Generative Powers of Ten

Xiaojuan Wang[1] Janne Kontkanen[2] Brian Curless[1, 2] Steven M. Seitz[1, 2] Ira Kemelmacher-Shlizerman [1, 2]
Ben Mildenhall[2] Pratul Srinivasan[2] Dor Verbin[2] Aleksander Holynski[2, 3]
[1]University of Washington [2]Google Research [3]UC Berkeley
powers-of-ten.github.io

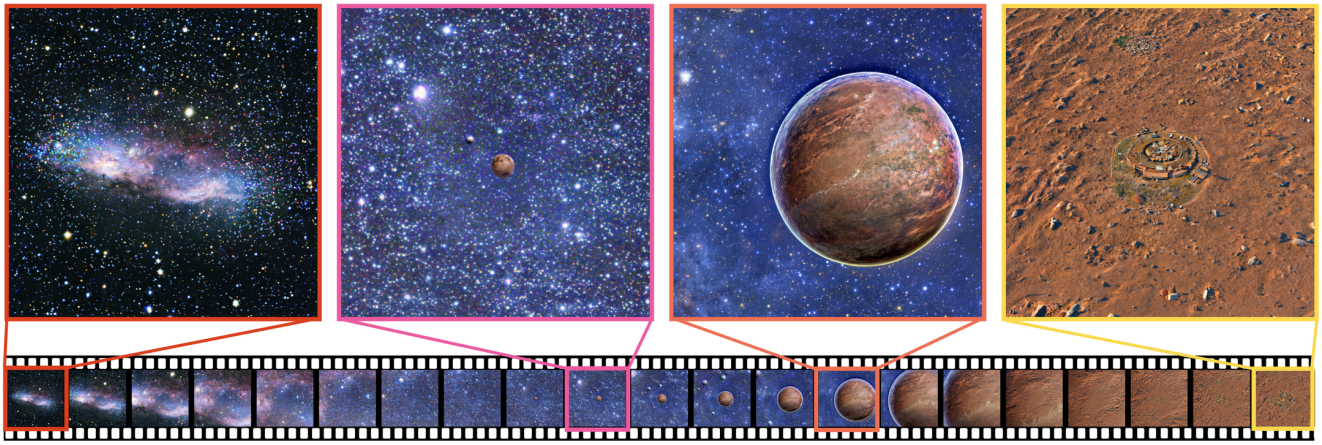

Figure 1. Given a series of prompts describing a scene at varying zoom levels, *e.g.*, from a distant galaxy to the surface of an alien planet, our method uses a pre-trained text-to-image diffusion model to generate a continuously zooming video sequence.

Abstract

We present a method that uses a text-to-image model to generate consistent content across multiple image scales, enabling extreme semantic zooms *into a scene,* e.g. *ranging from a wide-angle landscape view of a forest to a macro shot of an insect sitting on one of the tree branches. We achieve this through a joint multi-scale diffusion sampling approach that encourages consistency across different scales while preserving the integrity of each individual sampling process. Since each generated scale is guided by a different text prompt, our method enables deeper levels of zoom than traditional super-resolution methods that may struggle to create new contextual structure at vastly different scales. We compare our method qualitatively with alternative techniques in image super-resolution and outpainting, and show that our method is most effective at generating consistent multi-scale content.*

1. Introduction

Recent advances in text-to-image models [3, 6, 7, 15, 18, 19, 29] have been transformative in enabling applications like image generation from a single text prompt. But while digital images exist at a fixed resolution, the real world can be experienced at many different levels of scale. Few things exemplify this better than the classic 1977 short film *"Powers of Ten"*, shown in Figure 2, which showcases the sheer magnitudes of scale that exist in the universe by visualizing a continuous zoom from the outermost depths of the galaxy to the cells inside our bodies[1]. Unfortunately, producing animations or interactive experiences like these has traditionally required trained artists and many hours of tedious labor—and although we might want to replace this process with a generative model, existing methods have not yet demonstrated the ability to generate consistent content across multiple zoom levels.

Unlike traditional super-resolution methods, which generate higher-resolution content conditioned on the pixels of the original image, extreme zooms expose entirely new structures, *e.g.*, magnifying a hand to reveal its underlying skin cells. Generating such a zoom requires *semantic* knowledge of human anatomy. In this paper, we focus on solving this *semantic zoom* problem, *i.e.*, enabling text-

[1] https://www.youtube.com/watch?v=0fKBhvDjuy0

Figure 2. **Powers of Ten (*1977*)** This documentary film illustrates the relative scale of the universe as a single shot that gradually zooms out from a human to the universe, and then back again to the microscopic molecular level.

conditioned multi-scale image generation, to create *Powers of Ten*-like zoom videos. As input, our method expects a series of text prompts that describe different scales of the scene, and produces as output a multi-scale image representation that can be explored interactively or rendered to a seamless zooming video. These text prompts can be user-defined (allowing for creative control over the content at different zoom levels) or crafted with the help of a large language model (*e.g.*, by querying the model with an image caption and a prompt like *"describe what might you see if you zoomed in by 2x"*).

