Pixels2Peaks: Converting Terrain Images to Heightmaps

ARYAMAAN JAIN, Inria, Université Côte d'Azur, France
JAMES GAIN, University of Cape Town, South Africa
GUILLAUME CORDONNIER, Inria, Université Côte d'Azur, France

Fig. 1. From an input image (left), Pixels2Peaks generates a complete heightmap of both the visible and occluded landforms (center), ready for offline rendering (right). [Input image from the GeoPose3K dataset [Brejcha and Čadík 2017]]

A common process in authoring digital scenes for games, films, and virtual environments is for artists to construct 3D geometry that matches a 2D perspective reference image. In the case of bare-earth terrain, this is typically a manual process since, unlike for trees and buildings, few inverse reconstruction methods currently exist. To address this, we introduce a method for automatically inferring a detailed, consistent, and complete terrain heightmap from a single photographic image. Our initial phase involves extracting camera parameters and a 3D pointmap from the input image, which is then transformed into a heightmap. However, this only recovers the unoccluded portions of the terrain visible from the perspective of the image. The next phase thus entails the generation of plausible occluded regions using a diffusion model trained on terrain elevation data. The entire process is guided by three consistency principles: geomorphological consistency (the features of the occluded terrain resemble the visible portions), hydrological consistency (the river network is uninterrupted and flows reliably), and view consistency (the shape of the rendered terrain accurately matches the input image). We demonstrate that our method obeys these principles, reliably generates terrains across various scales, and integrates with scene authoring workflows.

CCS Concepts: • **Computing methodologies** → **Shape modeling**.

ACM Reference Format:
Aryamaan Jain, James Gain, and Guillaume Cordonnier. 2026. Pixels2Peaks: Converting Terrain Images to Heightmaps. *ACM Trans. Graph.* 45, 4, Article 104 (July 2026), 17 pages. https://doi.org/10.1145/3811288

Authors' Contact Information: Aryamaan Jain, aryamaan.jain@inria.fr, Inria, Université Côte d'Azur, Sophia-Antipolis, France; James Gain, jgain@cs.uct.ac.za, University of Cape Town, Cape Town, South Africa; Guillaume Cordonnier, guillaume.cordonnier@inria.fr, Inria, Université Côte d'Azur, Sophia-Antipolis, France.

1 Introduction

Mountainous scenes form an iconic backdrop in many films, video games, and virtual worlds. The creation of such digital landscapes is a fundamental task in computer graphics, with a reference photograph or concept art as a common starting point. The journey from this 2D inspiration to a complete navigable 3D world is typically a laborious manual process. An artist must interpret the 2D reference image, model the matching 3D geometry, and ensure the final asset is both aesthetically pleasing and physically plausible. This workflow is time-consuming, requires significant expertise, and presents a major bottleneck in production pipelines.

We address the challenge of automating this process by reconstructing a complete and detailed 3D terrain heightmap, the standard representation used by artists, from a single 2D image (Figure 1). This provides an immediate base for the scene, allowing artists to focus on creative refinement rather than routine modeling. This represents an ill-posed inverse problem, since a single photograph captures only a partial, view-dependent snapshot of the terrain. Detail may be obscured by distance, but, more importantly, large sections are likely to be occluded from the viewpoint of the camera by folds in the landscape, particularly in mountainous terrain.

We believe that three consistency attributes are key to a viable solution for single-image terrain reconstruction:

(1) Geomorphological consistency — the generated landscape follows the geomorphological features of the visible regions.
(2) Hydrological consistency — the river network is locally and globally coherent, with unbroken channels that flow downhill to form proper drainage patterns.
(3) View consistency — the reconstructed heightfield matches the reference image when rendered from the original viewpoint.

Existing methods fail to achieve one or more of these key attributes. Photogrammetric techniques for 3D terrain reconstruction, such as Structure from Motion and Multi-View Stereo [Westoby et al. 2012], rely on constraints across multiple viewpoints and cannot be directly applied in a single-image scenario. Another route is to employ terrain-specific procedural modeling or simulation [Galin et al. 2019], but these generative methods do not support the tight constraints needed to match real-world reference images while at the same time retaining global feature consistency, such as a realistic flow network. Finally, while there is an extant solution to single-image terrain reconstruction [Takahashi et al. 2022], it does not achieve the threefold consistency defined here. In contrast, our model explicitly addresses visual plausibility, geomorphological, hydrological, and view consistency within the solution.

For our process, we first infer a 3D pointmap and camera parameters from the input image, and project them into an incomplete heightmap. This heightmap provides constraints for a guided diffusion model that adds coherent detail to the visible areas and inpaints the occluded areas. Our diffusion model operates on shifting tiles to allow large resolution outputs.

We achieve visual plausibility by training our model on a large-scale curated dataset of scanned real-world terrains. We enforce geomorphological, hydrological and view consistency through guidance terms incorporated into the iterative denoising of the diffusion process. These guidance terms favor coherent drainage networks, align slope and curvature statistics between visible and occluded regions, and ensure the alignment of the terrain ridges to the silhouettes from the original image. Furthermore, the heightfield resolution is automatically adapted to the scale of the terrain captured by the input image.

Our results demonstrate the generation of diverse and plausible terrains that are consistent with the input photograph. Our approach for estimating camera tilt and terrain scale outperforms horizon-line detectors and monocular geometry models. Ablation studies validate our guidance terms; for instance, view guidance reduces the reconstruction error, while hydrological guidance successfully resolves common artifacts, like unrealistic water basins. Compared to a generalist image-to-3D model, our method robustly generates physically-consistent terrains. We further demonstrate our method's versatility by creating text-to-terrain and sketch-to-terrain pipelines. The entire process executes in approximately 5 minutes on a GPU.

In summary, our key technical contributions are:

(1) A multi-stage inference process that converts a single source image into a terrain heightmap of appropriate resolution.
(2) A terrain-aware diffusion model with guidance terms designed to promote hydrological, geomorphological, and view consistency.
(3) A large 1.2 TB heightmap dataset curated to isolate landscapes with noticeable relief (the elevation difference, ranging from large hills to mountains) at 1m resolution.

2 Related work

Although there is little prior work that directly addresses the task of generating complete, high-fidelity terrain heightmaps from a single photograph, we do build upon several distinct lines of research. In particular, our work sits at the intersection of first-person terrain authoring, learning-based terrain models, and single-image 3D reconstruction.

2.1 First-person terrain authoring

Providing digital artists with authoring tools to interactively and controllably shape terrains has received considerable attention [Galin et al. 2019]. Generally, a painting or sketching metaphor is adopted along with either a top-down or first-person canvas perspective. Closest to our work are first-person sketching interfaces that allow an artist to draw internal and external silhouettes from which a plausible 3D terrain is inferred. The earliest methods [Cohen et al. 2000] established the principle of projecting the 2D end-points of a silhouette stroke onto either a ground plane or an existing heightfield in order to localize them in 3D. A terrain is then generated to match these constraining spatial curves by simple deformation of the ground plane [Cohen et al. 2000], localized warping of wavelet noise keyed to the multiresolution properties of the silhouette [Gain et al. 2009], minimal diffusion-based deformation of features on an existing terrain [Tasse et al. 2014a,b], or a patchwork approach of cutting and blending terrain fragments [dos Passos and Igarashi 2013]. While these authoring systems enforce view consistency by design, they often fail to respect geomorphological and hydrological consistency. This is partly because sketches do not contain the wealth of depth cues available in a photograph and partly because the underlying deformation and blending schemes are general and do not respect geomorphological properties. For instance, the vertical projection of the silhouette onto the ground plane is assumed to be linear [Cohen et al. 2000], edited separately from a top-down perspective [Gain et al. 2009], or derived by deforming or blending existing terrain features [dos Passos and Igarashi 2013; Tasse et al. 2014a]. This either requires supplying additional information not inherent in the original sketch or a significant distortion of the underlying geomorphology.

