Terrain Diffusion Network: Climatic-Aware Terrain Generation with Geological Sketch Guidance

Zexin Hu[1], Kun Hu[1,*], Clinton Mo[1], Lei Pan[2], Zhiyong Wang[1]
[1]The University of Sydney
[2]Civil Aviation Flight University of China
zehu4485@sydney.edu.au, kun.hu@sydney.edu.au, clmo6615@uni.sydney.edu.au,
leipan.cafuc@hotmail.com, zhiyong.wang@sydney.edu.au

Abstract

Sketch-based terrain generation seeks to create realistic landscapes for virtual environments in various applications such as computer games, animation and virtual reality. Recently, deep learning based terrain generation has emerged, notably the ones based on generative adversarial networks (GAN). However, these methods often struggle to fulfill the requirements of flexible user control and maintain generative diversity for realistic terrain. Therefore, we propose a novel diffusion-based method, namely terrain diffusion network (TDN), which actively incorporates user guidance for enhanced controllability, taking into account terrain features like rivers, ridges, basins, and peaks. Instead of adhering to a conventional monolithic denoising process, which often compromises the fidelity of terrain details or the alignment with user control, a multi-level denoising scheme is proposed to generate more realistic terrains by taking into account fine-grained details, particularly those related to climatic patterns influenced by erosion and tectonic activities. Specifically, three terrain synthesisers are designed for structural, intermediate, and fine-grained level denoising purposes, which allow each synthesiser concentrate on a distinct terrain aspect. Moreover, to maximise the efficiency of our TDN, we further introduce terrain and sketch latent spaces for the synthesizers with pre-trained terrain autoencoders. Comprehensive experiments on a new dataset constructed from NASA Topology Images clearly demonstrate the effectiveness of our proposed method, achieving the state-of-the-art performance. Our code is available at https://github.com/TDNResearch/TDN.

Introduction

In the real world, terrains are subject to a variety of climatic conditions, such as temperature variations, erosion from water or wind, and the presence of vegetation. When employing an automated terrain generation method, it is essential to accurately depict such weather events and natural phenomena, while closely adhering to a user's structural guidance for controllability to avoid unintended outcomes. Yet, conventional controllable example-based and sketch-based methods (Talgorn and Belhadj 2018; Guérin et al. 2017) often compromise the generation of realistic terrains in terms of geologically precise fine-grained characteristics. For optimal fidelity, it requires more advanced controllable terrain synthesis techniques that are capable of producing diverse and intricate features with fine granularity.

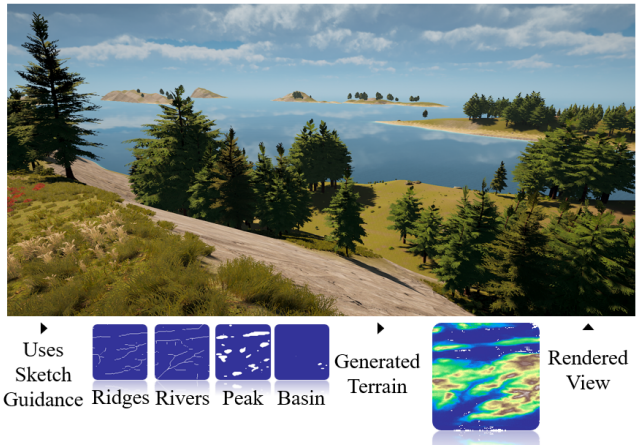

Figure 1: Sample terrain generated by our Terrain Diffusion Network (TDN), which is controlled by user-provided sketch guidance with ridges, rivers, basins and peaks.

Due to the success of deep learning techniques for computer vision (Wu et al. 2016; Jin et al. 2017; Siarohin et al. 2018; Zheng and Liu 2020), the potential of deep-learning methods has been explored for high fidelity terrain generation. The majority of these studies are based on generative adversarial networks (GAN) (Beckham and Pal 2017; spick and walker 2019). However, GANs suffer from the trade off between fidelity and divergence to user-provided conditions.

Recently, diffusion methods (Dhariwal and Nichol 2021) have shown promising results in generating a wide variety of images based on user prompts. This suggests potential to overcome the challenges found in terrain generation: they foster diversity at a fine-grained level and offer the opportunity to enable the generation of climatic morphogenesis. Nonetheless, the stochastic nature of diffusion models presents a significant challenge for users' capability to guide the terrain generation process. To the best of our knowledge, there is no existing study with the controllable diffusion architecture for fine-grained and diversified terrain generation.

*Corresponding author.

Therefore, we propose a novel deep architecture based on diffusion, namely terrain diffusion network (TDN). It employs a generation process by denoising Gaussian noise-perturbed terrain latent representations to produce realistic and visually appealing terrains that closely align with the user provided sketch guidance. TDN allows users to explicitly define the structural-level terrain characteristics through their input sketches for rivers, ridges, basins, and peaks, as shown in Fig. 1. To adequately consider the user guidance, a multi-level denoising scheme with a terrain sketch encoder, that aims to structural consistency, is devised to replace the monolithic denoising in the conventional diffusion process. It employs a coarse-to-fine strategy, utilising multiple terrain synthesizers with direct user guidance to articulate structural, intermediate, and fine-grained geological patterns. This facilitates the transformation of low-resolution inputs into high-resolution terrain outputs. Specifically, different denoisor weights are learnt for distinct synthesis stages, and they can focus on different levels of terrain patterns. In detail, the structural synthesiser focuses more on the overall terrain components, such as rivers, ridges, basins, and peaks, whilst the fine-grained synthesiser provides further terrain details such as climatic patterns including geomorphological erosion and tectonic events. To facilitate the efficiency of our TDN, we further introduce terrain and sketch latent spaces with lower dimensions by utilizing pre-trained autoencoders.