At its core, our method relies on a joint sampling algorithm that uses a set of parallel diffusion sampling processes distributed across zoom levels. These sampling processes are coordinated to be consistent through an iterative frequency-band consolidation process, in which intermediate image predictions are consistently combined across scales. Unlike existing approaches that accomplish similar goals by repeatedly increasing the effective image resolution (*e.g.*, through super-resolution or image outpainting), our sampling process jointly optimizes for the content of all scales at once, allowing for both (1) plausible images at each scale and (2) consistent content across scales. Furthermore, existing methods are limited in their ability to explore wide ranges of scale, since they rely primarily on the input image content to determine the added details at subsequent zoom levels. In many cases, image patches contain insufficient contextual information to inform detail at deeper (*e.g.*, 10x or 100x) zoom levels. On the other hand, our method grounds each scale in a text prompt, allowing for new structures and content to be conceived across extreme zoom levels. In our experiments, we compare our work qualitatively to these existing methods, and demonstrate that the zoom videos that our method produces are notably more consistent. Finally, we showcase a number of ways in which our algorithm can be used, *e.g.*, by conditioning purely on text or grounding the generation in a known (real) image.

2. Prior Work

Super-resolution and inpainting. Existing text-to-image based super resolution models [1, 22] and outpainting models [1, 16, 20, 27] can be adapted to the zoom task as autoregressive processes, *i.e.*, by progressively outpainting a zoomed-in image, or progressively super-resolving a zoomed-out image. One significant drawback of these approaches is that later-generated images have no influence on the previously generated ones, which can often lead to suboptimal results, as certain structures may be entirely incompatible with subsequent levels of detail, causing error accumulation across recurrent network applications.

Perpetual view generation. Starting from a single view RGB image, perpetual view generation methods like Infinite Nature [11] and InfiniteNature-Zero [12] learn to generate unbounded flythrough videos of natural scenes. These methods differ from our generative zoom in two key ways: (1) they translate the camera in 3D, causing a "fly-through" effect with perspective effects, rather than the "zoom in" our method produces, and (2) they synthesize the fly-through starting from a single image by progressivly inpainting unknown parts of novel views, wheras we generate the entire zoom sequence simultaneously and coherently across scales, with text-guided semantic control.

Diffusion joint sampling for consistent generation. Recent research [2, 10, 28, 30] leverages pretrained diffusion models to generate arbitrary-sized images or panoramas from smaller pieces using joint diffusion processes. These processes involve concurrently generating these multiple images by merging their intermediate results within the sampling process. In particular, *DiffCollage* [30] introduces a factor graph formulation to express spatial constraints among these images, representing each image as a node, and overlapping areas with additional nodes. Each sampling step involves aggregating individual predictions based on the factor graph. For this to be possible, a given diffusion model needs to be finetuned for different factor nodes. Other works such as *MultiDiffusion* [2] reconciles different denoising steps by solving for a least squares optimal solution: *i.e.*, averaging the diffusion model predictions at overlapping areas. However, none of these approaches can be applied to our problem, where our jointly sampled images have spatial correspondence at vastly different spatial scales.

3. Preliminaries

Diffusion models [5, 8, 23–26] generate images from random noise through a sequential sampling process. This sampling process reverses a destructive process that gradually adds Gaussian noise on a clean image $\mathbf{x}$. The intermediate noisy image at time step t is expressed as:

$$\mathbf{z}_t = \alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}_t,$$

where $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a standard Gaussian noise, and α_t and σ_t define a fixed noise schedule, with larger t corresponding to more noise. A diffusion model is a neural network ϵ_θ that predicts the approximate clean image $\hat{\mathbf{x}}$ directly, or equivalently the added noise ϵ_t in $\mathbf{z}_t$. The network is trained with the loss:

$$\mathcal{L}(\theta) = \mathbb{E}_{t\sim U[1,T],\epsilon_t\sim\mathcal{N}(\mathbf{0},\mathbf{I})}[w(t)\|\epsilon_\theta(\mathbf{z}_t; t, y) - \epsilon_t\|_2^2],$$

where y is an additional conditioning signal like text [16, 17, 21], and $w(t)$ is a weighting function typically set to 1 [8]. A standard choice for ϵ_θ is a U-Net with self-attention and cross-attention operations attending to the conditioning y.

