentation mask with corresponding text prompts, our method selectively applies conditional diffusion steps to specified regions of the image, generating outputs that adhere to the user specified composition. We also achieve compositional text-to-3D scene generation by applying our method to a score distillation sampling–based text-to-3D generation pipeline. Our proposed method takes 3D bounding boxes and text prompts as input and generates coherent 3D scenes while providing control over size and positioning of individual assets. Specifically, our contributions are the following:

- We introduce **locally conditioned diffusion**, a method that allows greater compositional control over existing 2D diffusion models.
- We introduce a method for compositional 3D synthesis by applying locally conditioned diffusion to a score distillation sampling–based 3D generative pipeline.
- We introduce key camera pose sampling strategies, crucial for compositional 3D generation.

2. Related work

2D diffusion models. Advances in large-scale 2D diffusion models trained on internet-scale data [7, 30, 39, 36, 37, 40, 43] have allowed generation of high-quality images that stay accurate to complex text prompts. While text-conditioned diffusion models excel at reproducing the semantics of a prompt, compositional information is usually ignored. Variants of existing methods [39] instead condition their models with semantic bounding boxes. This change

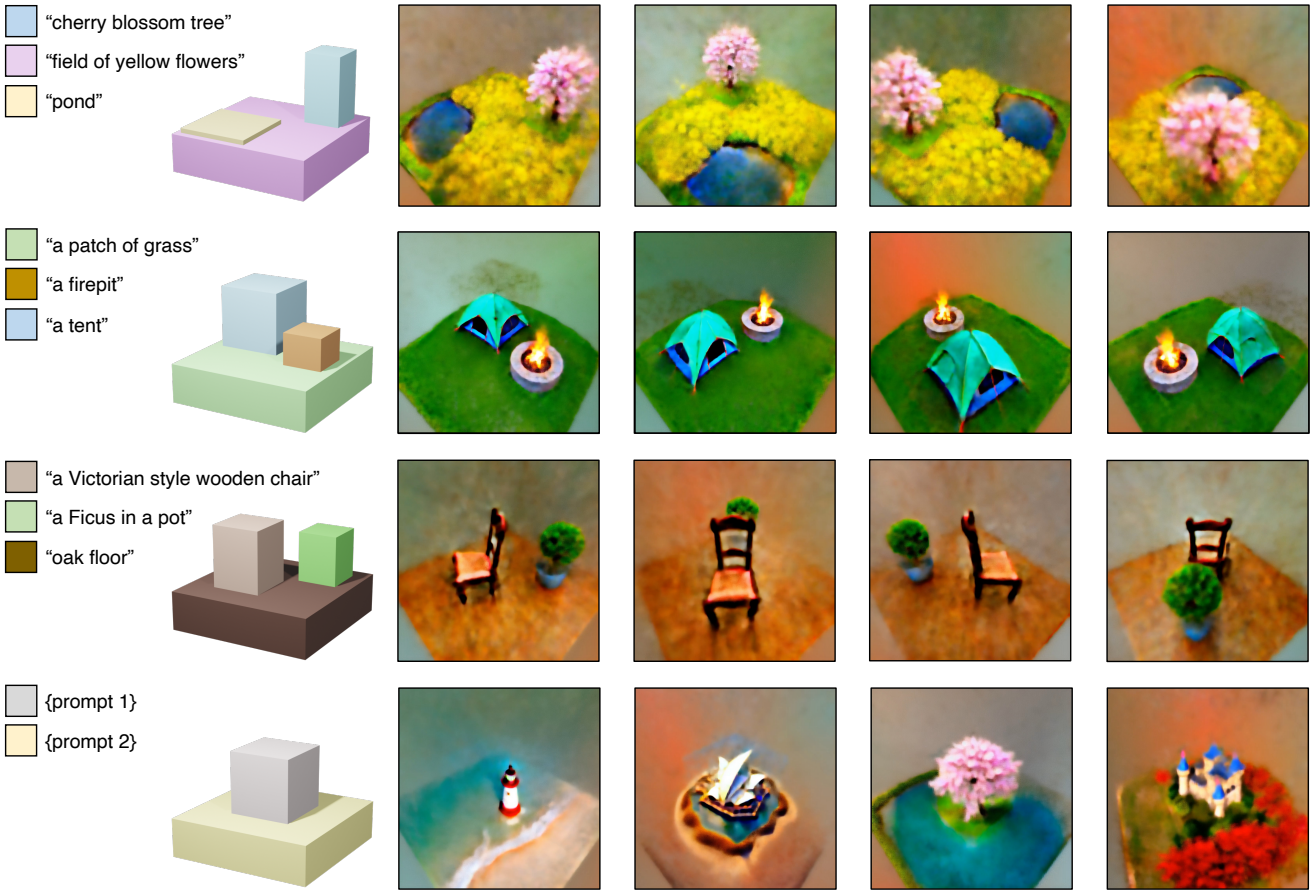

Figure 1. **Results of our method.** Given user-input bounding boxes with corresponding text prompts, our method is able to generate high-quality 3D scenes that adhere to the desired layout with seamless transitions. Our locally conditioned diffusion method blends multiple objects into a single coherent scene, while simultaneously providing control over the size and position of individual scene components. Text prompts for bottom row (from left): (1) "a lighthouse" and (2) "a beach"; (1) "the Sydney Opera House" and (2) "a desert"; (1) "a cherry blossom tree" and (2) "a lake"; (1) "a small castle" and (2) "a field of red flowers". Videos of our results can be found in the supplementary materials.