In summary, the key contributions of this study are:

- A novel deep architecture that takes multiple geometric factors into account for controllable terrain generation.
- A multi-level denoising synthesizer to formulate both structural and fine-grained terrain patterns aiming for producing realistic and climatic-aware terrain patterns.
- A new dataset is constructed from NASA Topology Images (Allen 2005) to evaluate the effectiveness of the proposed TDN. Comprehensive experiments demonstrate that the proposed TDN is able to achieve high-quality realistic terrain synthesis with flexible user controls.

Related Work

Procedural Terrain Generation

Procedural terrain generation was first introduced by Mandelbrot and Mandelbrot (1982). Generally, procedural methods involved manipulating fractal noise by using a predefined set of rules, algorithms, or functions of input parameters to mimic visually faithful terrain features. Over time the methods became increasingly computationally efficient due to various research efforts that had been made. While procedural models are usually computationally efficient, they usually lack the capability to involve user control, stemmed from the stochastic nature of the approach. To address this, constrained fractals has been introduced, which combined deterministic features such as user-prescribed terrain projections (Belhadj and Audibert 2005) or deterministic splines (Derzapf et al. 2011) with stochastic fractals. More recently, Génevaux et al. (2013); Gaillard et al. (2019) both proposed pipelines that allow users to have more flexibility for controllable generation by making minor adjustments to input parameters. However, while providing user control, terrains generated with procedural methods tend to appear pristine and lack signs of erosion and weathering, making them less realistic than real ones shaped by climatic morphogenesis.

Simulation-based Terrain Generation

Simulation-based methods addressed some of the trade-off challenges experienced by procedural methods. They typically emulated geomorphological processes, such as erosion and tectonics, to generate realistic terrain features. Simulating erosion with approximately tuned parameters helped create more realistic terrain height maps (Cordonnier et al. 2016). Cordonnier et al. (2017a) simulated tectonic features to generate ranges, valleys and other large-scale terrain features. Krištof et al. (2009) simulated hydraulic events to generate terrains. Recent research took advantage of the advancements in computational capacity to enhance user interactions with terrain generation (Benes and Forsbach 2001; Mei, Decaudin, and Hu 2007; Vanek et al. 2011).For instance, in Cordonnier et al. (2017b) a simulation system was proposed to simulate geomorphological events at an unprecedentedly high speed. Yet, simulation-based methods still suffered from the limited flexibility of user control and expensive computational costs for more extensive scenarios.

Example-based Terrain Generation

Procedural and simulation-based methods often required meticulous parameter tuning to generate desired terrains and lacked intuitive ways to involve user control. In contrast, example-based methods used more intuitive user guidance such as sketches and images for terrain generation (spick and walker 2019; Li et al. 2006). Zhou et al. (2007) proposed a more direct user control scheme that blends patches from real terrains containing height fields with user-defined sketches. Šťava et al. (2008) incorporated small-scale simulation-based models with an interactive editing scheme. Rusnell, Mould, and Eramian (2009) utilized surface deformation to match user constraints. Tasse et al. (2020); Scott and Dodgson (2021) generated new terrains based on existing samples. While example-based methods generally provided a high level of control, they often involve a trade-off between realism/geological correctness and controllability (Hnaidi et al. 2010; Gain, Marais, and Straßer 2009; Vanek et al. 2011). In other words, the increased control can typically results in relatively lower-quality terrains.
.

Deep-Learning based Terrain Generation

The deep learning approach had gained impressive performance and increasing popularity for the fields in computer vision and graphics. Deep generation methods such as Generative Adversarial Networks (GAN) (Radford, Metz, and Chintala 2015) and conditional GANs (cGAN) (Mirza and Osindero 2014), which accept multimodal guidance like images and texts to condition on the generation process, have been applied to sketch-based terrain generation and achieved state-of-the-art results (Guérin et al. 2017). However, due to

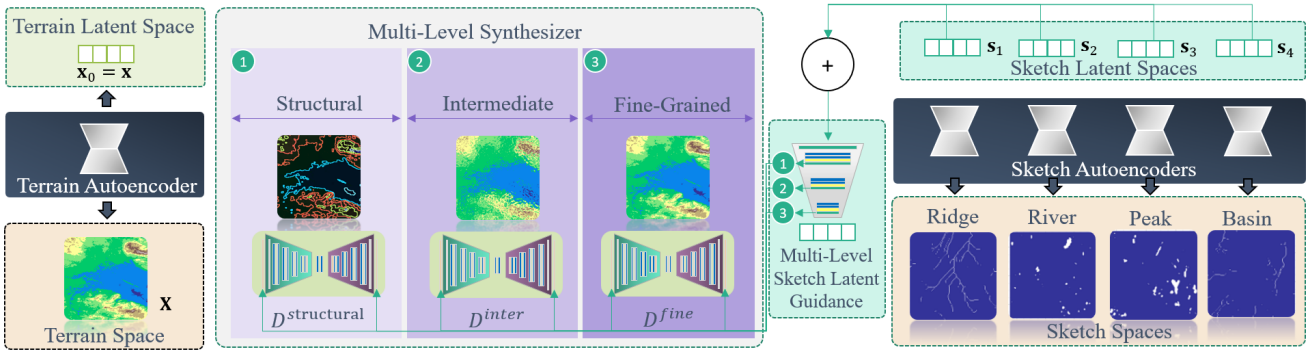

Figure 2: Illustration of our proposed Terrain Diffusion Network (TDN).

the inherently stochastic nature of GANs, the generative process still lacks user control for fine-grained details without using multiple datasets (Salimans et al. 2016). For example, GAN-based methods fall short in incorporating climate specific factors when solely using real-world terrain data, which requires the training of several distinct GANs, each focuses on one dataset for a specific climatic aspect.

Recently, a significant progress in the field of image-to-image generations has been achieved by leveraging diffusion models (Li et al. 2023; Huang et al. 2023; Wang et al. 2023; Mei and Patel 2023). These methods deliver an unparalleled level of fidelity and user control (Zhang and Agrawala 2023; Zhang et al. 2023). Although diffusion models demonstrated unprecedented diversity and fidelity in its generation, the inherent stochastic nature of diffusion process imposes significant challenges for implementing effective user control. In contrast, our TDN devises a terrain diffusion approach integrating multi-level user guidance to generate climate-aware and plausible terrain that align with user sketches. TDN empowers users with granular control while eliminating the necessity for training on multiple specialized datasets.