Once the diffusion model is trained, various sampling methods [8, 13, 24] are designed to sample efficiently from the model, starting from pure noise $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and iteratively denoising it to a clean image. These sampling methods often rely on classifier-free guidance [8], a process which uses a linear combination of the text-conditional and unconditional predictions to achieve better adherence to the conditioning signal:

$$\hat{\epsilon}_t = (1 + \omega)\epsilon_\theta(\mathbf{z}_t; t, y) - \omega\epsilon_\theta(\mathbf{z}_t; t).$$

This revised $\hat{\epsilon}_t$ is used as the noise prediction to update the noisy image $\mathbf{z}_t$. Given a noisy image and a noise prediction, the estimated clean image $\hat{\mathbf{x}}_t$ is computed as $\hat{\mathbf{x}}_t = (\mathbf{z}_t - \sigma_t\hat{\epsilon}_t)/\alpha_t$. The iterative update function in the sampling process depends on the sampler used; in this paper we use DDPM [8].

4. Method

Let $y_0, ..., y_{N-1}$ be a series of prompts describing a single scene at varying, corresponding zoom levels $p_0, ..., p_{N-1}$ forming a geometric progression, *i.e.*, $p_i = p^i$ (we typically set p to 2 or 4). Our objective is to generate a sequence of corresponding $H \times W \times C$ images $\mathbf{x}_0, ..., \mathbf{x}_{N-1}$ from an existing, pre-trained, text-to-image diffusion model. We aim to generate the entire set of images jointly in a zoom-consistent way. This means that the image $\mathbf{x}_i$ at any specific zoom level p_i, should be consistent with the center $H/p \times W/p$ crop of the zoomed-out image $\mathbf{x}_{i-1}$.

We propose a *multi-scale joint sampling* approach and a corresponding *zoom stack* representation that gets updated in the diffusion-based sampling process. In Sec. 4.1, we introduce our zoom stack representation and the process that allows us to render it into an image at any given zoom level. In Sec. 4.2, we present an approach for consolidating multiple diffusion estimates into this representation in a consistent way. Finally, in Sec. 4.3, we show how these components are used in the complete sampling process.

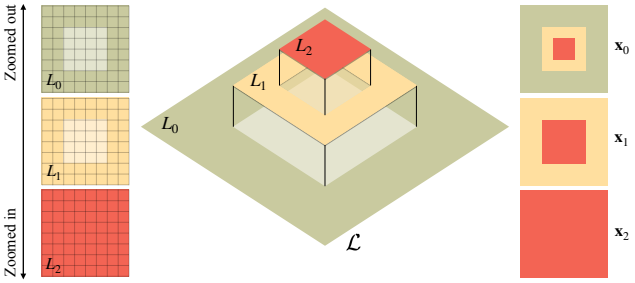

Figure 3. **Zoom stack.** Our representation consists of N layer images L_i of constant resolution (left). These layers are arranged in a pyramid-like structure, with layers representing finer details corresponding to a smaller spatial extent (middle). These layers are composited to form an image at any zoom level (right).

4.1. Zoom Stack Representation

Our zoom stack representation, which we denote by $\mathcal{L} = (L_0, ..., L_{N-1})$, is designed to allow rendering images at any zoom level $p_0, ..., p_{N-1}$. The representation, illustrated in Fig. 3, contains N images of shape $H \times W$, one for each zoom level, where the ith image L_i stores the pixels corresponding to the ith zoom level p_i.

Image rendering. The rendering operator, which we denote by $\Pi_{\text{image}}(\mathcal{L}; i)$, takes a zoom stack $\mathcal{L}$ and returns the image at the ith zoom level $p_i = p^i$. We denote by $\mathcal{D}_i(\mathbf{x})$ the operator which downscales the image $\mathbf{x}$ by factor p_i, and zero-pads the image back to size $H \times W$; and we denote by M_i the corresponding $H \times W$ binary image which has value 1 at the center $H/p_i \times W/p_i$ patch and value 0 at padded pixels. The operator $\mathcal{D}_i$ operates by prefiltering the image with a truncated Gaussian kernel of size $p_i \times p_i$ and resampling with a stride of p_i. As described in Alg. 1, an image $\mathbf{x}_i$ at the ith zoom level is rendered by starting with L_i, and iteratively replacing its central $H/p_j \times W/p_j$ crop with $\mathcal{D}_{j-i}(L_j)$, for $j = i + 1, ..., N - 1$. (In Alg. 1 we denote by $\odot$ the elementwise multiplication of a binary mask M with an image.) This process guarantees that rendering at different zoom levels will be consistent at overlapping central regions.