allows greater control over the composition of the generated image. However, bounding-box-conditioned models must be trained with annotated image data [3]. These datasets are often much more limited in size, which restricts the diversity of the resulting diffusion model. Our locally conditioned diffusion approach leverages pre-trained text-conditioned 2D diffusion models to generate high-quality images with better compositional control without restricting the complexity of user-provided text-prompts.

Compositional image generation. Recent work found that Energy-Based Models (EBMs) [8, 9, 20, 11, 12] tend to struggle with composing multiple concepts into a single image [8, 25]. Noting that EBMs and diffusion models are functionally similar, recent work improves the expressivity of diffusion by borrowing theory from EBMs. For example, [25] achieves this by composing gradients from denoisers conditioned on separate text-prompts in a manner similar to classifier-free guidance as proposed by [7]. Existing work, such as Composable-Diffusion [25], however, apply composition to the entire image, offering no control over the position and size of different concepts. Our locally conditioned diffusion approach selectively applies denoising steps over user-defined regions, providing increased compositional control for image synthesis while ensuring seamless transitions.

Text-to-3D diffusion models. Recent advances in 2D diffusion models have motivated a class of methods for performing text-to-3D synthesis. Existing methods leverage 2D diffusion models trained on internet-scale data to achieve text-to-3D synthesis. Notably, DreamFusion [35] with Imagen [40], Score Jacobian Chaining (SJC) [48] with StableDiffusion [39] and Magic3D [22] with eDiff-I [1] and StableDiffusion [39]. Previous methods [35, 48, 22, 28] perform 3D synthesis by denoising rendered views of a differentiable 3D representation. This process is coined Score Distillation Sampling (SDS) by the authors of DreamFu-

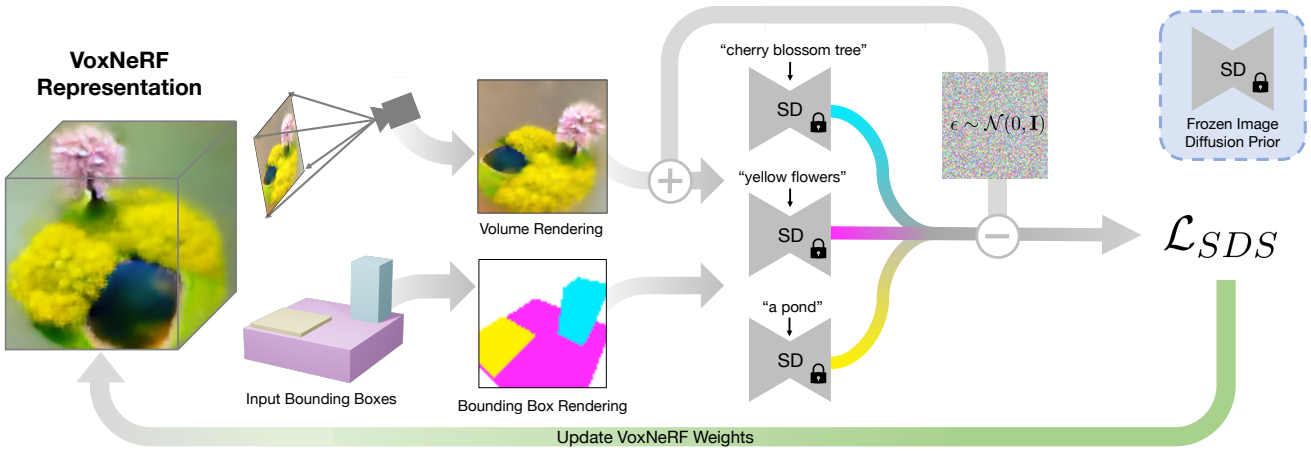

Figure 2. **Overview of our method.** We generate text-to-3D content using a score distillation sampling–based pipeline. A latent diffusion prior is used to optimize a Voxel NeRF representation of the 3D scene. The latent diffusion prior is conditioned on a bounding box rendering of the scene, where a noise estimation on the image is formed for every input text prompt, and denoising steps are applied based on the segmentation mask provided by the bounding box rendering.

sion [35]. Intuitively, SDS ensures that all rendered views of the 3D representation resemble an image generated by the text-conditioned 2D diffusion model. Current methods are able to generate high quality 3D assets from complex text prompts. However, they are unable to create 3D scenes with specific compositions. Our proposed method enables explicit control over size and position of scene components.

