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

Zexin Hu¹, Kun Hu¹,*, Clinton Mo¹, Lei Pan², Zhiyong Wang¹

¹The University of Sydney
²Civil 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 synthesizers are designed for structural, intermediate, and fine-grained level denoising purposes, which allow each synthesizer 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.

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.
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 denoiser 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.

For instance, 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.

**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. Kristof 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. Št'ava 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 higher level of control, they often involve a trade-off between realism/geomorphological correctness and controllability (Hnaidi et al. 2010; Gain, Marais, and Straßer 2009; Vanek et al. 2011). In other words, the increased control can typically result 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
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 with multi-level terrain synthesizers. 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 synthesizers 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 $X \in \mathbb{R}^{C \times W \times H}$. We denote the sketch guidance as a set $S = \{S_1, ..., S_n, ..., S_N\}$, with each user sketch $S_n \in \mathbb{R}^{C \times W \times H}$. These sketches and the terrain map can be viewed as one channel ($C = 1$) images. Fig. 1 depicts an example of paired $X$ and $S$, 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 $X \approx Z'(x = Z(X))$, where $x \in \mathbb{R}^d$ is the latent representation of $X$. 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):

$$\mathcal{L}(X, x_{rec}) = \frac{1}{C \sum_j H_j W_j} \left\| \phi_j(\hat{X}) - \phi_j(X) \right\|.$$  (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 $S_n$ into $s_n$ 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 $S$, 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 $s \in \mathbb{R}^d$. Note that $s$ aligns in dimensionality with the terrain latent representation $x$, 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 $x_t$ and $x_{t-1}$ can be formulated as:

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

where $q(x_t | x_{t-1})$ is a conditional probability for a forward diffusion process, which follows a Gaussian distribution with mean $\mu_t$ and variance $\Sigma_t$. In practice, starting from the original terrain data $x_0 = x$, the forward diffusion process is traceable as follows (Ho, Jain, and Abbeel 2020):

$$q(x_{1:T} | x_0) = \prod_{t=1}^{T} q(x_t | x_{t-1}).$$

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

$$x_t \sim q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\alpha_t} x_0, (1 - \alpha_t) \mathbf{I}).$$

Now the reverse process can be conducted based on a distribution, of which the probability is defined as $p(x_{t-1} | x_t)$. 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 $D_\theta(x_t | \theta)$, where $\theta$ contains learnable weights. Mathematically, we have:

$$p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)),$$

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

$$p_\theta(x_{0:T}) = p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t).$$

According to Luo (2022), $\theta$ can be optimized with a loss:

$$L_t = \mathbb{E}_{x_0, x_t, \epsilon} [||\epsilon - \epsilon_\theta(\sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon, t)||^2],$$

where $x_0$ is the noise free terrain latent representation, $\epsilon$ is the noise sampled from $\mathcal{N}(0, I)$ and the noise scheduler $\alpha_t = \prod_{s=0}^{t} \alpha_s$. Our primary goal is to utilize our neural network $D_\theta$ to estimate the noise $\epsilon$ as $\epsilon_\theta$.

### 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 $s$ and its latent representation $\bar{s}$ from users, a conditioned diffusion process can be formulated as (Song et al. 2020):

$$p_\theta(x_{0:T} | s) = p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t, s).$$

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

$$L_t = \mathbb{E}_{x_0, s, \epsilon} [||\epsilon - \epsilon_\theta(\sqrt{\alpha_t} x_0 | s + \sqrt{1 - \alpha_t} \epsilon, t)||^2].$$

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 $s$ shares identical dimension $d$ with the noise perturbed terrain representation $x_t$. Specifically, $s$ is concatenated with $x_t$ natively through a convolution layer. It is worth noting $s$ is constant throughout all diffusion steps $t$. To this end, the network is able to estimate $x_0 | s$ in an iterative manner, which is the generated output. We also enhance the control by employing skip connection, allowing the generated terrain to 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 sketchs.