Noise rendering. At every denoising iteration of DDPM [8], each pixel is corrupted by globally-scaled i.i.d. Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Since we would like images rendered at different zoom levels to be consistent, it is essential to make sure the added noise is also consistent, with overlapping region across different zoom levels sharing the same noise structure. Therefore, we use a rendering operator similar to Π_{image} which converts a set of independent noise images, $\mathcal{E} = (E_0, ..., E_{N-1})$ into a single zoom-consistent noise $\epsilon_i = \Pi_{\text{noise}}(\mathcal{E}; i)$. However, because downsampling involves prefiltering, which modifies the statistics of the resulting noise, we upscale the jth downscaled noise component by p_j/p_i to preserve the variance,

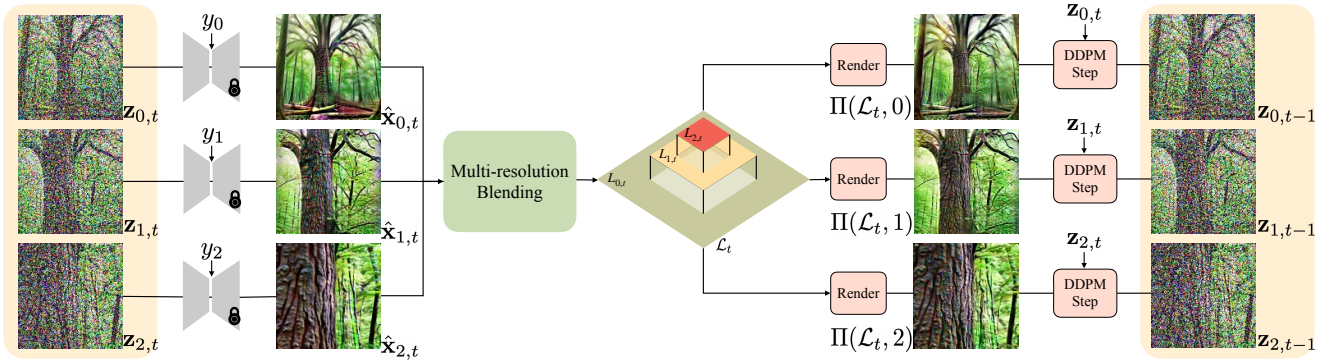

Figure 4. **Overview of a single sampling step.** (1) Noisy images $\mathbf{z}_{i,t}$ from each zoom level, along with the respective prompts y_i are simultaneously fed into the same pretrained diffusion model, returning estimates of the corresponding clean images $\hat{\mathbf{x}}_{i,t}$. These images may have inconsistent estimates for the overlapping regions that they all observe. We employ *multi-resolution blending* to fuse these regions into a consistent zoom stack $\mathcal{L}_t$ and re-render the different zoom levels from the consistent representation. These re-rendered images $\Pi_{\text{image}}(\mathcal{L}_t; i)$ are then used as the clean image estimates in the DDPM sampling step.

ensuring that the noise satisfies the standard Gaussian distribution assumption, *i.e.*, that $\boldsymbol{\epsilon}_i = \Pi_{\text{noise}}(\mathcal{E}; i) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for all levels i.

Algorithm 1 Image and noise rendering at scale i.

Set $\mathbf{x} \leftarrow L_i, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $j = i+1, \ldots, N-1$ **do**
  $\mathbf{x} \leftarrow M_{j-i} \odot \mathcal{D}_{j-i}(L_j) + (1 - M_{j-i}) \odot \mathbf{x}$
  $\boldsymbol{\epsilon} \leftarrow (p_j/p_i) M_{j-i} \odot \mathcal{D}_{j-i}(E_j) + (1 - M_{j-i}) \odot \boldsymbol{\epsilon}$
end for
return $\mathbf{x}, \boldsymbol{\epsilon}$

4.2. Multi-resolution blending

Equipped with a method for rendering a zoom stack and sampling noise at any given zoom level, we now describe a mechanism for integrating multiple observations of the same scene $\mathbf{x}_0, \ldots, \mathbf{x}_{N-1}$ at varying zoom levels $p_0, \ldots, p_{N-1}$ into a consistent zoom stack $\mathcal{L}$. This process is a necessary component of the consistent sampling process, as the diffusion model applied at various zoom levels will produce inconsistent content in the overlapping regions. Specifically, the jth zoom stack level L_j is used in rendering multiple images at all zoom levels $i \leq j$, and therefore its value should be consistent with multiple image observations (or diffusion model samples), namely $\{\mathbf{x}_i : i \leq j\}$. The simplest possible solution to this is to naïvely average the overlapping regions across all observations. This approach, however, results in blurry zoom stack images, since coarser-scale observations of overlapping regions contain fewer pixels, and therefore only lower-frequency information.