2.2 Learning-based terrain models

Alongside procedural methods and simulation, example-based learning represents one of three major strategies for terrain synthesis [Galin et al. 2019]. Because generative models are trained on a corpus of existing terrains from which a novel terrain can be synthesized, the results tend to be visually plausible. With careful model selection, they can also provide interactive feedback. Finally, with suitable conditioning, a variety of user controls, such as top-down sketching of ridges and rivers [Cai et al. 2022; Gain et al. 2015; Guérin et al. 2017; Hu et al. 2024; Lochner et al. 2023; Zhou et al. 2007], painting of elevation bands and terrain types [Gain et al. 2015; Guérin et al. 2017; Li et al. 2022; Perche et al. 2023; Valencia-Rosado et al. 2020], and specification of a general landscape class [Lochner et al. 2023; Zhao et al. 2019], are available. Apart from authoring terrains from scratch, another significant use case is the upsampling of existing low-resolution terrains [Argudo et al. 2018; Kubade et al. 2020; Zhao et al. 2019], possibly with additional conditioning on aerial images, and experiments with specific landforms [Li et al. 2022; Valencia-Rosado et al. 2020].

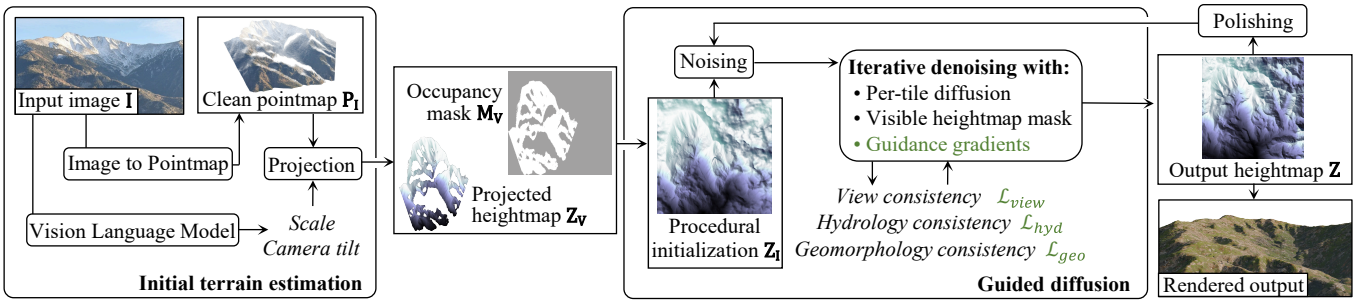


Fig. 2. Method overview. We start from an input photograph (©Pixabay), which is processed into a cleaned pointmap accompanied by scale and camera tilt parameters. This allows us to output an occupancy mask and an initial heightmap of the visible region. We complete the initial heightmap procedurally, and then refine it with a diffusion model, guided by hydrological, geomorphological, and view consistency. This leads to the final heightmap, after an optional polishing step.

Different approaches are employed, including regression [Argudo et al. 2018; Kubade et al. 2020], texture synthesis [Gain et al. 2015; Scott and Dodgson 2021; Zhou et al. 2007], generative adversarial networks [Cai et al. 2022; Guérin et al. 2017; Jain et al. 2024c; Naik et al. 2022; Perche et al. 2023; Spick and Walker 2019; Valencia-Rosado et al. 2020; Zhao et al. 2019], and diffusion models [Hu et al. 2024; Jain et al. 2022; Lochner et al. 2023].

Of these, diffusion models are the most promising for co-opting to single-view synthesis: they are more visually plausible while avoiding the training instability or repetition artifacts exhibited by CGANs [Guérin et al. 2017], and capture a wider range of landforms than texture synthesis. However, these models fail to enforce hydrological consistency and can lead to terrains with large-scale endorheic basins or small-scale pits. The one exception is the depression-breaching (a process that carves simulated drainage channels to route water out of closed sinks) terrain synthesis of Scott and Dodgson [2021], which uses a multi-resolution texture optimization approach to pit removal. Nevertheless, besides depression breaching, this approach only considers local information for generating the heightmap, which was shown to be inferior to diffusion models [Lochner et al. 2023]. In our work, we use depression breaching to *guide* the diffusion model to enforce hydrological consistency.

2.3 Single-image 3D reconstruction

Single-image 3D reconstruction is a fundamentally ill-posed problem, initially addressed with geometric cues and shape-from-shading [Horn 1986; Zhang et al. 1999]. Deep learning methods infer explicit 3D structures (voxel grids [Choy et al. 2016], point clouds [Fan et al. 2017], meshes [Wang et al. 2018]) or learn implicit representations [Mescheder et al. 2019]. Neural Radiance Fields (NeRFs) [Mildenhall et al. 2021], originally multi-view, were also adapted to learn geometry implicitly from a single image for novel view synthesis [Yu et al. 2021]. A key shift has been leveraging pre-trained 2D diffusion models as priors. DreamFusion [Poole et al. 2022] introduced Score Distillation Sampling (SDS) to optimize 3D representations like NeRFs without 3D data, later extended to single-image reconstruction [Liu et al. 2023a; Qian et al. 2023]. Recent breakthroughs like Hunyuan3D have introduced large-scale feed-forward models capable of producing high-fidelity general 3D assets directly from a single images [Hong et al. 2023; Li et al. 2025; Tencent 2025; Zhang et al. 2024]. Despite strong performance on common objects from large general datasets, these methods are not tailored to natural landscapes and do not enforce physical correctness.

Takahashi *et al.* [2022] do, however, incorporate terrain-specific adaptation in their image-to-terrain pipeline. This involves first extracting a depth map and shadow-free color map from the perspective of the camera, then rasterizing this onto a heightfield using a top-down orthogonal projection, and finally completing the elevation and color of the missing portions using a modified CGAN model. While the general progression from depth estimation, through projection, to generative model completion is similar, our architecture diverges significantly in other respects. We focus solely on elevation, do not assume a fixed terrain resolution, employ diffusion models instead of CGANs, train on a 1m resolution dataset (instead of 30m or 90m) and infer camera parameters rather than require them to be provided. Most crucially, we incorporate specific modifications into our model to address geomorphological, hydrological, and view consistency.

3 Overview

In the absence of constraints, synthesizing a landscape from a single perspective photograph is inherently ill-posed. Therefore, we first constrain the process by introducing a set of consistency attributes that act to enforce the plausibility of the resulting terrain and its alignment with the input, as follows:

Geomorphological consistency: While natural landscapes can exhibit marked diversity, this is less the case for mountain ranges, where features are shaped by a common tectonic, climatic, and lithologic history. In our context, this means that occluded regions should be structurally consistent with visible regions in the input photograph. We choose to interpret this through a statistical alignment of slope and curvature between visible and occluded regions, as these metrics have proven to be an effective and efficient basis for landform characterization [Argudo et al. 2025; Jasiewicz and Stepinski 2013].