3. Diffusion preliminaries

Recent work has shown that diffusion models can achieve state-of-the-art quality for image generation tasks [7]. Specifically, Denoising Diffusion Probabilistic Models (DDPMs) implement image synthesis as a denoising process. DDPMs begin from sampled Gaussian noise x_T and apply T denoising steps to create a final image x_0. The forward diffusion process q is modelled as a Markov chain that gradually adds Gaussian noise to a ground truth image according to a predetermined variance schedule $\beta_1, \beta_2, \ldots, \beta_T$

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}\right) \quad (1)$$

The goal of DDPMs is to train a diffusion model to revert the forward process. Specifically, a function approximator $\boldsymbol{\epsilon}_\phi$ is trained to predict the noise $\boldsymbol{\epsilon}$ contained in a noisy image x_t at step t. $\boldsymbol{\epsilon}_\phi$ is typically represented as a convolutional neural network characterised by its parameters ϕ. Most successful models [7, 15, 41] train their models using a simplified variant of the variational lower bound on the data distribution:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{t,x,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\phi\left(x_t, t\right)\right\|^2\right] \quad (2)$$

with t uniformly sampled from $\{1, \ldots, T\}$. The resulting update step for obtaining a sample for x_{t-1} from x_t is then

$$x_{t-1} = x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\phi(x_t, t) + \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \mathcal{N}(0, \mathbf{I}) \quad (3)$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, $\alpha_t = 1 - \beta_t$.

Text-to-image diffusion models build upon the above theory to introduce conditional diffusion processes using classifier-free guidance [14]. Given a condition y, usually represented as a text prompt, a diffusion model $\boldsymbol{\epsilon}_\phi(x_t, t, y)$ is trained to predict noise in an image as shown in Eq. 2. During training, conditioning y is randomly dropped out, leaving the diffusion model to predict noise without it. At inference, noise prediction is instead represented by:

$$\hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y) = \boldsymbol{\epsilon}_\phi(x_t, t, \emptyset) + s\Big(\boldsymbol{\epsilon}_\phi(x_t, t, y) - \boldsymbol{\epsilon}_\phi(x_t, t, \emptyset)\Big) \quad (4)$$

Where s is a user-defined constant controlling the degree of guidance and $\boldsymbol{\epsilon}(x_t, t, \emptyset)$ represents the noise prediction without conditioning.

4. Locally conditioned diffusion

We introduce **locally conditioned diffusion** as a method for providing better control over the composition of images generated by text-conditioned diffusion models. The key insight of our method is that we can selectively apply denoising steps conditioned on different text prompts to specific regions of an image.

Given a set of text prompts $\{y_1, \ldots, y_P\}$, classifier-free guidance [14] provides a method for predicting denoising

Figure 3. **2D locally conditioned diffusion results.** Given coarse segmentation masks as input, our method is able to generate images that follow the specified layout while ensuring seamless transitions. Results in the first row are generated using GLIDE [30], while the second and third rows show results generated using StableDiffusion [39].

steps conditioned on y_i:

$$\hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_i) = \boldsymbol{\epsilon}_\phi(x_t, t, \emptyset) + s\Big(\boldsymbol{\epsilon}_\phi(x_t, t, y_i) - \boldsymbol{\epsilon}_\phi(x_t, t, \emptyset)\Big) \tag{5}$$

Using a user-defined semantic segmentation mask m, where each pixel $m[j]$ has integer value $[1, P]$, the overall noise prediction can then be represented by selectively applying noise predictions to each labelled image patch:

$$\hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_{1:P}, m) = \sum_{i=1}^{P} \mathbb{1}_i(m) \odot \hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_i) \tag{6}$$

Where $\mathbb{1}_i(m)$ is the indicator image with equivalent dimensionality as m and

$$\mathbb{1}_i(m)[j] = \begin{cases} 1, & \text{if } m[j] = i \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

The proposed locally conditioned diffusion method is summarized in Algorithm 1.

Although a large proportion of noise predictions are not used, in practice only one diffusion model $\boldsymbol{\epsilon}_\phi$ is queried. All calls to the model for each unique text-conditioning y_i can be batched together for increased efficiency.

Our locally conditioned diffusion method generates high-fidelity 2D images that adhere to the given semantic segmentation masks. Note that, while each segment of the image is locally conditioned, there are no visible seams in the resulting image and transitions between differently labelled regions are smooth, as shown in Fig. 3 (see Sec. 6 for more details).