### 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 synthesiser 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 $\theta_{\text{structural}}$ with:

$$L_{t}^{\text{structural}} = \mathbb{E}_{x_{0}|s,t,\epsilon}[(\epsilon - \epsilon_{\theta_{0}}(\sqrt{\alpha_{t}}x_{0}|s + \sqrt{1 - \alpha_{t}}\epsilon, t))^{2}].$$

(10)

Likewise, we further introduce an intermediate synthesiser $D_{\text{inter}}(x_{t}|\theta_{\text{inter}})$. 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 $\theta_{\text{inter}}$ is optimized by:

$$L_{t}^{\text{inter}} = \mathbb{E}_{x_{0}|s,t,\epsilon}[(\epsilon - \epsilon_{\theta_{0}}(\sqrt{\alpha_{t}}x_{0}|s + \sqrt{1 - \alpha_{t}}\epsilon, t))^{2}].$$

(11)

Lastly, the fine-grained synthesiser $D_{\text{fine-grained}}(x_{t}|\theta_{f})$ 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:

$$L_{t}^{\text{fine-grained}} = \mathbb{E}_{x_{0}|s,t,\epsilon}[(\epsilon - \epsilon_{\theta_{f}}(\sqrt{\alpha_{t}}x_{0}|s + \sqrt{1 - \alpha_{t}}\epsilon, t))^{2}].$$

(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 $144 \times 144$. 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 retain 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.

<table>
<thead>
<tr>
<th>Methods</th>
<th>FID ↓</th>
<th>MSE ↓</th>
<th>SSIM ↑</th>
<th>CD ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAN 2017</td>
<td>4.6599</td>
<td>0.0391</td>
<td>0.7288</td>
<td>0.5658</td>
</tr>
<tr>
<td>VQGAN 2021</td>
<td>6.5117</td>
<td>0.0548</td>
<td>0.5476</td>
<td>1.9774</td>
</tr>
<tr>
<td>SD 2022</td>
<td>8.2326</td>
<td>0.0854</td>
<td>0.5557</td>
<td>1.1171</td>
</tr>
<tr>
<td>ControlNet 2023</td>
<td>7.8923</td>
<td>0.0829</td>
<td>0.4508</td>
<td>1.4611</td>
</tr>
<tr>
<td>Gligen 2023</td>
<td>7.1527</td>
<td>0.0711</td>
<td>0.4394</td>
<td>1.6371</td>
</tr>
<tr>
<td>TDN (Ours)</td>
<td>0.4402</td>
<td>0.0059</td>
<td>0.8289</td>
<td>0.3545</td>
</tr>
</tbody>
</table>

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 VQGAN (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 sketch 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
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 synthesizers 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). 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 Sketch Autoencoders (SAEs)** removes the sketch autoencoders, and the sketches share the autoencoder 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

<table>
<thead>
<tr>
<th>Methods</th>
<th>FID</th>
<th>MSE</th>
<th>SSIM</th>
<th>CD</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o MLS</td>
<td>0.5071</td>
<td>0.0071</td>
<td>0.7703</td>
<td>0.4144</td>
</tr>
<tr>
<td>w/o ILS</td>
<td>1.8879</td>
<td>0.0177</td>
<td>0.8128</td>
<td>0.4037</td>
</tr>
<tr>
<td>w/o SAEs</td>
<td>1.9401</td>
<td>0.0312</td>
<td>0.5752</td>
<td>1.3725</td>
</tr>
<tr>
<td>w/o DSG</td>
<td>1.8957</td>
<td>0.0195</td>
<td>0.6975</td>
<td>0.9998</td>
</tr>
<tr>
<td>TDN (Ours)</td>
<td>0.4402</td>
<td>0.0059</td>
<td>0.8289</td>
<td>0.3545</td>
</tr>
</tbody>
</table>

Table 2: Ablation study on the proposed TDN.

## 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 plausible terrains. TDN can handle conflicting and complex user sketches. In the first example, the ridges overlap 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. 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
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.