Hydrological consistency: Large-scale mountain structures are subject to fluvial erosion, which prevents the formation of significant isolated valley structures, since these basins would otherwise fill to become massive lakes subject to rapid erosion through spillage. An overabundance of unrealistic pits and basins is a well-known shortcoming of localized terrain synthesis [Scott and Dodgson 2021]. To enforce hydrological consistency, there should always be a continuous downhill flow path for water to flow away from any given point [Martz and Garbrecht 1998]. The literature primarily offers two operations to modify a heightfield to respect this property: filling and breaching. Since filling produces unrealistically flat regions, we chose to rely on breaching. This can be recast as a minimization of the breaching volume, which is the volume of terrain that must be removed to create a continuous flow path through the barriers that enclose pits.

View consistency: There should be a close pixel-wise match between the placement of silhouettes (ridge lines) in the input photograph and a rendering of the generated terrain with the same view parameters. Furthermore, no additional spurious silhouettes should be introduced. We chose silhouettes as our basis, in preference to measures of texture similarity, because of their perceptual salience [Todd 2004].

Our goal is thus to synthesize a terrain heightmap from an input image while conforming to these constraints. We structure our approach (see Figure 2) into two primary phases: the construction of an incomplete terrain from a photograph, and subsequent completion of the full landscape through guided diffusion.

First, in order to generate a sparse heightmap covering the topographic areas visible in the photograph (Section 4), we estimate a sparse 3D pointmap and then segment out foreground objects and the background sky of the image. To resolve camera orientation and scene scale ambiguities, we query a Vision-Language Model (VLM) for semantic priors, such as plausible slope and elevation. These priors constrain the geometric projection, yielding a partial but appropriately scaled and oriented heightmap.

Second, we complete the occluded portions of the terrain (Section 5) by tiling the heightmap using a guided diffusion model. The tiling process is more expensive than a single inpainting step, but adapts well to a variety of sampling scales and heightmap resolutions. We guide the diffusion process with differentiable formulations of our three identified consistency constraints. Specifically, we minimize the difference in slope and curvature statistics between visible and occluded areas (geomorphological consistency), penalize sinks by comparing against a depression breached terrain (hydrological consistency), and prevent known silhouettes in the photograph from being altered (view consistency). Diffusion is initialized with multi-scale, view- and hydrologically-consistent noise and is followed by an iterative polishing loop to produce a complete, coherent, high-resolution output terrain.

The efficacy of our terrain completion phase is heavily reliant on sufficient high-resolution real-world terrain data, since this is needed to train the underlying diffusion model. To address this, we employ a multi-stage curation pipeline (Section 6) to filter source U.S. Geological Survey data [U.S. Geological Survey 2024] and assemble a 1.2TB collection of suitable terrain with relief.

4 Initial terrain estimation from a single image

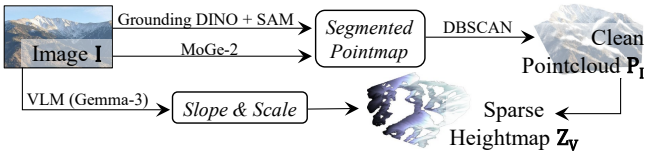


Fig. 3. Initial terrain estimation. Grounding DINO and SAM isolate mountains in the input image, while MoGe-2 extracts a pointmap. After DBSCAN cleans this pointmap, it is projected into a sparse heightmap using slope and scale parameters estimated by the Gemma-3 VLM.

In this section, we describe our method for deriving an initial, incomplete terrain heightmap covering the visible mountainous regions in a single input image (see Figure 3). We treat our photographic input as in-the-wild and make no assumptions as to the availability of either intrinsic or extrinsic camera parameters, such as focal length and camera orientation. This first estimation phase must thus overcome two obstacles: the local inherent geometric ambiguity of individual pixels and the unknown global scale and orientation of the scene. The former is addressed by applying a pretrained geometry estimation model (MoGe-2 [Wang et al. 2025]), which provides a 3D pointmap (a per-pixel 3D coordinate map). The latter is solved by querying a Vision-Language Model (VLM) [Google 2025] to deduce approximate camera pitch and scene extent. This allows us to calibrate the base plane and derive a partial heightfield from the pointmap.

4.1 Sparse 3D pointcloud reconstruction

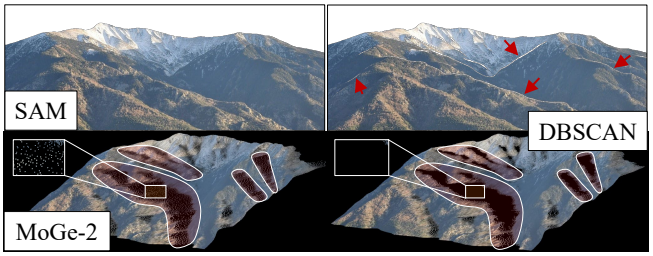


Fig. 4. Pointcloud extraction. The input image (as in Figure 2) is processed with Segment Anything Model (SAM) guided by Grounding-DINO to remove the foreground and sky (top-left). The input is then converted to a pointmap using MoGe-2 and masked by the SAM output, which, however, contains noise along the structure boundaries (noise highlighted; bottom-left). DBSCAN is applied to clean these noisy points (location indicated by red arrows in top-right and highlighted in bottom-right).

Our pipeline accepts a single input image **I**, which can take the form of a real-world photograph, landscape painting, or AI-generated scene (see Section 7). From this, our initial goal is to derive a 3D pointcloud covering the terrain visible in the image.

We first process the input image to mask out all foreground objects and background sky using the Segment Anything Model (SAM) [Kirillov et al. 2023]. This segmentation is additionally guided using bounding boxes provided by the Grounding-DINO model [Liu et al. 2023b] instructed with the prompt "all mountains" (Figure 4, top-left).

At the same time, we employ the MoGe-2 model [Wang et al. 2025] to map each image pixel to a 3D point. The resulting 3D pointmap obviates the need for explicit management of the camera intrinsics. This pointmap is masked out on the basis of the segmentation so that foreground and background pixel-points are excluded and only terrain coverage remains (Figure 4, bottom-left).

Finally, we filter the resulting pointmap using the DBSCAN [Ester et al. 1996] density-based clustering algorithm. This step is necessary to remove sparse noisy artifacts, a known issue of the pointmap prediction model that arises along object boundaries due to noisy training data [Wang et al. 2025]. As shown in Figure 4 (right), this filtering discards outliers that would otherwise corrupt the final heightmap, yielding the clean pointcloud $\mathbf{P_I}$.

4.2 Projection onto an incomplete heightmap

The standard toolchain in digital terrain processing relies heavily on a heightmap representation and so we must convert our cleaned 3D pointcloud to this format. For this purpose, we define a sparse heightmap $\mathbf{Z_V}$ on a 2D baseplane with an accompanying binary mask $\mathbf{M_V}$ to indicate which heightmap pixels contain projected points from $\mathbf{P_I}$. This heightmap is characterized by its resolution (the number of elevation samples), the sampling density (the separation between samples in *meters / pixel*), and, consequently, the scene extent (the landscape area covered by $\mathbf{Z_V}$ in *metres*). Unfortunately, this conversion requires knowledge of the camera's extrinsic parameters in order to orient the 3D point cloud. One option is to make use of a horizon line detector. However, it is more effective (as demonstrated in Section 7.2) to exploit landscape-specific priors.