Algorithm 1 Locally conditioned diffusion

Require: Diffusion models $\hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_i)$, guidance scale s, semantic mask m
  $x_T \sim \mathcal{N}(0, I)$ ▷ Initialize Gaussian noise image
  for $t = T, \ldots, 1$ **do**
    $\epsilon_i \leftarrow \hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_i)$ ▷ Individual noise predictions
    $\epsilon \leftarrow \hat{\boldsymbol{\epsilon}}_\phi(x_t, t, \emptyset)$ ▷ Unconditional noise prediction
    $\epsilon_{\text{sem}} \leftarrow \sum_{i=1}^{P} \mathbb{1}_i(m) \odot s(\epsilon_i - \epsilon)$ ▷ Combine noise predictions
    $x_{t-1} = \text{Update}(x_t, \epsilon_{\text{sem}})$ ▷ Apply denoising step
  end for

5. Compositional 3D synthesis

To make compositional text-to-3D synthesis as simple as possible, our method takes 3D bounding boxes with corresponding text prompts as input. The goal of our method is to generate 3D scenes that contain objects specified by the text prompts while adhering to the specific composition provided by the input bounding boxes. In this section, we describe our method and how we apply locally conditioned diffusion in 2D to enable controllable generation in 3D.

Text-to-3D with Score Distillation Sampling. Our method builds off existing SDS-based text-to-3D methods [22, 35, 48]. SDS-based methods leverage a 3D scene representation parameterized by θ is differentiably rendered at a sampled camera pose, generating a noised image $g(\theta)$ which is passed into an image diffusion prior. Our method

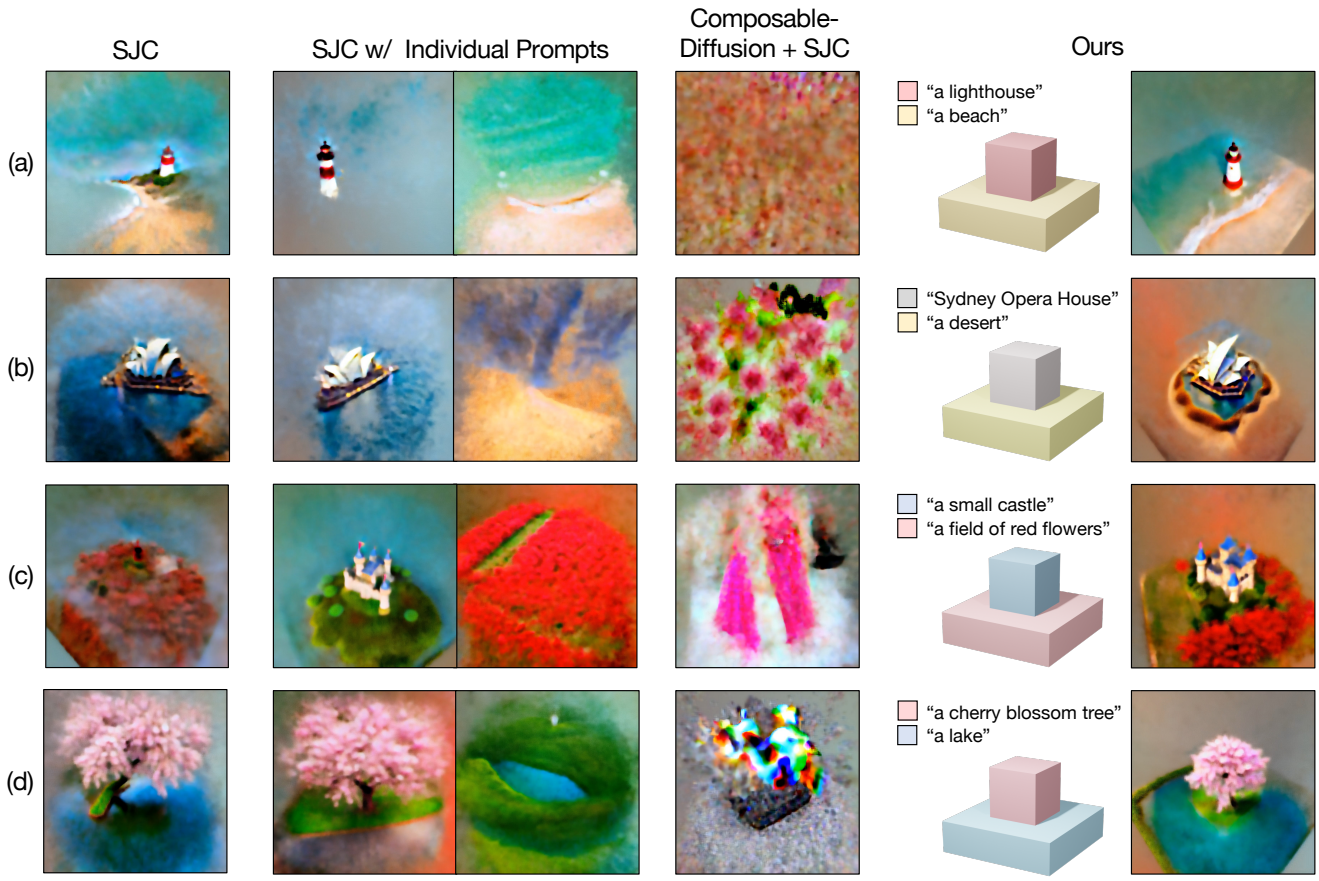

Figure 4. **Baseline comparisons.** Left to right: (i) SJC results using a single text prompt, (ii) SJC generating each scene component independently, (iii) SJC combined with Composable-Diffusion [25], and (iv) our method with corresponding bounding boxes and text prompts. Generations for each row use the text prompts listed on the right. Results in the first column are generated by combining individual text prompts with the connecting phrase “in the middle of”, e.g. “a lighthouse in the middle of a beach”. Our method successfully composes different objects into a coherent scene while following the user input bounding boxes.

builds off SJC [48], therefore we follow their pipeline, using a Voxel NeRF [6, 24, 45, 47, 52] representation and a volumetric renderer. The image diffusion prior provides the gradient direction to update scene parameters θ.