Methodology

As shown in Fig. 2, our proposed Terrain Diffusion Network (TDN) is composed of two primary components: terrain and sketch autoencoders and multi-level terrain synthesisers. The autoencoders with U-Net (Ronneberger, Fischer, and Brox 2015) like structures streamlines a terrain map by converting it into a lower-dimensional latent representation for reducing computational demands during the diffusion process. The diffusion process leverages multi-level synthesisers to create a latent terrain representation, with diversity and flexibility under the guidance of user-provided sketches.

Controllable Terrain Synthesis

Our method uses a set of user's sketches as the guidance, including rivers, ridges, basins, and peaks to generate a realistic terrain map $\mathbf{X} \in \mathbb{R}^{C\times W\times H}$. We denote the sketch guidance as a set $\mathbb{S} = \{\mathbf{S}_1, ..., \mathbf{S}_n, ..., \mathbf{S}_N\}$, with each user sketch $\mathbf{S}_n \in \mathbb{R}^{C\times W\times H}$. These sketches and the terrain map can be viewed as one channel ($C = 1$) images. Fig. 1 depicts an example of paired $\mathbf{X}$ and $\mathbb{S}$, where $N = 4$ in our study covers four major control factors in terrain synthesis.

Terrain and Sketch Autoencoder

A terrain autoencoder is introduced to compress the terrain data into a latent space. This enables the projection of terrain data into a more manageable, lower-dimensional space, thereby reducing the computational demands of the diffusion process (Rombach et al. 2022). The autoencoder is based on a U-Net architecture with an encoder Z and a decoder Z'. In the encoding stage, the target terrain is funneled through the down-sampling encoder components to generate the latent representation. Subsequently, in the decoding stage, the terrain is reconstructed via the up-sampling decoder components. The overall process can be formulated as $\mathbf{X} \approx Z^{'}(\mathbf{x} = Z(\mathbf{X}))$, where $\mathbf{x} \in \mathbb{R}^d$ is the latent representation of $\mathbf{X}$. A perceptual loss function is adopted to train the autoencoder (Hinton and Salakhutdinov 2006), which encourages the autoencoder to preserve high-level features integral to the terrain perception. The computation of perceptual loss involves passing both the ground truth terrain and the reconstructed terrain through a pre-trained loss network. Mathematically, we have (Pan et al. 2016):

$$\mathcal{L}(\mathbf{X}, \mathbf{x}_{rec}) = \frac{1}{\mathbf{C}_j\mathbf{H}_j\mathbf{W}_j} \left\| \phi_j(\hat{\mathbf{X}}) - \phi_j(\mathbf{X}) \right\|. \quad (1)$$

Similarly, we adopt autoencoders with similar structures to encode user-provided sketch guidance. Encoder networks Z_n, $n = 1, ..., N$ are introduced for rivers, ridges, basins, and peaks, respectively, to compress $\mathbf{S}_n$ into $\mathbf{s}_n$ the n-th sketch latent space. An additional transformer is employed to integrate the information from these encoded latent data and create a comprehensive yet condensed overview of the input sketch guidance $\mathbb{S}$, as illustrated in Fig. 1.

Subsequently, this aggregated sketch representation is processed through a convolutional layer and a sketch guidance vector can be obtained as $\mathbf{s} \in \mathbb{R}^d$. Note that $\mathbf{s}$ aligns in dimensionality with the terrain latent representation $\mathbf{x}$, ensuring a coherent and efficient guided diffusion process.

Terrain Diffusion

The diffusion model conducts a procedure that involves adding noise to a sample in the latent terrain space and then using a deep neural network to reverse the noise-perturbed sample back to its original latent representation. Specifically,

the diffusion process introduces a Gaussian noise to the latent representation iteratively. At the t-th step, the relationship between $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ can be formulated as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \mu_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \mathbf{\Sigma}_t = \beta_t\mathbf{I})), \tag{2}$$

where $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ is a conditional probability for a forward diffusion process, which follows a Gaussian distribution with mean μ_t and variance $\mathbf{\Sigma}_t$. In practice, starting from the original terrain data $\mathbf{x}_0 = \mathbf{x}$, the forward diffusion process is traceable as follows (Ho, Jain, and Abbeel 2020):

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}). \tag{3}$$

By reparametrising Eq. (2) with $\epsilon_0, ..., \epsilon_{t-2}, \epsilon_{t-1} \sim \mathcal{N}(0, I)$, and $\alpha_t = 1 - \beta_t, \bar{\alpha}_t = \prod_{s=0}^{t} \alpha_s$, we have:

$$\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}). \tag{4}$$

Now the reverse process can be conducted based on a distribution, of which the probability is defined as $p(\mathbf{x}_{t-1}|\mathbf{x}_t)$. It formulates a terrain latent representation in the previous sampling step $t-1$ by providing its current state at t. Specifically, this probability and thus the corresponding reverse process are estimated by a neural network $D_\theta(\mathbf{x}_t|\theta)$, where θ contains learnable weights. Mathematically, we have:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, \mathbf{t}), \Sigma_\theta(\mathbf{x}_t, \mathbf{t})), \tag{5}$$

where $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is an estimation of $p(\mathbf{x}_{t-1}|\mathbf{x}_t)$. Note $D_\theta(\mathbf{x}_t|\theta)$ is also known as a denoiser or synthesiser in diffusion. The trajectory of the reverse process would be:

$$p_\theta(\mathbf{x}_{0:T}) = p_\theta(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t). \tag{6}$$

According to Luo (2022), θ can be optimized with a loss:

$$L_t = \mathbb{E}_{\mathbf{x}_0, \mathbf{t}, \epsilon}[||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathbf{t})||^2], \tag{7}$$

where x_0 is the noise free terrain latent representation, ϵ is the noise sampled from $\mathcal{N}(0, I)$ and the noise scheduler $\bar{\alpha}_t = \prod_{s=0}^{t} \alpha_s$. Our primary goal is to utilize our neural network D_θ to estimate the noise ϵ as ϵ_θ.