To solve this, we propose an approach we call *multi-resolution blending*, which uses Laplacian pyramids to selectively fuse the appropriate frequency bands of each observation level, which prevents aliasing as well as over-blurring. We show an outline of this process in Fig. 5. More concretely, to update the ith layer in the zoom stack, we begin by cropping all samples $j \geq i$ to match with the content of the ith level, and rescaling them back to $H \times W$. We then analyze each of these $N-i-1$ images into a Laplacian pyramid [4], and average across corresponding frequency bands

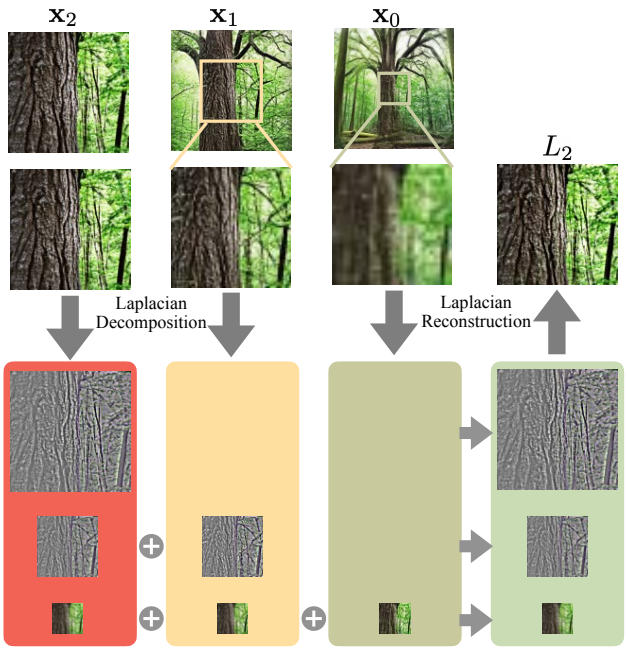

Figure 5. **Multi-resolution blending.** We produce a consistent estimate for Layer L_i in the zoom stack by merging the $H/p_j \times W/p_j$ central region of the corresponding zoomed out images $\mathbf{x}_j$ for $j \leq i$. This merging process involves (1) creating a Laplacian pyramid from each observation, and blending together the corresponding frequency bands to create a blended pyramid. This blended pyramid is recomposed into an image, which is used to update the layer L_i.

(see Figure 5), resulting in an average Laplacian pyramid, which can be recomposed into an image and assigned to the ith level of the zoom stack. This process is applied for each layer of the zoom stack L_i, collecting from all further zoomed-out levels $j \geq i$.

4.3. Multi-scale consistent sampling

Our complete *multi-scale joint sampling* process is shown in Alg. 2. Fig. 4 illustrates a single sampling step t: Noisy images $\mathbf{z}_{i,t}$ in each zoom level along with the respective prompt y_i are fed into the pretrained diffusion model in parallel to predict the noise $\hat{\boldsymbol{\epsilon}}_{i,t-1}$, and thus to compute the estimated clean images $\hat{\mathbf{x}}_{i,t}$. Equipped with our *multi-resolution blending* technique, the clean images are consolidated into a *zoom stack*, which is then rendered at all zoom levels, yielding consistent images $\Pi_{\text{image}}(\mathcal{L}_t; i)$. These images are then used in a DDPM update step along with the input $\mathbf{z}_t$ to compute the next $\mathbf{z}_{t-1}$.

Algorithm 2 Multi-scale joint sampling.

Set $\mathcal{L}_T \leftarrow \mathbf{0}$, $\mathbf{z}_{i,T} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \forall i = 0, ..., N-1$
for $t = T, \ldots, 1$ **do**
  $\mathcal{E} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  parfor $i = 0, \ldots, N-1$ **do**
    $\mathbf{x}_{i,t} = \Pi_{\text{image}}(\mathcal{L}_t; i)$
    $\boldsymbol{\epsilon}_i = \Pi_{\text{noise}}(\mathcal{E}; i)$
    $\mathbf{z}_{i,t-1} = \text{DDPM\_update}(\mathbf{z}_{i,t}, \mathbf{x}_{i,t}, \boldsymbol{\epsilon}_i)$
    $\hat{\boldsymbol{\epsilon}}_{i,t-1} = (1+\omega)\boldsymbol{\epsilon}_\theta(\mathbf{z}_{i,t-1}; t-1, y_i)$
      $-\omega\boldsymbol{\epsilon}_\theta(\mathbf{z}_{i,t-1}; t-1)$
    $\hat{\mathbf{x}}_{i,t-1} = (\mathbf{z}_{i,t-1} - \sigma_{t-1}\hat{\boldsymbol{\epsilon}}_{i,t-1})/\alpha_{t-1}$
  end parfor
  $\mathcal{L}_{t-1} \leftarrow \text{Blending}(\{\hat{\mathbf{x}}_{i,t-1}\}_{i=0}^{N-1})$
end for
return $\mathcal{L}_0$