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, g(\theta)) = \mathbb{E}_{t,\epsilon}\left[w(t)(\hat{\epsilon}(x_t, y, t) - \epsilon)\frac{\partial x}{\partial \theta}\right] \quad (8)$$

This process is repeated for randomly sampled camera poses, as the text-conditioned image diffusion prior pushes each rendered image towards high density regions in the data distribution. Intuitively, SDS ensures images rendered from all camera poses resembles an image generated by the text-conditioned diffusion prior.

Bounding-box-guided text-to-3D synthesis. To achieve text-to-3D scene generations that adhere to user input bounding boxes, our method takes the standard SDS-based pipeline and conditions the image diffusion prior with renderings of the input bounding boxes. Specifically, our method works as follows. First, a random camera pose is sampled and a volume rendering of the 3D scene model is generated, we call this image x_t. Using the same camera pose, a rendering of the bounding boxes is also generated, we call this image m. This image is a segmentation mask, where each pixel contains an integer value corresponding to a user input text prompt. The volume rendering is then passed in to the image diffusion prior which provides the necessary gradients for optimizing the 3D scene representation. However, instead of conditioning the image diffusion prior on a single text prompt, we generate denoising steps for all text prompts with corresponding bounding boxes visible from the sampled camera pose. We then selectively apply these denoising steps to the image based on the segmentation mask m, and backpropagate the gradients to the 3D scene as usual. This is equivalent to applying the noise estimator described in Eq. 6 to the SDS gradient updates.

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, g(\theta), m) = \mathbb{E}_{t,\epsilon}\left[w(t)(\hat{\boldsymbol{\epsilon}}_\phi(x_t, t, y_{1:P}, m) - \epsilon)\frac{\partial x}{\partial \theta}\right] \quad (9)$$

While previous SDS-based text-to-3D methods ensure all rendered views of the 3D scene lie in the high probability density regions in the image prior conditioned on a single text prompt, our method ensures that all rendered views also align with the rendered bounding box segmentation masks. An overview of our method is provided in Fig. 2.

Object-centric camera pose sampling. As discussed in prior work [35, 22, 48], high classifier-free guidance weights are crucial for SDS methods to work. While image generation methods typically use guidance weights in the range of [2, 50], methods such as DreamFusion use guidance weights up to 100 [35]. Using a high guidance scale leads to mode-seeking properties which is desirable in the context of SDS-based generation. However, mode-seeking properties in image diffusion priors have the tendency of generating images with the object at the center of the image. When applying high guidance weights to locally conditioned diffusion, it is possible for the resulting image to ignore semantic regions that are off center, since mode-seeking behaviour of the diffusion model expects the object described by the text prompt to be at the center of the image, while the semantic mask only applies gradients from off-centered regions. In the context of our method, this mode-seeking behavior causes off-centered bounding box regions to become empty.

We combat this effect using *object-centric camera pose sampling*. While existing works [35, 22, 48] sample camera poses that are always pointed at the origin of the 3D scene model, in our method, we randomly sample camera poses that point at the center of each object bounding box instead. This means that during optimization of the 3D scene, each bounding box region will have the chance at appearing at the center of the image diffusion prior.

Locally conditioned diffusion with latent diffusion models. Existing SDS-based methods, such as DreamFusion [35] and Magic3D [22], leverage image diffusion priors in their method [1]. While SJC [48] uses a very similar methodology, their method actual employs a latent diffusion prior in the form of StableDiffusion [39]. Therefore, volume renderings of the 3D scene lies in the latent space instead of the image space. Note that previous work [34] has shown that the latent space is essentially a downsampled version of the image space, meaning we are still able to apply locally conditioned diffusion to the latent space.

[1] In Magic3D, a latent diffusion prior is also used, but the gradient of the encoder in the latent diffusion model is provided to convert gradient updates in the latent space back to the image space.

6. Experiments

We show qualitative results on compositional text-to-2D and text-to-3D generation. For 3D results, we mainly compare against SJC [48] as it is the best-performing publicly-available text-to-3D method. We also implemented a version of SJC that leverages Composable-Diffusion [25] as an additional baseline.