Multi-Level Sketch Guidance Integration

The terrain generation is challenging to control via an unconditional diffusion process. Thus, we propose a diffusion process uses sketch guidance for enhanced controllability in terrain synthesis. Specifically, by giving the sketch prompt $\mathbb{S}$ and its latent representation $\mathbf{s}$ from users, a conditioned diffusion process can be formulated as (Song et al. 2020):

$$p_\theta(\mathbf{x}_{0:T}|\mathbf{s}) = p_\theta(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{s}). \tag{8}$$

It conditions on Eq. (6) with added user guidance $\mathbf{s}$. By formulating Eq. (7) with the condition of user sketch guidance $\mathbf{s}$, the conditioned synthesiser can be optimized regarding its weights θ by solving the following optimization problem:

$$L_t = \mathbb{E}_{\mathbf{x}_0|s, \mathbf{t}, \epsilon}[||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0|s + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathbf{t})||^2]. \tag{9}$$

In practice, how to provide appropriate guidance remains as an open question. While the conventional classifier-free (Ho and Salimans 2022) or the cross-attention mechanisms used in methods such as stable diffusion (SD) (Rombach et al. 2022) may work in some contexts, they are not the optimal solution for terrain generation as they were primarily designed for dimensionality matching (e.g., caused by cross-modal discrepancy) between their guidance and target data. Unlike these approaches, our terrain sketch encoder aims to preserve the structural consistency between the sketch guidance and the terrain latent representations. We introduce a mechanism to integrate sketch guidance directly into the denoising process, where the guidance $\mathbf{s}$ shares identical dimension d with the noise perturbed terrain representation $\mathbf{x}_t$. Specifically, $\mathbf{s}$ is concatenated with $\mathbf{x}_t$ natively through a convolution layer. It is worth noting $\mathbf{s}$ is constant throughout all diffusion steps t. To this end, the network is able to estimate $\hat{\mathbf{x}}_0|\mathbf{s}$ in an iterative manner, which is the generated output. We also enhance the control by employing skip connection, allowing the generated terrain be optimised towards user guidance in the cases where diffusion process introduce too much diversity. To summarize, this intrinsic integration enables the diffusion process to retain structural correspondence in line with the user provided sketches.

Multi-Level Synthesisers

Existing methods primarily conduct diffusion iteratively by applying a single denoiser neural network. The weights of the denoiser are fixed. Thus, the user's sketch guidance is applied with the same weight irrespective of the current noise level at a particular step t, forcing diffusion to balance between adhering to user's sketch and generating properable terrain. By dealing with both structural level terrain patterns and fine-grained level terrain climatic characteristics, a single synthesiser with the conventional approach would become a generalist that lacks specialization. This falls short of precisely generating fine-grained details or obtaining consistency in adhering to the user's structural sketch guidance. Therefore, in our approach, we propose a diffusion process within multiple latent spaces, where the diffusion process is guided by multiple synthesisers of stratified geological levels: structural-level, intermediate-level, and fine-grained level. Specifically, the fine-grained synthesiser plays a role in preserving and reconstructing climatic details, whilst at the structural-level synthesiser generates terrain patterns that closely match the user's input sketch guidance.

Unlike e-Diff (Balaji et al. 2022), our multi-synthesiser employs different latent spaces to focus on different granular levels of the terrain. Leeb et al. (2022) showed that varying latent space dimensions affects different aspects of terrain generation. We crafted our synthesizer with different latent spaces, each tailored to specific terrain generation levels. User sketches are encoded to the terrain's latent dimension for direct diffusion integration.

The structural-level synthesiser controls the generation of lowest resolution latent representation. At this stage, the primary structural information is expected to be reconstructed.

Specifically, based on Eq. (9), we optimize $\theta_{structural}$ with:

$$L_t^{\text{structural}} = \mathbb{E}_{\mathbf{x}_0|s,\mathbf{t},\epsilon}[||\epsilon - \epsilon_{\theta_s}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0|s + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathbf{t})||^2]. \tag{10}$$

Likewise, we further introduce an intermediate synthesiser $D_{inter}^{\text{inter}}(\mathbf{x}_t|\theta_{inter})$. It controls the generation of the coarse level details and plays a balancing role between reconstructing both structural and fine-grained level perturbed patterns, where θ_{inter} is optimized by:

$$L_t^{\text{inter}} = \mathbb{E}_{\mathbf{x}_0|s,\mathbf{t},\epsilon}[||\epsilon - \epsilon_{\theta_i}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0|s + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathbf{t})||^2]. \tag{11}$$

Lastly, the fine-grained synthesiser $D_{\text{fine-grained}}(\mathbf{x}_t|\theta_f)$ controls the fine grained generation by focusing on denoising terrain details such as climatic factors to synthesis results that closely match with realistic terrains. It is optimized by:

$$L_t^{\text{fine-grained}} = \mathbb{E}_{\mathbf{x}_0|s,\mathbf{t},\epsilon}[||\epsilon - \epsilon_{\theta_f}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0|s + \sqrt{1-\bar{\alpha}_t}\epsilon, \mathbf{t})||^2]. \tag{12}$$

Despite similar structures, the three synthesizers are trained separately. Their varied weights allow for tailored guidance at different stages of the diffusion process.

Experiments & Discussions

Dataset

The dataset used in this study is collected from NASA and the key features of terrains are extracted using Pysheds (Bartos 2020) packages. The images have a 1:6400 meter scale, with each extracted elevation map being 144×144. The sketches extracted from an image contain basins, peaks, rivers, and ridges. Sketches are produced using Pysheds' Digital Elevation Map (DEM) conditioning techniques, such as pit filling and flow direction determination. We obtained 10,446 samples for training and 2,611 for testing.