The problem can be divided into two parts: determining camera orientation and the scene scale. We begin by reducing the degrees of orientation freedom by assuming no camera roll. In practice, photographers tend to naturally align their camera with the horizon. This is borne out by an examination of the GeoPose3K dataset [Brejcha and Čadík 2017], comprising ∼3k in-the-wild terrain photographs with known camera parameters, where the roll variation was found to be minimal ($\mu = 0.01°$, $\sigma = 1.62°$) compared to the wider distribution in pitch ($\mu = -0.04°$, $\sigma = 8.82°$).

Unfortunately, even a scalar estimation of the camera's pitch (or tilt), critical as it is to correctly orient the ground plane, remains ill-conditioned. Standard approaches, such as detecting vanishing points and horizon lines, are ill-suited to mountainous terrain lacking geometric priors like architectural straight lines.

Instead, we address this ambiguity by leveraging the semantic text-based interpretation afforded by Vision-Language Models (VLMs), which combine the best aspects of computer-vision encoders and large language models. We posit that a VLM can infer plausible physical properties of a depicted scene, which can then serve as priors to constrain the geometric reconstruction. In particular, we seek estimates for the mean slope s_{VLM} and elevation range $[z_{\min}, z_{\max}]_{\text{VLM}}$ to provide targets for the unknown pitch and scene scale, respectively. To obtain these priors, we query the Gemma-3 VLM [Google 2025] with the input image $\mathbf{I}$ and a structured text prompt requesting estimates of mean slope and minimum and maximum elevation across all visible mountains. To elicit a more reasoned and stable response, we employ Chain-of-Thought reasoning [Wei et al. 2022], prompted by an instruction to think step-by-step [Kojima et al. 2022]. To enhance robustness and mitigate prompt sensitivity, we then apply self-consistency [Wang et al. 2022] by taking an aggregate over several prompt variations (see Appendix), leading to final estimates for s_{VLM}, and $[z_{\min}, z_{\max}]_{\text{VLM}}$.

The VLM-estimate of slope s_{VLM} provides a prior for resolving the camera's optimal pitch angle, θ^*. To find θ^*, we undertake a grid search over a discretized range of candidate pitch angles $\{\theta_i\}$. For each θ_i, the point cloud $\mathbf{P_I}$ is rotated by the corresponding rotation matrix $\mathbf{R}(\theta_i)$ and orthographically projected onto the global ground plane to generate a temporary heightmap $\mathbf{Z}_i$ and occupancy mask $\mathbf{M}_i$. Finally, the optimal pitch θ^* is obtained by minimizing the absolute difference between the average rotation-invariant slope s_i [Jain et al. 2024a] of the unmasked pixels in $\mathbf{Z}_i$ and the VLM prediction:

$$\theta^* = \arg\min_{\theta_i} |s_i - s_{\text{VLM}}| \tag{1}$$

With the optimal pitch θ^* and orientation $\mathbf{R}(\theta^*)$ now established, what remains is a fit of the scene scale. This is achieved by uniformly scaling and translating the pointcloud associated with θ^*, so that it fits the elevation range $[z_{\min}, z_{\max}]_{\text{VLM}}$ predicted by the VLM. The scaled pointcloud is then projected onto the heightmap using a user-selected sampling density dx.

This process yields a final output for this phase: a *visible heightmap* consisting of an incomplete heightfield $\mathbf{Z_V}$ at a user-specified sampling density dx, and a corresponding binary occupancy mask $\mathbf{M_V}$ indicating the cells containing valid elevation data.

5 Guided diffusion for terrain completion

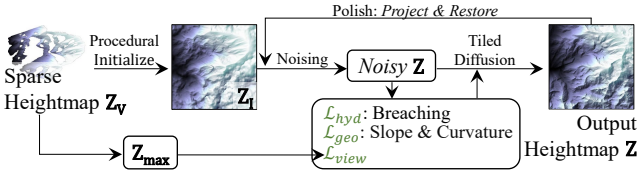


Fig. 5. Guided Diffusion Overview. The sparse heightmap is procedurally initialized and noised. Tiled reverse diffusion is then applied, guided by hydrological, geomorphological, and view consistencies, followed by a final polishing step to produce the output heightmap.

Having derived an initial heightmap for the visible terrain, our next task is to synthesize the occluded regions in a physically and visually consistent manner (see Figure 5). We frame this as a completion problem, addressed with a Diffusion Model adapted for inpainting (Section 5.1) and trained on a curated terrain dataset (Section 6). The core of our method is a novel guidance mechanism

that steers the reverse diffusion process towards hydrological, geomorphological, and view consistency (Section 5.2). For this guided process to converge effectively, we employ a procedural initialization scheme (Section 5.3). We also use a resolution-adaptive tiled approach (Section 5.4) to ensure scalability to large terrains. Finally, an additional iterative polishing step (Section 5.5) refines the results to achieve high-quality outputs.

5.1 Terrain completion

Our method builds upon Denoising Diffusion Probabilistic Models (DDPMs) [Ho et al. 2020]. These combine a forward diffusion process, which gradually corrupts data by applying noise, with a learned reverse denoising process.

In our case, the forward process $q(\mathbf{Z}_0, t)$ successively injects Gaussian noise into an initially clean heightmap data sample $\mathbf{Z}_0$ over time $t \in [0, 1]$. Note that we can directly sample a noisy version $\mathbf{Z}_t$ for any arbitrary timestep t from $\mathbf{Z}_0$ in closed form:

$$\mathbf{Z}_t = q(\mathbf{Z}_0, t) = \sqrt{\bar{\alpha}_t}\mathbf{Z}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{2}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a random Gaussian noise sample and the schedule $\bar{\alpha}_t$ is a monotonically decreasing function that controls the noise variance over time, such that $t = 0$ represents the clean data and $t = 1$ represents pure noise.

The reverse process learns to recover structure through denoising. It relies on a U-Net [Ronneberger et al. 2015], $\boldsymbol{\epsilon}_\theta(\mathbf{Z}_t, t)$, which is trained to predict the noise component $\boldsymbol{\epsilon}$ in a noisy sample $\mathbf{Z}_t$. During reverse diffusion, the model iteratively refines a noisy heightfield $\mathbf{Z}_t$ to produce a cleaner version $\mathbf{Z}_{t-\Delta t}$, culminating in $\mathbf{Z}_0$.

This is designed as a completion process, so the original visible regions in $\mathbf{Z}_\mathbf{V}$ must be recovered intact. To achieve this, we introduce the correctly noised version of the visible heightmap $\mathbf{Z}_\mathbf{V}$ masked according to the occupancy mask $\mathbf{M}_\mathbf{V}$, following the RePaint [Lugmayr et al. 2022] strategy:

$$\mathbf{Z}_t \leftarrow (1 - \mathbf{M}_\mathbf{V}) \odot \mathbf{Z}_t + \mathbf{M}_\mathbf{V} \odot q(\mathbf{Z}_\mathbf{V}, t) \tag{3}$$

This step anchors the regions visible in the photograph, while allowing the rest of the terrain to be synthesized freely. This reinjection is disabled during the final denoising steps ($t < 0.05$) to enable smooth blending between known and synthesized regions and refine away any inaccuracies in the initial heightmap estimation.

5.2 Guided reverse process

The core of our contribution is a mechanism for steering the reverse diffusion process towards hydrological, geomorphological, and view consistency. This is built on the foundation of gradient-based guidance [Dhariwal and Nichol 2021], in which a guidance term enforces desired image properties.