6.1. Compositional 2D results

Implementation details. We apply our locally conditioned diffusion method to existing text-conditioned diffusion models: GLIDE [30] and StableDiffusion [39]. We use pre-trained models provided by the authors of each respective paper to implement locally conditioned diffusion. Each image sample takes 10–15 seconds to generate on an NVIDIA A100 GPU, where duration varies according to the number of distinct semantic regions provided. Note that sampling time increases sub-linearly with respect to number of regions/prompts, this is because calls to the same model for each text-conditioning can be done in a single batch.

Qualitative results. We provide qualitative examples in Fig. 3. Our method is able to generate high-fidelity images that adhere to the input semantic masks and text prompts. Note that our method does not aim at generating images that follow the exact boundaries of the input semantic masks, instead it strives to achieve seamless transitions between different semantic regions. A key advantage of locally conditioned diffusion is that it is agnostic to the network architecture. We demonstrate this by showing that our method works on two popular text-to-image diffusion models GLIDE [30] and StableDiffusion [39].

6.2. Compositional 3D results

Implementation details. Our compositional text-to-3D method builds upon the SJC [48] codebase. Following SJC, we use a Voxel NeRF to represent the 3D scene model and StableDiffusion [39] as the diffusion prior for SDS-based generation. The Voxel NeRF representing the 3D scene is set to a resolution of 100^3. This configuration uses ≈ 10 GB of GPU memory. The original SJC method uses an emptiness loss scheduler to improve the quality of generated scenes. Our method also leverages this emptiness loss; please refer to the original SJC [48] for more details.

Qualitative results. We provide qualitative examples of compositional text-to-3D generations with bounding box guidance in Fig. 1. Notice that our method is able to generate coherent 3D scenes using simple bounding boxes with corresponding text prompts. Our method generates results that adhere to the input bounding boxes, allowing users to edit the size and position of individual scene components

Figure 5. **Size and position control.** Our method provides size and position control of individual scene components through user-defined bounding boxes. Our method provides fine-grained control over scene composition while ensuring each components blends seamlessly into the overall scene.

before generation. Fig. 5 shows generated results of the same scene prompts with differing bounding box sizes and positions. Note that our method is able to adapt to the user's input and generate scenes with varying compositions.

Baseline comparisons. We compare our method to different variants of SJC [48]. Namely, (i) SJC generations using a single prompt for the entire scene, (ii) individual SJC generations for each scene component, and (iii) an implementation of Composable-Diffusion [25] combined with SJC. Although DreamFusion [35] and Magic3D [22] have also shown to generate high-quality results, both models leverage image diffusion priors (Imagen [40] and eDiff-I [1]) that are not publicly available. However, it is important to note that our method can theoretically be applied to any SDS-based method. This can be achieved by replacing the image diffusion model in DreamFusion [35] and Magic3D [22] with the locally conditioned method described above.

We provide qualitative results for our method and each baseline in Fig. 4. In our experiments we attempt to compose two scene components into a coherent scene. Specifically, we choose an object-centric prompt that describes individual objects, paired with a scene-centric prompt that describes a background or an environment.

We observe that SJC fails to capture certain scene components when composing multiple scene components into a single prompt. Our method is able to capture individual scene components while blending them seamlessly into a coherent scene.

For object-centric prompts, SJC is able to create high-quality 3D generations. However, scene-centric prompts such as “a desert” or “a beach” end up generating dense volumes that resemble the text-prompt when rendered from different angles, but fail to reconstruct reasonable geometry.

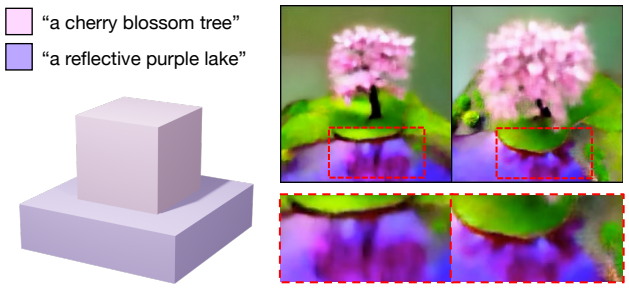

Figure 6. **Seamless transitions.** Our method is able to smoothly transition between scene components in different bounding boxes. In this example, we can see the reflection of the cherry blossom tree in the lake.

By defining bounding boxes for each scene component, our method provides coarse guidance for the geometry of the scene, this helps generate results with fewer “floater” artifacts. One option for compositional scene generation is to generate each scene component individually and then combine them manually afterwards. However, blending scene components together in a seamless manner takes considerable effort. Our method is able to blend individual objects with scene-level detail. As shown in Fig. 6, although the cherry blossom tree and the reflective purple lake correspond to different bounding box regions, our method is able to generate reflections of the tree in the water. Such effects would not be present if each scene component were generated individually and then manually combined.

We also compare our method to a composable implementation of SJC using Composable-Diffusion [25]. However, this method fails to generate reasonable 3D scenes.