Implementation Details

TDN generates a latent representation of a terrain that is initially perturbed with noise through a specialized noise scheduler. To ensure a more seamless and gradual process of noise addition, we incorporate a cosine scheduler originally proposed by Nichol and Dhariwal (2021). The U-Net like synthesisers in TDN comprise 3 downsampling blocks, 3 upsampling blocks, and a middle block with 8-head self-attention mechanisms employed for embedding the diffusion step t. With a total of $\sim$ 1.216 billion parameters, the model's training is conducted with a learning rate of 1.0e-05 and a batch size of 6. During inference, TDN takes a set of user sketches as the input and iteratively generates a noise-free terrain latent representation. In total, 36 steps are taken to derive the final latent representation. We used one NVIDIA RTX 3090 GPU card to train our model. It took 82 hours in total, with 8 hours training Terrain VAE, 12 hours training sketch VAE and 62 hours training multi-diffusion. Given that the desired terrain dimension remains unchanged, there is no need to retrain the latent space for different inputs. When introducing new categories of sketch guidance or removing existing ones, only the sketch autoencoder would require retraining to ensure it correctly extracts the necessary features for projection into the latent space.

Methods	FID ↓	MSE ↓	SSIM ↑	CD ↓
GAN 2017	4.6599	0.0391	0.7288	0.5658
VQGAN 2021	6.5117	0.0548	0.5476	1.9774
SD 2022	8.2326	0.0854	0.5557	1.1171
ControlNet 2023	7.8923	0.0829	0.4508	1.4611
GliGen 2023	7.1527	0.0711	0.4394	1.6371
TDN (Ours)	**0.4402**	**0.0059**	**0.8289**	**0.3545**

Table 1: Comparisons between terrain generation methods.

Overall Performance

To demonstrate the effectiveness of TDN for terrain generation, we compare it with two state-of-the-art approaches: 1) GAN-based methods: GAN (Guérin et al. 2017) and VQ-GAN (Esser, Rombach, and Ommer 2021), and 2) diffusion-based methods: Stable Diffusion (Rombach et al. 2022) and ControlNet (Zhang and Agrawala 2023), Gligen (Li et al. 2023). We utilize the metrics: Frechet Inception Distance (FID) (Chong and Forsyth 2020) and Mean Squared Error (MSE) to measure the performance by assessing the similarity between the generated terrains and the corresponding ground truth. We further introduced additional structural metrics: Structural Similarity (SSIM) and Chamfer Distance (CD). SSIM considers the overall terrain distribution and CD calculates the distance of the nearest neighbours.

Our TDN exhibits superior performance and surpasses all existing techniques in quantitative evaluations, as shown in Table 1. The FID scores suggest TDN generates terrain with high fidelity. Note that general diffusion methods including Stable Diffusion (SD), ControlNet and Gligen are inferior to GAN-based methods, especially regarding the structural patterns as indicated FID. Diffusion model struggles to extract appropriate features without specific terrain domain knowledge. SD cannot accurately condition on multiple sketches, and causing the subsequent poor performance of ControlNet and Gligen as they relied on the pre-trained stable diffusion model. In contrast, TDN successfully addresses this issue in diffusion by incorporating terrain specific guidance. For GAN-based methods, they cannot provide adequate adherence of user sketech data whilst maintaining a high perceptual quality. Although VQGAN has gained superior performance for extensive scenarios, vanilla GAN outperforms VQGAN for terrain synthesis. GAN outperformed VQ-GAN due to different mechanisms used for integrating the user sketches: GAN adopts a straightforward way to involve guidance via convolution, whilst VQGAN utilizes cross-attentions. It suggests the uniqueness of terrain generation and necessities distinct mechanisms.

Fig. 3 visualises generated terrains, comparing TDN with GAN, SD, and the ground truth. GAN and SD were selected as they are top performers of their respective approaches. TDN exhibits the ability to preserve a compelling degree of terrain's natural fidelity while adhering to user input sketch. This ability allows the generation of a more climate-aware and plausible terrain compare to GAN-based methods. The visual results align with our quantitative assessments, the distribution of the terrain generated by our model exhibits a