4.4. Photograph-based Zoom

In addition to using text prompts to generate the entire zoom stack from scratch, our approach can also generate a sequence zooming into an existing photograph. Given the most zoomed-out input image $\boldsymbol{\xi}$, we still use Alg. 2, but we additionally update the denoised images to minimize the following loss function before every blending operation:

$$\ell(\hat{\mathbf{x}}_{0,t}, ..., \hat{\mathbf{x}}_{N-1,t}) = \sum_{i=0}^{N-1} \|\mathcal{D}_i(\hat{\mathbf{x}}_{i,t}) - M_i \odot \boldsymbol{\xi})\|_2^2, \quad (1)$$

where, as we defined in Sec. 4.1, $\mathcal{D}_i(\mathbf{x})$ downscales the image $\mathbf{x}$ by a factor p_i and pads the result back to $H \times W$, and M_i is a binary mask with 1 at the center $H/p_i \times W/p_i$ square and 0 otherwise. Before every blending operation we apply 5 Adam [9] steps at a learning rate of 0.1. This simple optimization-based strategy encourages the estimated

Figure 6. Selected images of our generated zoom sequences beginning with a provided real image. Left: Zoom from a man on a picnic blanket into the skin cells on his hand. Right: Zoom from a girl holding a leaf into the intricate vein patterns on the leaf. Face is blurred for anonymity.

Figure 7. Selected stills from our generated zoom videos (columns). Please refer to the supplementary materials for complete text prompts.

clean images $\{\hat{\mathbf{x}}_{i,t-1}\}_{i=0}^{N-1}$ to match with the content provided in $\boldsymbol{\xi}$ in a zoom-consistent way. We show our generated photograph-based zoom sequences in Fig. 6.

4.5. Implementation Details

For the underlying text-to-image diffusion model, we use a version of Imagen [21] trained on internal data sources, which is a cascaded diffusion model consisting of (1) a base model conditioned on a text prompt embedding and (2) a super resolution model additionally conditioned the low resolution output from the base model. We use its default DDPM sampling procedure with 256 sampling steps, and we employ our *multi-scale joint sampling* to the base model only. We use the super resolution model to upsample each generated image independently.

5. Experiments

In Figs. 6, 7, 8, 9, and 10, we demonstrate that our approach successfully generates consistent high quality zoom sequences for arbitrary relative zoom factors and a diverse set of scenes. Please see our supplementary materials for a full collection of videos. Sec. 5.1 describes how we generate text prompts, Sec. 5.2 demonstrates how our method outperforms diffusion-based outpainting and super-resolution models, and Sec. 5.3 justifies our design decisions with an ablation study.

5.1. Text Prompt Generation

We generate a collection of text prompts that describe scenes at varying levels of scales using a combination of ChatGPT [14] and manual editing. We start with prompting ChatGPT with a description of a scene, and asking it to formulate the sequence of prompts we might need for different zoom levels. While the results from this query are often plausible, they often (1) do not accurately match the corresponding requested scales, or (2) do not match the distribution of text prompts that the text-to-image model is able to most effectively generate. As such, we manually refine the prompts. A comprehensive collection of the prompts used to generate results in the paper are provided in the supplemental materials, along with the initial versions automatically produced by ChatGPT. In the future, we expect LLMs (and in particular, multimodal models) to automatically produce a sequence of prompts well suited for this application. In total, we collect a total of 10 examples, with the prompts sequence length varying form 6 to 16.

5.2. Baseline Comparisons

Fig. 8 compares zoom sequences generated with our method and without (*i.e.*, independently sampling each scale). When compared to our results, the independently-generated images similarly follow the text prompt, but clearly do not correspond to a single consistent underlying scene.

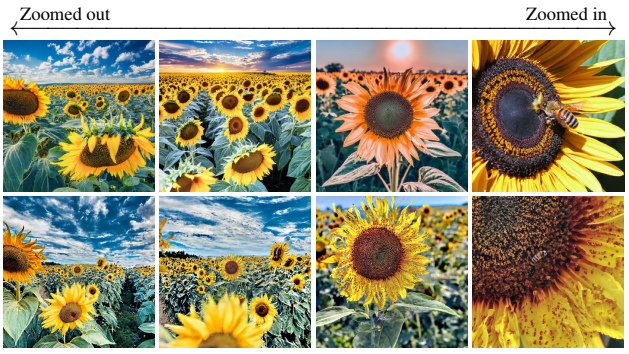

Figure 8. Generated zoom sequences with independent sampling (top) and our multi-scale sampling (bottom). Our method encourages different levels to depict a consistent underlying scene, while not compromising the image quality.