First, we use the network's noise prediction $\boldsymbol{\epsilon}_\theta(\mathbf{Z}_t, t)$ to estimate the expected final heightmap $\hat{\mathbf{Z}}_0$ [Ho et al. 2020]:

$$\hat{\mathbf{Z}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{Z}_t - \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{Z}_t, t)\right) \tag{4}$$

From the predicted heightmap $\hat{\mathbf{Z}}_0$, we extract three penalty terms that measure the degree of hydrological $\mathcal{L}_{\text{hyd}}$, geomorphological $\mathcal{L}_{\text{geo}}$, and view $\mathcal{L}_{\text{view}}$ inconsistency. These are combined into a single weighted composite guidance loss $\mathcal{L}_{\text{guide}}$, as follows:

$$\mathcal{L}_{\text{guide}} = \lambda_{\text{hyd}}\mathcal{L}_{\text{hyd}} + \lambda_{\text{geo}}\mathcal{L}_{\text{geo}} + \lambda_{\text{view}}\mathcal{L}_{\text{view}} \tag{5}$$

where λ_{hyd}, λ_{geo}, and λ_{view} are scalar weights controlling the contribution of each respective consistency loss. The gradient of the loss with respect to the current noisy state $\mathbf{Z}_t$ is then used to perturb the original noise prediction:

$$\boldsymbol{\epsilon}'_t = \boldsymbol{\epsilon}_\theta(\mathbf{Z}_t, t) - s \cdot \sqrt{1 - \bar{\alpha}_t} \cdot \nabla_{\mathbf{Z}_t}\mathcal{L}_{\text{guide}} \tag{6}$$

where s is the guidance scale. The modified noise estimate $\boldsymbol{\epsilon}'_t$ is then used in the scheduler's denoising step to compute $\mathbf{Z}_{t-\Delta t}$. We now define the individual consistency guidance terms.

Hydrological Guidance. This loss term improves flow consistency across the generated surface by penalizing depressions, which are local minima or sinks where water would pool erroneously. The loss uses a novel depression-breaching function H, which algorithmically carves drainage channels from these sinks to lower ground, ensuring every point on the surface has a continuous downhill path. This process creates a breached or hydrologically conditioned surface, essential for realistic erosion channels.

Formally, hydrological loss is defined as the difference between a smoothed and downsampled version of the prediction, with and without breaching:

$$\mathcal{L}_{\text{hyd}} = \mathbb{E}[(D_\sigma(\hat{\mathbf{Z}}_0) - \text{sg}[H(D_\sigma(\hat{\mathbf{Z}}_0))])^2] \tag{7}$$

Here, D_σ is a Gaussian smoothing and downsampling operator. Smoothing removes high-frequency noise that can manifest as spurious depressions during denoising, while downsampling mitigates the formation of artificially steep valleys. The stop-gradient operator $\text{sg}[\cdot]$ detaches the breached output from the computational graph, effectively treating it as a fixed target that is not subject to backpropagation of its gradients.

Since depression breaching is performed on every iteration, it can potentially degrade the performance of the denoising process. To overcome this hurdle, we extend the FastFlow framework [Jain et al. 2024b] to develop a novel GPU-parallel implementation of depression breaching H (see Algorithm 1) that uses pointer jumping to reduce the number of iterations from $O(n)$ to $O(\log n)$. The FastFlow framework provides a set of directed flow trees over the heightfield, where pointers are directed from a cell (the source) to a lower neighbour (the recipient). Depression breaching ensures that a recipient is always lower than its source by some ϵ. This is performed in parallel through a pointer jumping (or doubling) scheme.

Geomorphological Guidance. Our goal with this loss term is to encourage geomorphological stationarity, in which the geomorphological characteristics of the visible (known) regions are replicated in the occluded (unknown) regions, thus creating a unified seamless landscape. To achieve this, our geomorphological guidance loss $\mathcal{L}_{\text{geo}}$ minimizes the difference between the mean slope and curvature of the known and unknown regions, a concept analogous to style loss in neural style transfer [Gatys et al. 2015] and weak stationarity in texture synthesis [Efros and Leung 1999]. While numerous terrain descriptors are available [Argudo et al. 2025], we chose slope and curvature as being among the most fundamental [Moore et al. 1991].

Algorithm 1: GPU-Breach

Input : Elevation grid $\mathbf{Z}$, Slope differential ϵ
Output: Breached elevation grid $\mathbf{Z}$

Compute FastFlow recipients for all grid cells
$N \leftarrow$ number of grid cells in $\mathbf{Z}$
for $i \leftarrow 1$ **to** $\log_2(N)$ **do**
  foreach *grid cell* c *in* $\mathbf{Z}$ *in parallel* **do**
    $\mathbf{Z}[\text{recipient of } c] \leftarrow$ atomic-min$(\mathbf{Z}[c] - 2^{i-1}\epsilon, \mathbf{Z}[\text{recipient of } c])$
    recipient of $c \leftarrow$ recipient of recipient of c

Our geomorphological loss is derived from the same smoothed and downsampled prediction $D_\sigma(\hat{\mathbf{Z}}_0)$ used for hydrological guidance. Now, let $S(\cdot)$ and $C(\cdot)$ be operators that compute local slope and curvature (where $C(\cdot)$ computes the Laplacian using a standard 5-point stencil, which serves as a linear approximation of mean curvature), and let $\mathbf{M}_\sigma$ be the binary mask identifying the known visible regions at the downsampled resolution. The loss $\mathcal{L}_{\text{geo}}$ is then defined as:

$$\mathcal{L}_{\text{geo}} = \lambda_s \left(\mathbb{E}_{1-\mathbf{M}_\sigma}[S(D_\sigma(\hat{\mathbf{Z}}_0))] - \text{sg}[\mathbb{E}_{\mathbf{M}_\sigma}[S(D_\sigma(\hat{\mathbf{Z}}_0))]] \right)^2 + \lambda_c \left(\mathbb{E}_{1-\mathbf{M}_\sigma}[C(D_\sigma(\hat{\mathbf{Z}}_0))] - \text{sg}[\mathbb{E}_{\mathbf{M}_\sigma}[C(D_\sigma(\hat{\mathbf{Z}}_0))]] \right)^2 \quad (8)$$

where $\mathbb{E}_{\mathbf{M}_\sigma}[\cdot]$ and $\mathbb{E}_{1-\mathbf{M}_\sigma}[\cdot]$ denote the expectation over the known and unknown regions, respectively. As before, the stop-gradient operator $\text{sg}(\cdot)$ ensures that the guidance only affects the unknown regions. The scalars λ_s and λ_c control the relative weighting of the slope and curvature terms.

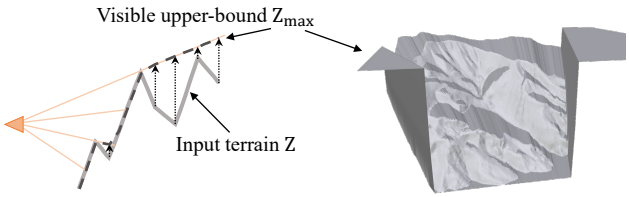


Fig. 6. We estimate the maximum terrain elevation envelope $\mathbf{Z}_{\text{max}}$ that preserves view consistency, by casting rays into the scene and intersecting them with the terrain. Hidden parts of the terrain are projected to the elevation of the lowest skimming ray immediately above them. We show a 3D view of $\mathbf{Z}_{\text{max}}$, with light grey for the visible elevations, and darker grey (dashed in 2D) for the projections of the hidden regions onto the envelope.