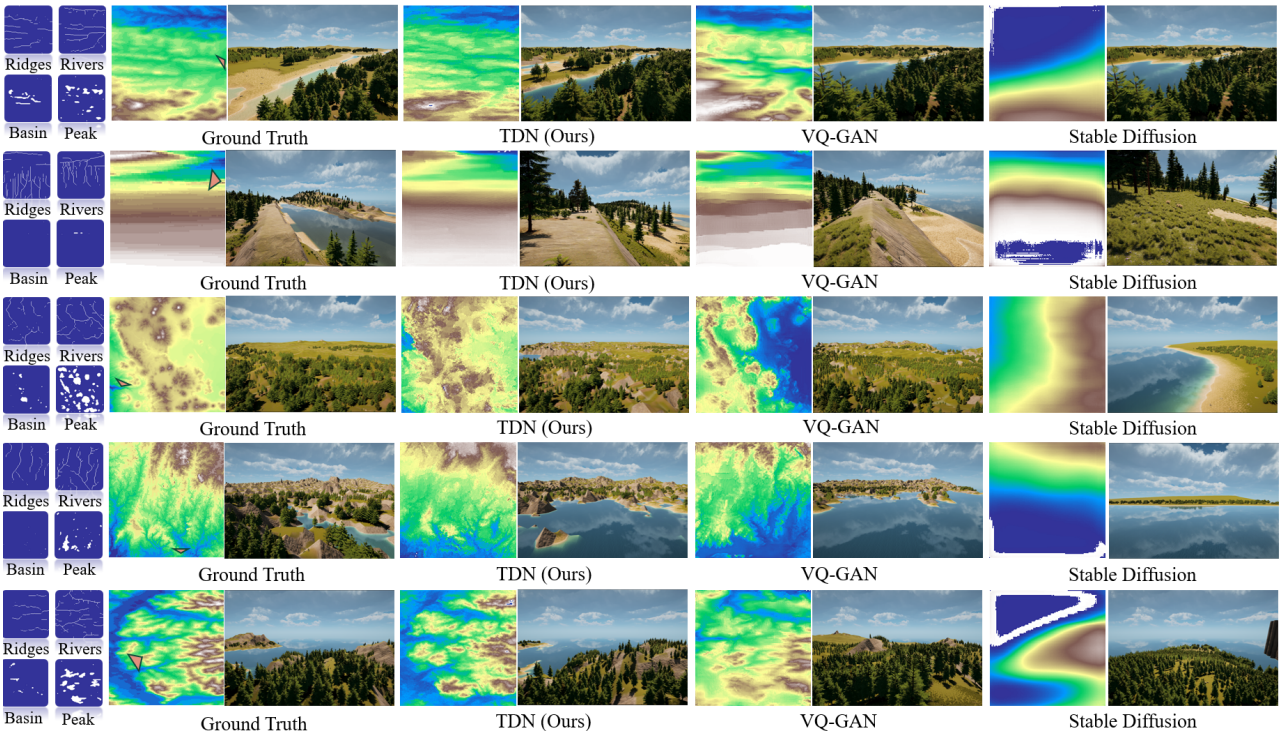

Figure 3: Comparisons between various terrain generation methods with user input sketches, height map and rendered view.

closer representation of the actual terrain data, as compared to those produced by GAN and SD. Specifically, SD cannot generate a closely matched terrain with the conventional cross-attention mechanism to integrate guidance, whilst our multiple synthesisers are capable of individually learning the denoising process at each step and significantly improves the control and performance that is specific to terrain generation.

Ablation Study

Ablation studies are conducted to further evaluate the effectiveness of individual mechanisms. Quantitative and qualitative results are shown in Table 2 and Fig. 4, respectively.

TDN w/o Multi-Level Synthesizers (MLS) follows a conventional diffusion design with a single denoiser. FID and MSE drop 13.18% and 16.55% compared with TDN's multi-level synthesizer, respectively. TDN's multi-level scheme introduces a degree of flexibility in the weighting of guidance, offering enhanced control. This dynamic guidance enables TDN to better adapt to various user demands.

TDN w/o Intermediate-Level Synthesizers (ILS) removes the intermediate-level synthesizer of TDN, and keeps the structural and fine-grained levels as a two-level denoising scheme. This setting significantly compromises the generation, compared with its three-level counterpart (TDN). It indicates the necessity to connect and transit from the structural stage to the fine-grained stage. The setting without the intermediate-level synthesizer suffers on the structual level generation as well as fine-grained generation. In Fig. 4, it loses some of its fine-grained details, and clearly deviates from the ground truth. Intriguingly, our experiments show that a model with a single synthesizer can sometimes surpass the performance of utilizing two synthesizers. This suggests that an efficient performance isn't merely a function of the number of synthesizers, but also lies in the nuances of how the individual synthesizer is employed. Such findings open avenues for further exploration into the specific roles and optimal usage of synthesizers in the terrain generation process.

TDN w/o Sketch Autoencoders (SAEs) removes the sketch autoencoders, and the sketches share the autoeconder with terrains. The sketch autoencoders facilitate the learning of specific weights for various sketch inputs. Fig. 4 shows the effectiveness of the sketch autoencoders in the generation process, which adheres more closely to the sketch.

TDN w/o Direct Sketch Guidance (DSG) uses cross-attention to integrate the guidance with the terrain. Our direct integration strategy is more effective for the consistency between the synthesized terrain and the input sketches, taking advantage of their directly matched latent patterns. Stable Diffusion and ControlNet have same issues as using cross-attention had an adverse effect for terrain generation.

Removing Guidance for Sketch Autoencoder

Our model can generate terrains without retraining the diffusion component as shown in Fig. 5 and Table 3. Omitting one guidance leads to noticeably different outputs, and results in substantially lower quantitative performance. Notably, the non-guided patterns are decided by the generation process itself. However, qualitative result remain satisfactory.

Methods	FID ↓	MSE ↓	SSIM ↑	CD ↓
w/o MLS	0.5071	0.0071	0.7703	0.4144
w/o ILS	1.8879	0.0177	0.8128	0.4037
w/o SAEs	1.9401	0.0312	0.5752	1.3725
w/o DSG	1.8957	0.0195	0.6975	0.9998
TDN (Ours)	**0.4402**	**0.0059**	**0.8289**	**0.3545**

Table 2: Ablation study on the proposed TDN.

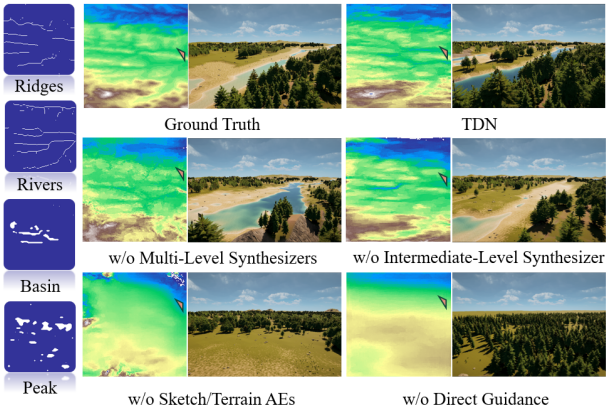

Figure 4: Qualitative study on various mechanisms.

Methods	FID ↓	MSE ↓	SSIM ↑	CD ↓
AE w/o ridges	0.7151	0.0498	0.1564	1.9048
AE w/o basins	0.7463	0.0466	0.2051	1.9260
TDN (Ours)	**0.4402**	**0.0059**	**0.8289**	**0.3545**

Table 3: Comparisons of different autoencoder settings.