Next, we compare our method to two autogressive generation approaches for generating zoom sequences: (1) Stable Diffusion's [1] outpainting model and (2) Stable Diffusion's "upscale" super-resolution model. We show representative qualitative results in Fig. 9.

Comparison to progressive outpainting. The outpainting baseline starts with generating the most zoomed-in image and progressively generates coarser scales by downsampling the previous generated image and outpainting the surrounding area. As in our method, the inpainting of each level is conditioned on the corresponding text prompt. In Fig. 9, we show that because of the causality of the autoregressive process, the outpainting approach suffers from gradually accumulating errors, *i.e.*, when a mistake is made at a given step, later outpainting iterations may struggle to produce a consistent image.

Comparison to progressive super-resolution. The super-resolution baseline starts with the most zoomed-out image and generates subsequent scales by super-resolving the upscaled central image region, conditioned on the corresponding text prompt. The low resolution input provides strong structural information which constrains the layout of the next zoomed-in image. As we can see in Fig. 9, this super-resolution baseline is not able to synthesize new objects that would only appear in the finer, zoomed-in scales.

5.3. Ablations

In Fig. 10, we show comparisons to simpler versions of our method to examine the effect of our design decisions.

Joint vs. Iterative update. Instead of performing multi-scale blending approach, we can instead iteratively cycle through the images in the zoom stack, and perform one sampling step at each level independently. Unlike fully independent sampling, this process does allow for sharing of information between scales, since the steps are still applied to renders from the zoom stack. We find that although this pro-

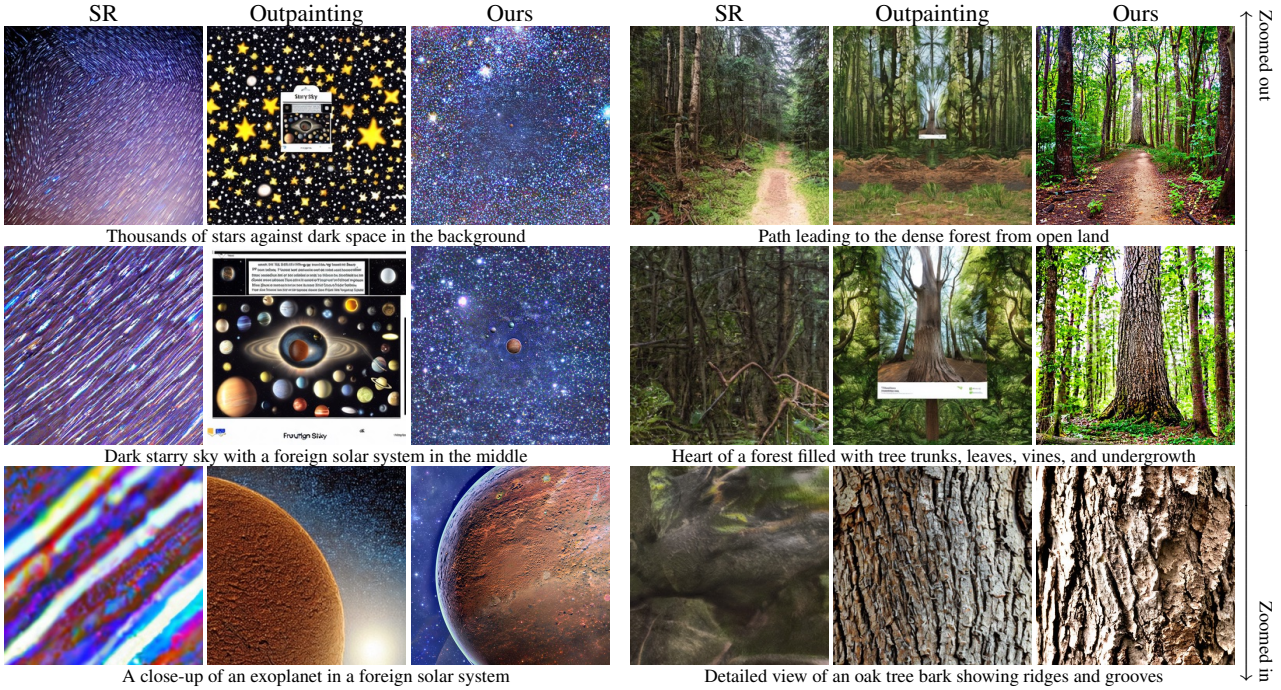

Figure 9. Comparisons with Stable Diffusion Outpainting and super-resolution (SR) models.