View Guidance. This loss term mitigates against any of the newly synthesized terrain obscuring the known terrain when viewed from the perspective of the original camera. Consequently, it also preserves the salience of the internal and external landscape silhouettes in the input image. We enforce this constraint by pre-computing a per-cell upper elevation bound over the visible heightmap $\mathbf{Z}_\mathbf{V}$. This takes the form of a bounding heightmap $\mathbf{Z}_{\text{max}}$, whose entries prescribe the maximum allowable elevation for cells in the regions being inferred. If an elevation in the inferred region breaches this cap, it will become visible from the camera perspective. It is computed (see Algorithm 2 and Figure 6) by casting rays from the camera viewpoint and matching elevations in the unknown regions to those of surface skimming rays. The resulting bounding heightmap $\mathbf{Z}_{\text{max}}$ is used as a soft constraint via the view-guidance loss:

$$\mathcal{L}_{\text{view}} = \mathbb{E}_{1-\mathbf{M}}[(\text{ReLU}(\hat{\mathbf{Z}}_0 - \text{sg}[\mathbf{Z}_{\text{max}}]))^2] \quad (9)$$

This loss penalizes values in $\hat{\mathbf{Z}}_0$ that exceed $\mathbf{Z}_{\text{max}}$. The one-sided ReLU function ensures that penalization only applies for elevations causing an occlusion (i.e., positive values), while the stop-gradient on $\mathbf{Z}_{\text{max}}$ prevents backpropagation.

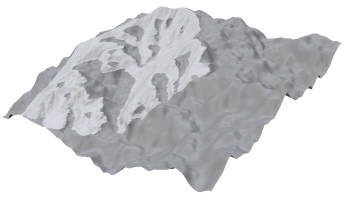
Algorithm 2: Ray-trace View Bound

Input : Elevation grid $\mathbf{Z}$, Camera parameters $\mathbf{V}$
Output: View bound $\mathbf{Z}_{\text{max}}$

$\mathcal{R} \leftarrow$ Get-Rays$(\mathbf{V})$
$\mathbf{Z}_{\text{max}} \leftarrow \infty$
while *any ray in* $\mathcal{R}$ *exists* **do**
  foreach *ray* $r \in \mathcal{R}$ *in parallel* **do**
    $r_{xyz} \leftarrow r_{xyz} + \delta r_{direction}$
    if r_x, r_y *out of grid* $\mathbf{Z}$ *bounds* **then**
      Terminate-Ray$(r, \mathcal{R})$
    else if $r_z \leq \mathbf{Z}[\lfloor r_x \rfloor, \lfloor r_y \rfloor]$ **then**
      Terminate-Ray$(r, \mathcal{R})$ ▷ Ray hits Z
    else
      $\mathbf{Z}_{\text{max}}[\lfloor r_x \rfloor, \lfloor r_y \rfloor] \leftarrow$ atomic-min$(\mathbf{Z}_{\text{max}}[\lfloor r_x \rfloor, \lfloor r_y \rfloor], r_z)$

5.3 Initialization for guided inpainting

In practice, our guidance mechanism struggles to produce plausible results when starting from a pure noise state ($t = 1$), for two reasons. First, the initial noise represents a highly variable initial topography that is usually too far from the input view to be correctable via the view consistency term. Second, the breaching operation within the hydrological consistency term requires the terrain to exhibit at least some structure and has little meaning for pure noise.



We address this by starting the reverse diffusion process not from pure noise at $t = 1$, corresponding to a fully learned approach, but from an intermediate timestep $t \approx 0.5$. The initial state at this timestep is based on noising a procedural initial guess (dark grey in the inset figure). This initial guess $\mathbf{Z}_\text{I}$ is constructed (see Algorithm 3) using an iterative procedural noise process, inspired by multi-scale value noise [Ebert et al. 2002], with added hydrological breaching and view clamping.

From a coarse starting scale, we add uniform noise to the unmasked (occluded) regions $(1 - \mathbf{M}_\mathbf{V})$, smooth with Laplacian diffusion, correct for hydrological consistency with breaching, clamp to the maximum elevation allowed for by view consistency, and

finally upsample and repeat the process at the next finer scale. As observed by Scott and Dodgson [2021], applying breaching at every scale also serves to prevent the formation of unnaturally narrow corrective valleys. The pass of Laplacian diffusion is used to reduce high-frequency artifacts and sharp discontinuities.

Algorithm 3: Noise Inpainting for Diffusion Initialization

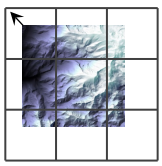
Input : Elevation grid $\mathbf{Z_V}$, Mask $\mathbf{M_V}$, View bound $\mathbf{Z}_{\max}$, Num octaves O
Output : Inpainted elevation $\mathbf{Z_I}$

Let $\mathcal{U}_k$, $\mathcal{D}_k$ be up/down-sampling operators by factor k
Function Constrain($\mathbf{Z}, \mathbf{M}, \mathbf{Z}_{max}$)
  $\mathbf{Z} \leftarrow$ Smooth($\mathbf{Z}$) ▷ *e.g.*, Laplacian Diffusion
  ▷ Breach unmasked regions
  $\mathbf{Z}' \leftarrow (1 - \mathbf{M}) \odot \mathbf{Z} + \mathbf{M} \odot \infty$
  $\mathbf{Z} \leftarrow (1 - \mathbf{M}) \odot$ GPU-Breach($\mathbf{Z}'$) $+ \mathbf{M} \odot \mathbf{Z}$
  return Minimum($\mathbf{Z}, \mathbf{Z}_{\max}$) ▷ View constraint
▷ Initialize at the coarsest level
$\mathbf{Z_I} \leftarrow$ Constrain($\mathcal{D}_{2^O}(\mathbf{Z_V}, \mathbf{M_V}, \mathbf{Z}_{\max})$)
▷ Iterate from coarse to fine
for $k \leftarrow O - 1$ **to** 0 **do**
  $\mathbf{Z}^k, \mathbf{M}^k, \mathbf{Z}^k_{\max} \leftarrow \mathcal{D}_{2^k}(\mathbf{Z_V}, \mathbf{M_V}, \mathbf{Z}_{\max})$
  $\boldsymbol{\epsilon} \leftarrow$ scale-dependent noise
  $\mathbf{Z}_{\text{up}} \leftarrow \mathcal{U}_2(\mathbf{Z_I} + \boldsymbol{\epsilon})$
  $\mathbf{Z}_{\text{merged}} \leftarrow (1 - \mathbf{M}^k) \odot \mathbf{Z}_{\text{up}} + \mathbf{M}^k \odot \mathbf{Z}^k$
  $\mathbf{Z_I} \leftarrow$ Constrain($\mathbf{Z}_{\text{merged}}, \mathbf{M}^k, \mathbf{Z}^k_{\max}$)

This process produces a procedural multi-scale heightfield $\mathbf{Z_I}$. Following a process akin to SDEdit [Meng et al. 2021], we then derive a starting state for our diffusion model by adding Gaussian noise to $\mathbf{Z_I}$, consistent with the noise schedule at timestep $t \approx 0.5$. This noisy state then serves as the starting point for the guided denoising process.