Out-of-Domain Generation

We showcase the versatility of TDN by generating terrains with human-provided sketches. Fig. 6 shows that TDN is responsive and capable of adapting to different user sketches whilst generating plausiable terrains. TDN can handle conflicting and complex user sketches. In the first example, the ridges overlaps with basins, peaks, and rivers. TDN shows successful and realistic generation. Especially when compared with a GAN-based method, GAN puts most of its weights on one sketch - the peaks of the terrain, and fails to take other inputs into consideration. The same finding can be verified through the second example. TDN can integrate climatic conditioning, as in the third example. Given seemingly random input, TDN produces a more realistic terrain compare to GAN. By synthesising these climatic conditions, TDN delivers an output that is both a faithful representation of the input and an authentic digital portrayal of real-world terrains.

Limitations

Several limitations are with TDN. First, the quality of the terrain generation is overly reliant on the scale of the training data. The diffusion model learns the data's distribution and can only denoise samples within the given distribution.

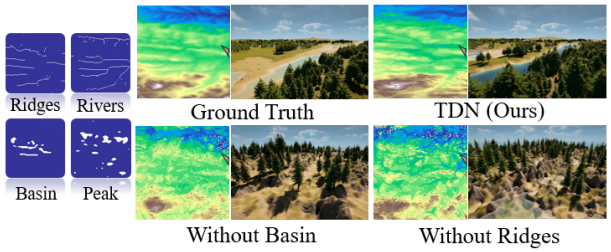

Figure 5: Qualitative results with different autoencoders.

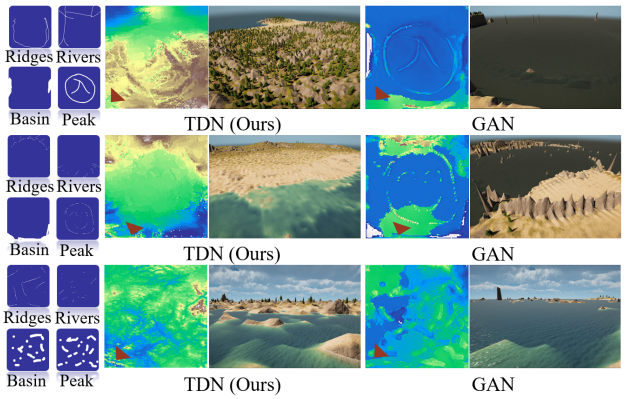

Figure 6: Out-of-domain terrain generation.

Consequently, attempts to encompass a wide range of terrain types come at the cost of sacrificing granular details. This is due to the challenging nature of obtaining high-resolution or ultra-high-resolution terrain data that contains a diverse array of terrain types. This might be rectified by having high quality data, for example, sourcing terrains from multiple sources, and hence building a large and comprehensive database. However, the sheer volume of the training data introduces a significant challenge, as the associated costs of training would be substantially high. Second, the training of the terrain and sketch autoencoders is done in an independent manner. When these components are combined with the diffusion model, the overall training process becomes computationally expensive. Moreover, the inference on a 144 × 144 terrain map takes around 11 seconds for 36 steps. It could potentially impede the model's accessibility and widespread adoption. Finally, the control is mainly sketch-based with a fixed number of user-provided sketches. Yet, it is more promising to integrate with flexible guidance such as: customized sketch categories rather than using all of them; and guidance in other modalities like text or video.

Conclusion

This study presents TDN - a diffusion-based method for terrain generation with user guidance. A multi-level denoising scheme with three synthesizers formulates structural and fine-grained level terrain characteristics. To maximise the efficiency of TDN, we introduce terrain and sketch latent spaces with pre-trained autoencoders to integrate user guidance. Our experiments demonstrate the superiority of TDN.

Acknowledgments

This study was partially supported by Australian Research Council (ARC) grant #DP210102674.

References

Allen, J. 2005. NASA Topography Image.

Balaji, Y.; Nah, S.; Huang, X.; Vahdat, A.; Song, J.; Kreis, K.; Aittala, M.; Aila, T.; Laine, S.; Catanzaro, B.; et al. 2022. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*.

Bartos, M. 2020. pysheds: simple and fast watershed delineation in python.

Beckham, C.; and Pal, C. 2017. A step towards procedural terrain generation with gans. *arXiv preprint arXiv:1707.03383*.

Belhadj, F.; and Audibert, P. 2005. Modeling landscapes with ridges and rivers: bottom up approach. In *International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, 447–450.

Benes, B.; and Forsbach, R. 2001. Layered data representation for visual simulation of terrain erosion. In *Spring Conference on Computer Graphics*, 80–86. IEEE.

Chong, M. J.; and Forsyth, D. 2020. Effectively unbiased fid and inception score and where to find them. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6070–6079.

Cordonnier, G.; Braun, J.; Cani, M.-P.; Benes, B.; Galin, E.; Peytavie, A.; and Guérin, E. 2016. Large scale terrain generation from tectonic uplift and fluvial erosion. In *Computer Graphics Forum*, volume 35, 165–175. Wiley Online Library.

Cordonnier, G.; Cani, M.-P.; Benes, B.; Braun, J.; and Galin, E. 2017a. Sculpting mountains: Interactive terrain modeling based on subsurface geology. *IEEE Transactions on Visualization and Computer Graphics*, 24(5): 1756–1769.

Cordonnier, G.; Galin, E.; Gain, J.; Benes, B.; Guérin, E.; Peytavie, A.; and Cani, M.-P. 2017b. Authoring landscapes by combining ecosystem and terrain erosion simulation. *ACM Transactions on Graphics*, 36(4): 1–12.

Derzapf, E.; Ganster, B.; Guthe, M.; and Klein, R. 2011. River networks for instant procedural planets. In *Computer Graphics Forum*, volume 30, 2031–2040. Wiley Online Library.

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

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12873–12883.

Gaillard, M.; Benes, B.; Guérin, E.; Galin, E.; Rohmer, D.; and Cani, M.-P. 2019. Dendry: A procedural model for dendritic patterns. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 1–9.

Gain, J.; Marais, P.; and Straßer, W. 2009. Terrain sketching. In *Symposium on Interactive 3D Graphics and Games*, 31–38.