Quantitative results. Following prior work [35, 16], we evaluate the CLIP R-Precision, the accuracy of retrieving the correct input caption from a set of distractors us-

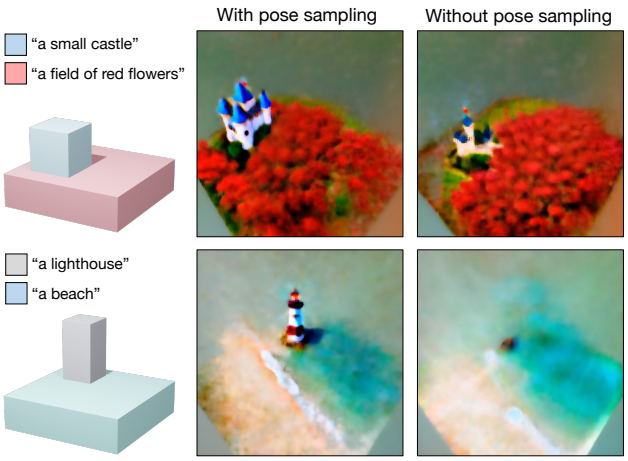

Figure 7. **Ablation over object-centric sampling.** Without object-centric sampling, our method fails fully capture off-centered scene components.

ing CLIP [36], of our compositional method. Tab. 1 reports CLIP R-Precision values for rendered views of scenes shown in Fig. 4 using our compositional method and SJC with a single prompt. Our method outperforms the baseline across all evaluation methods.

Table 1. CLIP R-Precision comparisons.

	R-Precision ↑		
Method	B/32	B/16	L/14
Single Prompt (SJC)	27..8	31.5	28.53
Composed (Ours)	**38.6**	**54.3**	**29.8**

Ablations. We found that object-centric camera pose sampling is essential for successful composition of multiple scene components. This is especially true for bounding boxes further away from the origin. We compare generations with and without object-centric pose sampling in Fig. 7. Note that our method tends to ignore certain scene components without object-centric sampling.

Speed evaluation. Unless stated otherwise, all results were generated by running our method for 10000 denoising iterations with a learning rate of 0.05 on a single NVIDIA RTX A6000. Note that scenes with a higher number of distinct text prompts require a longer period of time to generate. Using SJC, generating scene components individually causes generation time to scale linearly with number of prompts. In contrast, our method can compose the same number of prompts in a shorter amount of time, as calls to the same diffusion prior conditioned on different text-prompts can be batched together. Table 2 shows generation times for SJC and our method for 3000 denoising iterations.

Table 2. Generation times using SJC [48] for individual prompts and composing multiple prompts using our method.

	# of prompts		
Method	1	2	3
Individual (SJC)	8 mins	16 mins	24 mins
Composed (Ours)	8 mins	12 mins	15 mins

7. Discussion and Conclusions

Creating coherent 3D scenes is a challenging task that requires 3D design expertise and plenty of manual labor. Our method introduces a basic interface for creating 3D scenes without any knowledge of 3D design. Simply define bounding boxes for the desired scene components and fill in text prompts for what to generate in those regions.

Limitations and future work. Although text-to-3D methods using SDS [35, 48, 22] have shown promising results, speed is still a limiting factor. While advances in image-diffusion-model sampling [26, 18, 42, 49, 46] have enabled the generation of high-quality results in dozens of denoising steps, SDS method still require thousands of iterations before a 3D scene can be learned. SDS-based methods are also limited by their reliance on unusually high guidance scales [35]. A high guidance scale promotes mode-seeking, but leads to low diversity in the generated results. Concurrent works [2, 17] have shown other methods for controlling text-to-image diffusion synthesis with coarse segmentation masks. However, these methods require running a diffusion prior on multiple image patches before forming a single image, greatly increasing time needed to generate a single denoising step. In theory, these works could be applied in combination with our method, albeit with greatly increased time needed to generate a single 3D scene.

Ethical considerations. Generative models, such as ours, can potentially be used for spreading disinformation. Such misuses pose a societal threat and the authors of this paper do not condone such behavior. Since our method leverages StableDiffusion [39] as an image prior, it may also inherit any biases and limitations found in the 2D diffusion model.

Conclusion. Text-to-3D synthesis has recently seen promising advances. However, these methods mostly specialize in object-centric generations. Our method is an exciting step forward for 3D scene generation. Designing a 3D scene with multiple components no longer requires 3D modeling expertise. Instead, by defining a few bounding boxes and text prompts, our method can generate coherent 3D scenes that fit the input specifications.

References

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *ArXiv*, abs/2211.01324, 2022.

[2] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *ArXiv*, abs/2302.08113, 2023.

[3] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2016.

[4] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16123–16133, June 2022.

[5] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[6] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, 2022.

[7] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, abs/2105.05233, 2021.

[8] Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation with energy based models. In *Neural Information Processing Systems*, 2020.

[9] Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *ArXiv*, abs/1903.08689, 2019.

[10] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *2017 International Conference on 3D Vision (3DV)*, pages 402–411, 2017.