5.4 Resolution adaptive tiled diffusion

Depending on the scale of the scene, landscape images may range in extent anywhere from a few hundred metres in the case of small hills shot in close up to tens of kilometres for a panorama of a mountain range. With a fixed sampling density (*e.g.*, 4 m/pixel), this variation in scene extent results in a wide range of output heightmap resolutions. To accommodate this variability, we adopt a tiled denoising approach, based on the SpotDiffusion sliding window strategy [Frolov et al. 2025].



In general terms, tiled denoising operates by performing multiple diffusion processes in parallel over different portions of the heightmap and then combining the outputs. The details differ as to how tiles are positioned and blended. We choose SpotDiffusion [Frolov et al. 2025], which places tiles contiguously without overlap and removes seams by performing a global random translation of the tile set after each iteration, because it balances quality and computation cost. The original work of SpotDiffusion is targeted at panoramic images with a long strip of tiles that are shifted horizontally between iterations. We extend this to the 2D domain and perform random shifts along both the x- and y-axes. The inset illustrates this with a grid of parallel denoising processes and an arrow indicating the terrain offset toward the top-left.

We further improve scalability by automatically selecting the sampling density dx based on the scene extent, as predicted in Section 4. For instance, rather than denoising a 16km^2 terrain at dx = 1m, which produces a heightfield with 16000^2 grid cells, we can denoise it with a model trained at dx = 16m, resulting in a 1000^2 grid, and significantly reduce computation cost. To enable this, we train a collection of diffusion models to produce 512^2 heightfields at dx = 1m, 2m, 4m, 8m, and 16m by cropping and downsampling the tiles in our 1m dataset (Section 6).

5.5 Final polishing

At this juncture, our model synthesizes a terrain that is view-, geomorphologically- and hydrologically-consistent. Nevertheless, some spurious local inconsistencies remain (see Figure 15d). We therefore incorporate a final iterative super-resolution and refinement step to correct these remaining issues and elevate the quality of the result. Unlike the main diffusion phase, which incorporates consistency as soft guidance terms, this polishing step enforces compliance through hard projection. This iterative refinement loop is documented in Algorithm 4, and summarized below.

Algorithm 4: Iterative Heightmap Polishing

Input : Elevation grid $\mathbf{Z}$, View bound $\mathbf{Z}_{\max}$, Upfactor $\mathbf{u}$
Output : Polished heightmap $\mathbf{Z}$

Let $\mathcal{U}_k$, $\mathcal{D}_k$ be up/down-sampling operators by fixed factor k
$\mathbf{Z} \leftarrow \mathcal{U}_{\mathbf{u}}(\mathbf{Z})$
$\mathcal{T} \leftarrow [t_1, t_2, \ldots, t_n]$ ▷ $t_i > t_{i+1},\ t_1 \sim 0.05,\ n \sim 3$
for $t \in \mathcal{T}$ **do**
  $\mathbf{Z} \leftarrow \mathcal{U}_k$(GPU-Breach($\mathcal{D}_k(\mathbf{Z})$))
  $\mathbf{Z} \leftarrow$ Minimum($\mathbf{Z}, \mathbf{Z}_{\max}$)
  $\mathbf{Z}_t \leftarrow q(\mathbf{Z}, t)$ ▷ Add noise
  $\mathbf{Z} \leftarrow$ Guided-Denoise($\mathbf{Z}_t, t$)

To begin, the heightmap is upsampled via bicubic interpolation to reduce the sampling interval dx. Each subsequent refinement iteration consists of two phases:

(1) **Constraint Projection:** The heightmap $\mathbf{Z}$ is projected back onto the space of valid solutions. The view constraint is enforced via element-wise clamping against the bounding heightmap $\mathbf{Z}_{\max}$, while the flow constraint is enforced by breaching after downsampling (as before preventing the formation of unnatural steep-sided valleys).
(2) **Realism Restoration:** To reduce artifacts arising from the projection, an SDEdit-style diffusion step is applied to the projected output after noising it as the initial state. Starting from a timestep close to completion (initially $\sim$ 0.05) facilitates a short trajectory, sufficient to smooth artifacts and

restore high-frequency details without significantly altering the now constraint-compliant structure. On each iteration, the timestep is brought successively closer to completion.

This strategy successively refines the heightmap, improving constraint adherence while enhancing realism.

6 Terrain dataset curation

Training our diffusion model is data intensive and its inference quality relies on a dataset of real-world mountainous terrain data that is large-scale, high-resolution, freely available, with noticeable relief and cleaned. Unfortunately, as shown in Table 1, no existing dataset meets all these requirements, necessitating our own curation process.

Table 1. A comparison of terrain datasets. Our dataset provides a unique combination of open access, overall size, high individual tile resolution, small sampling distance (dx), and data cleaning.

Dataset	Open	Size	Tile Res.	dx(m)	Cleaned
[Guérin et al. 2017]	No	NA	3600^2	30	No
[Argudo et al. 2018]	**Yes**	3.5GB	Variable	2	No
[Perche et al. 2023]	No*	NA	1000^2	5, 30	**Yes**
[Lochner et al. 2023]	No	NA	256^2	2.39	**Yes**
[Czerkawski et al. 2025]	**Yes**	973GB	356^2	30	No
Ours	**Yes**	**1.2TB**	$\mathbf{10012}^2$	**1**	**Yes**

* Provide links to raw source but not a cleaned dataset.

Our training is predicated on both a short sampling distance (dx = 1m) and high tile resolution ($10{,}012^2$). This combination captures fine detail for close-up views and supports downsampling to multiple sampling distances (1m, 2m, 4m, 8m, 16m), while maintaining a viable tile resolution. Furthermore, raw elevation data often contains artifacts, missing regions, or non-mountainous features. A dedicated cleaning and curation process is therefore necessary to ensure the model learns from high-quality, relevant terrain.

Our starting point is the USGS 3DEP 1m dataset [U.S. Geological Survey 2024], which contains over 100,000 high-resolution ($10{,}012^2$) tiles at a 1-meter sampling distance. This is fed into a multi-step filtering pipeline, as follows:

Heuristic thumbnail culling. The entire collection of raw TIFF files of the USGS 3DEP is unmanageably large. Instead, we perform a first round of data reduction based on the 300×385 JPEG thumbnail images accompanying each TIFF. Absolute elevation is not available because these are 8-bit normalized greyscale images. However, this is sufficient for culling tiles with data voids covering more than half the image, because this manifests as blacked out pixels (Figure 7a). This is followed by filtering out relatively flat tiles employing a texture-based heuristic. Each thumbnail is segmented into a grid of 32×32 patches and we calculate the mean of the per patch standard deviation in greyscale value. Below a mean threshold of 5, the terrain is considered flat (Figure 7b) and the image is culled. Together, these two heuristics effectively reduce our candidate pool by 70%, to approximately 30,000 images.

Iterative learning-based filtering. Some relatively flat regions, particularly those with complex non-mountainous river systems and erosion gulleys, slip past our variance-based culling. To address this, we iteratively fine-tune a ResNet-18 [He et al. 2016] image classifier. We begin with a manually-classified seed dataset of 1000 negative (e.g., meandering river systems, as in Figure 7c) and 1500 positive (e.g., dendritic ridges and valleys, as in Figure 7d) cases. In each cycle, the trained model classifies a new subset of data, which we then manually review for misclassifications, adding the corrected samples back to the training set. After three such refinement iterations, our dataset is filtered down to approximately 9,000 high-confidence thumbnails.