Génevaux, J.-D.; Galin, E.; Guérin, E.; Peytavie, A.; and Benes, B. 2013. Terrain Generation Using Procedural Models Based on Hydrology. *ACM Transactions on Graphics*, 32(4).

Guérin, É.; Digne, J.; Galin, E.; Peytavie, A.; Wolf, C.; Benes, B.; and Martinez, B. 2017. Interactive example-based terrain authoring with conditional generative adversarial networks. *ACM Transactions on Graphics*, 36(6): 228–1.

Hinton, G. E.; and Salakhutdinov, R. R. 2006. Reducing the dimensionality of data with neural networks. *science*, 313(5786): 504–507.

Hnaidi, H.; Guérin, E.; Akkouche, S.; Peytavie, A.; and Galin, E. 2010. Feature based terrain generation using diffusion equation. In *Computer Graphics Forum*, volume 29, 2179–2186. Wiley Online Library.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33: 6840–6851.

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Huang, L.; Chen, D.; Liu, Y.; Shen, Y.; Zhao, D.; and Zhou, J. 2023. Composer: Creative and controllable image synthesis with composable conditions. *arXiv preprint arXiv:2302.09778*.

Jin, Y.; Zhang, J.; Li, M.; Tian, Y.; Zhu, H.; and Fang, Z. 2017. Towards the automatic anime characters creation with generative adversarial networks. *arXiv preprint arXiv:1708.05509*.

Krištof, P.; Beneš, B.; Křivánek, J.; and Šťava, O. 2009. Hydraulic erosion using smoothed particle hydrodynamics. In *Computer Graphics Forum*, volume 28, 219–228. Wiley Online Library.

Leeb, F.; Bauer, S.; Besserve, M.; and Schölkopf, B. 2022. Exploring the Latent Space of Autoencoders with Interventional Assays. *Advances in Neural Information Processing Systems*, 35: 21562–21574.

Li, Q.; Wang, G.; Zhou, F.; Tang, X.; and Yang, K. 2006. Example-based realistic terrain generation. In *Advances in Artificial Reality and Tele-Existence: 16th International Conference on Artificial Reality and Telexistence, ICAT 2006, Hangzhou, China, November 29-December 1, 2006. Proceedings*, 811–818. Springer.

Li, Y.; Liu, H.; Wu, Q.; Mu, F.; Yang, J.; Gao, J.; Li, C.; and Lee, Y. J. 2023. GLIGEN: Open-Set Grounded Text-to-Image Generation. *CVPR*.

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

Mandelbrot, B. B.; and Mandelbrot, B. B. 1982. *The fractal geometry of nature*, volume 1. WH freeman New York.

Mei, K.; and Patel, V. 2023. VIDM: Video Implicit Diffusion Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8): 9117–9125.

Mei, X.; Decaudin, P.; and Hu, B.-G. 2007. Fast hydraulic erosion simulation and visualization on GPU. In *Pacific Conference on Computer Graphics and Applications*, 47–56. IEEE.

Mirza, M.; and Osindero, S. 2014. Conditional Generative Adversarial Nets. arXiv:1411.1784.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 8162–8171. PMLR.

Pan, T.; Zhongliang, F.; Lili, W.; and Kai, Z. 2016. Perceptual loss with fully convolutional for image residual denoising. In *Chinese Conference on Pattern Recognition*, 122–132. Springer.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10684–10695.

Ronneberger, O.; Fischer, P.; and Brox, T. 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*, 234–241. Springer.

Rusnell, B.; Mould, D.; and Eramian, M. 2009. Feature-rich distance-based terrain synthesis. *The Visual Computer*, 25: 573–579.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X. 2016. Improved techniques for training gans. *Advances in Neural Information Processing Systems*, 29.

Scott, J. J.; and Dodgson, N. A. 2021. Example-based terrain synthesis with pit removal. *Computers & Graphics*, 99: 43–53.

Siarohin, A.; Sangineto, E.; Lathuiliere, S.; and Sebe, N. 2018. Deformable gans for pose-based human image generation. In *IEEE conference on computer vision and pattern recognition*, 3408–3416.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

spick, r. r.; and walker, j. 2019. Realistic and textured terrain generation using GANs. In *ACM SIGGRAPH European Conference on Visual Media Production*, 1–10.

Šťava, O.; Beneš, B.; Brisbin, M.; and Křivánek, J. 2008. Interactive terrain modeling using hydraulic erosion. In *ACM Siggraph/Eurographics Symposium on Computer Animation*, 201–210.

Talgorn, F.-X.; and Belhadj, F. 2018. Real-time sketch-based terrain generation. In *Proceedings of Computer Graphics International*, 13–18.

Tasse, F. P.; Emilien, A.; Cani, M.-P.; Hahmann, S.; and Bernhardt, A. 2020. First person sketch-based terrain editing. In *Graphics Interface*, 217–224. AK Peters/CRC Press.

Vanek, J.; Benes, B.; Herout, A.; and Stava, O. 2011. Large-scale physics-based terrain editing using adaptive tiles on the GPU. *IEEE Computer Graphics and Applications*, 31(6): 35–44.

Wang, Q.; Kong, D.; Lin, F.; and Qi, Y. 2023. DiffSketching: Sketch Control Image Synthesis with Diffusion Models. *arXiv preprint arXiv:2305.18812*.

Wu, J.; Zhang, C.; Xue, T.; Freeman, B.; and Tenenbaum, J. 2016. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in Neural Information Processing Systems*, 29.

Zhang, L.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

Zhang, Z.; Zhao, Z.; Yu, J.; and Tian, Q. 2023. ShiftDDPMs: Exploring Conditional Diffusion Models by Shifting Diffusion Trajectories. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(3): 3552–3560.

Zheng, Z.; and Liu, J. 2020. GAN: efficient style transfer using single style image. *arXiv preprint arXiv:2001.07466*.

Zhou, H.; Sun, J.; Turk, G.; and Rehg, J. M. 2007. Terrain synthesis from digital elevation models. *IEEE Transactions on Visualization and Computer Graphics*, 13(4): 834–848.