[11] Ruiqi Gao, Yang Song, Ben Poole, Ying Nian Wu, and Diederik P. Kingma. Learning energy-based models by diffusion recovery likelihood. *ArXiv*, abs/2012.08125, 2020.

[12] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Kristjanson Duvenaud, and Richard S. Zemel. Learning the stein discrepancy for training and evaluating energy-based models without sampling. In *International Conference on Machine Learning*, 2020.

[13] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. StyleNeRF: A style-based 3D-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021.

[14] Jonathan Ho. Classifier-free diffusion guidance. *ArXiv*, abs/2207.12598, 2022.

[15] Jonathan Ho, Ajay Jain, and P. Abbeel. Denoising diffusion probabilistic models. *ArXiv*, abs/2006.11239, 2020.

[16] Ajay Jain, Ben Mildenhall, Jonathan T. Barron, P. Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 857–866, 2021.

[17] Álvaro Barbero Jiménez. Mixture of diffusers for scene composition and high resolution image generation. *ArXiv*, abs/2302.02412, 2023.

[18] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. *ArXiv*, abs/2106.00132, 2021.

[19] Adam R Kosiorek, Heiko Strathmann, Daniel Zoran, Pol Moreno, Rosalia Schneider, Sona Mokra, and Danilo Jimenez Rezende. Nerf-vae: A geometry aware 3d scene generative model. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5742–5752. PMLR, 18–24 Jul 2021.

[20] Yann LeCun, Sumit Chopra, Raia Hadsell, Aurelio Ranzato, and Fu Jie Huang. A tutorial on energy-based learning. 2006.

[21] Yiyi Liao, Katja Schwarz, Lars Mescheder, and Andreas Geiger. Towards unsupervised learning of generative models for 3d controllable image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[22] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *ArXiv*, abs/2211.10440, 2022.

[23] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[24] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *ArXiv*, abs/2007.11571, 2020.

[25] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. *ArXiv*, abs/2206.01714, 2022.

[26] Zhaoyang Lyu, Xu Xudong, Ceyuan Yang, Dahua Lin, and Bo Dai. Accelerating diffusion models via early stop of the diffusion process. *ArXiv*, abs/2205.12524, 2022.

[27] Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. Gnerf: Gan-based neural radiance field without posed camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6351–6361, October 2021.

[28] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. *ArXiv*, abs/2211.07600, 2022.

[29] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[30] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2021.

[31] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11453–11464, June 2021.

[32] Michael Niemeyer and Andreas Geiger. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[33] Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. Stylesdf: High-resolution 3d-consistent image and geometry generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13503–13513, June 2022.

[34] Dong Huk Park, Grace Luo, C. M. Toste, Samaneh Azadi, Xihui Liu, Maka Karalashvili, Anna Rohrbach, and Trevor Darrell. Shape-guided diffusion with inside-outside attention. *ArXiv*, abs/2212.00210, 2022.

[35] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *ArXiv*, abs/2209.14988, 2022.

[36] 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 *International Conference on Machine Learning*, 2021.

[37] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv*, abs/2204.06125, 2022.

[38] Daniel Rebain, Mark Matthews, Kwang Moo Yi, Dmitry Lagun, and Andrea Tagliasacchi. Lolnerf: Learn from one look. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1558–1567, June 2022.

[39] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10674–10685, 2021.

[40] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *ArXiv*, abs/2205.11487, 2022.

[41] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2021.

[42] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *ArXiv*, abs/2202.00512, 2022.

[43] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv*, abs/2210.08402, 2022.

[44] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: Generative radiance fields for 3D-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[45] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer. Deepvoxels: Learning persistent 3d feature embeddings. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2441, 2018.

[46] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ArXiv*, abs/2010.02502, 2020.

[47] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5449–5459, 2021.

[48] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. *ArXiv*, abs/2212.00774, 2022.

[49] Daniel Watson, Jonathan Ho, Mohammad Norouzi, and William Chan. Learning to efficiently sample from diffusion probabilistic models. *ArXiv*, abs/2106.03802, 2021.

[50] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[51] Yinghao Xu, Menglei Chai, Zifan Shi, Sida Peng, Ivan Skorokhodov, Aliaksandr Siarohin, Ceyuan Yang, Yujun Shen, Hsin-Ying Lee, Bolei Zhou, and S. Tulyakov. Discoscene: Spatially disentangled generative radiance fields for controllable 3d-aware scene synthesis. *ArXiv*, abs/2212.11984, 2022.

[52] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5491–5500, 2021.

[53] Xin-Yang Zheng, Yang Liu, Peng-Shuai Wang, and Xin Tong. Sdf-stylegan: Implicit sdf-based stylegan for 3d shape generation. *CoRR*, abs/2206.12055, 2022.

[54] Peng Zhou, Lingxi Xie, Bingbing Ni, and Qi Tian. CIPS-3D: A 3D-Aware Generator of GANs Based on Conditionally-Independent Pixel Synthesis. *arXiv preprint arXiv:2110.09788*, 2021.