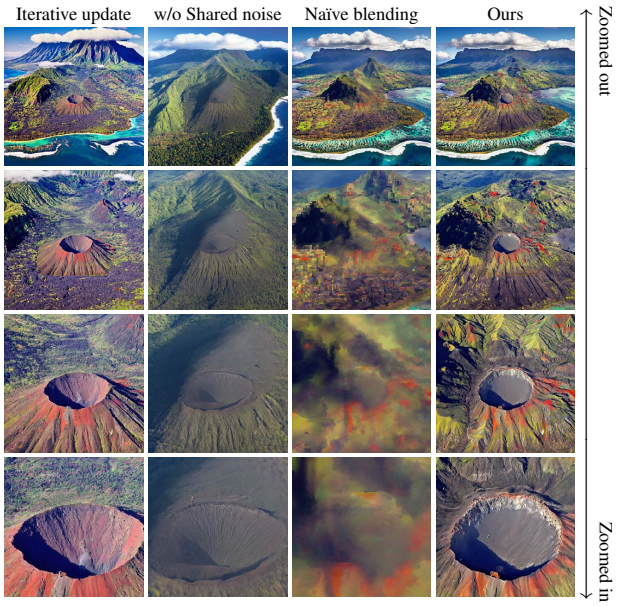

Figure 10. **Ablations.** We evaluate other options for multi-scale consistency: (1) iteratively updating each level separately, (2) naïve multi-scale blending, (3) removing the shared noise.

duces more consistent results than independent sampling, there remain inconsistencies at stack layer boundaries.

Shared vs. random noise Instead of using a shared noise Π_{noise}, noise can be sampled independently for each zoom level. We find that this leads to blur in the output samples.

Comparison with naïve blending. Instead of our multi-scale blending, we can instead naïvely blend the observations together, *e.g.*, as in MultiDiffusion [2]. We find that this leads to blurry outputs at deeper zoom levels.

6. Discussion & Limitations

A significant challenge in our work is discovering the appropriate set of text prompts that (1) agree with each other across a set of fixed scales, and (2) can be effectively generated consistently by a given text-to-image model. One possible avenue of improvement could be to, along with sampling, optimize for suitable geometric transformations between successive zoom levels. These transformations could include translation, rotation, and even scale, to find better alignment between the zoom levels and the prompts.

Alternatively, one can optimize the text embeddings, to find better descriptions that correspond to subsequent zoom levels. Or, instead, use the LLM for in-the-loop generation, *i.e.*, by giving LLM the generated image content, and asking it to refine its prompts to produce images which are closer in correspondence given the set of pre-defined scales.

Acknowledgements. We thank Ben Poole, Jon Barron, Luyang Zhu, Ruiqi Gao, Tong He, Grace Luo, Angjoo Kanazawa, Vickie Ye, Songwei Ge, Keunhong Park, and David Salesin for helpful discussions and feedback. This work was supported in part by UW Reality Lab, Meta, Google, OPPO, and Amazon.

References

[1] Stability AI. Stable-diffusion-2-inpainting. https://huggingface.co/stabilityai/stable-diffusion-2-inpainting. 2, 7

[2] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 2, 8

[3] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1

[4] Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In *Readings in computer vision*, pages 671–679. Elsevier, 1987. 4

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[6] Dave Epstein, Allan Jabri, Ben Poole, Alexei A Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *arXiv preprint arXiv:2306.00986*, 2023. 1

[7] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 1

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[10] Yuseung Lee, Kunho Kim, Hyunjin Kim, and Minhyuk Sung. Syncdiffusion: Coherent montage via synchronized joint diffusions. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2

[11] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *European Conference on Computer Vision*, pages 515–534. Springer, 2022. 2

[12] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021. 2

[13] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 3

[14] OpenAI. Chatgpt [large language model]. *https://chat.openai.com/chat*. 7

[15] Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit H Bermano, Eric Ryan Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. State of the art on diffusion models for visual computing. *arXiv preprint arXiv:2310.07204*, 2023. 1

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

[17] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3

[18] 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. 1

[19] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. *arXiv preprint arXiv:2307.06949*, 2023. 1

[20] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 2

[21] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 3, 7

[22] 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*, 45(4):4713–4726, 2022. 2

[23] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 2

[24] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3

[25] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[26] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[27] Luming Tang, Nataniel Ruiz, Chu Qinghao, Yuanzhen Li, Aleksander Holynski, David E Jacobs, Bharath Hariharan, Yael Pritch, Neal Wadhwa, Kfir Aberman, and Michael Rubinstein. Realfill: Reference-driven generation for authentic image completion. *arXiv preprint arXiv:2309.16668*, 2023. 2

[28] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *arXiv preprint arXiv:2307.01097*, 2023. 2

[29] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 1

[30] Qinsheng Zhang, Jiaming Song, Xun Huang, Yongxin Chen, and Ming-Yu Liu. Diffcollage: Parallel generation of large content with diffusion models. *arXiv preprint arXiv:2303.17076*, 2023. 2