Final curation of full-resolution tiles. We download the corresponding full-resolution TIFF files for tiles that pass the initial rounds of culling. Then, on the basis of the detailed data newly available, a final two-step verification is performed: tiles containing any data voids are removed, as are those without significant vertical relief (quantified as an elevation range of less than 200 meters). This yields a final curated dataset of 4,002 high-quality TIFF files, each with a $10{,}012^2$ resolution at a 1-meter sampling density, totaling 1.2 TB.

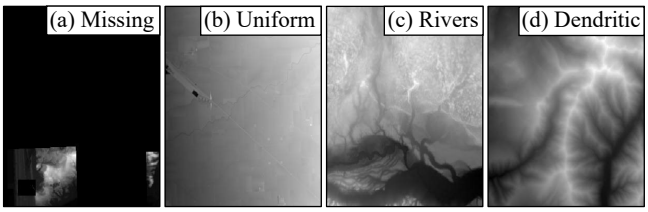


Fig. 7. Our multi-stage filtering pipeline for dataset curation, illustrated with thumbnail examples. Heuristic filters first discard thumbnails with large data voids (a) or low texture variance, indicative of flat terrain (b). Subsequently, a trained classifier distinguishes unwanted features (c) from high-relief terrain (d).

7 Results and evaluation

We implement our method using Python and PyTorch, and optimize using custom CUDA kernels, with Google Gemma-3-27b-it [Google 2025] applied to VLM queries. The terrain heightmaps depicted in the paper have been rendered in Blender. Our code, trained models, and dataset are available at: https://gitlab.inria.fr/landscapes/pixels2peaks.

Training and inference were performed on a node equipped with an Intel Xeon 6710E CPU clocked at 3.2GHz, with 256GB RAM and an Nvidia H100 NVL GPU with 94GB of memory.

We trained five separate diffusion models to generate 512^2 terrain heightmaps at 1m, 2m, 4m, 8m, and 16m resolutions. Each 6-level U-Net model (71M parameters, 2 ResNet layers per block, with self-attention at lower resolution scales) uses DDPM (T=1000) and was trained from scratch for 500,000 steps (10 days per model) using cropped and resized data. Training employed the AdamW optimizer [Loshchilov and Hutter 2017] with batch size 16 and a cosine annealing learning rate schedule decaying from 10^{-4} to 10^{-6}.

For inference, we set the default guidance terms to the values shown in Table 2.

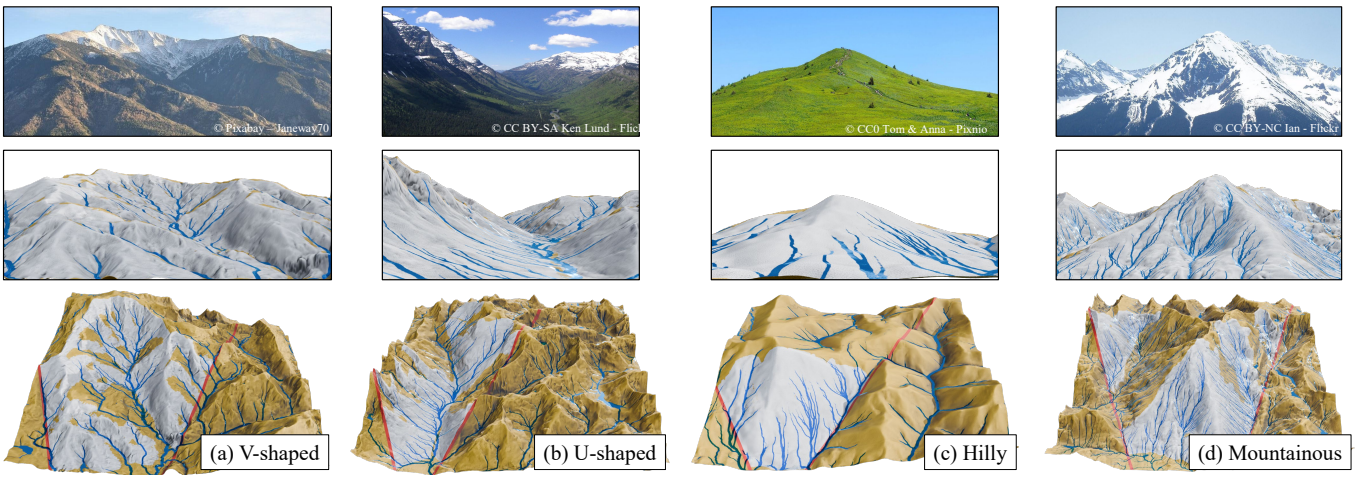


Fig. 8. Diversity in landscape forms (from left to right): a V-shaped valley, a U-shaped glacial valley, a nearby hill, and a high-elevation mountain. Given the input photography (top row), we show the reconstructed landscape from the same camera viewpoint (middle) and a pulled-back view (bottom row). To provide a better visual handle on geomorphological and view consistency, we shade regions of the terrain visible from the camera viewpoint in grey, and generated occluded regions in taupe. The lateral limits of the camera frustrum are indicated with red lines.

Table 2. Guidance parameters

Parameter	s	λ_{hyd}	λ_{geo}	λ_{view}	λ_{s}	λ_{c}
Value	10^0	10^8	10^6	10^5	10^0	10^0

Breaking down performance, the entire image-to-heightmap pipeline completes within roughly 5 minutes. The pre-diffusion steps require $\approx$ 80s, of which VLM inference accounts for $\approx$ 60s. The guided diffusion process generates a 1024^2 terrain using a 512^2 model in $\approx$ 150s. The subsequent polishing step, which includes minor corrections and upsampling to 2048^2 takes $\approx$ 70s.

The main factor that scales computation cost is the number of tiles in the denoising grid. By default, the sampling density dx is chosen so as to produce a 1024^2 resolution heightfield before final upsampling, with an associated 3×3 grid of 9 tiles. Each tile measures 512^2, though an enlarged border is needed to accommodate random shifts of the heightfield in x and y. Occasionally, a higher resolution heightfield and larger tile grid may be required for particularly expansive and distant panoramas.

7.1 Qualitative results: terrain diversity

Our terrain synthesis is capable of matching a variety of landforms. This is demonstrated in Figure 8(top) using source photographs of landscapes incorporating: a) V-shaped valleys formed by fluvial processes, b) U-shaped valleys resulting from glacial erosion, c) low-lying hills, and d) high-elevation mountain peaks.

We observe that the generated heightmaps (middle and bottom rows) consistently match the landforms in the photographs, that the mountain silhouettes are preserved, and that our results are free of large water bodies typical of depressions in hydrologically inconsistent terrains. Qualitatively, from a geomorphological-consistency perspective, there are no abrupt transitions in the type of landforms between visible and occluded areas. Likewise, violations in view consistency are rare, as demonstrated by the scarcity of taupe (occluded) pixels in the camera perspective (middle row).

Figure 22 provides additional qualitative results for a selection of input images from the GeoPose3K dataset, demonstrating robustness and the generation of coherent occluded regions from novel viewpoints.

7.2 Validation: initial terrain estimation

We now turn to a validation of the individual components of our algorithm, beginning with initial terrain estimation. First, we evaluate the pointmap estimation provided by MoGe-2 to establish its quantitative accuracy and qualitative robustness on terrain data. Following this, we evaluate the necessity of using a Visual Language Model (VLM) to estimate the tilt of the camera and the elevation range captured in the image.

MoGe-2 quantitative validation. We evaluated MoGe-2's terrain depth estimation using 512 Phong-shaded synthetic images and ground-truth depth maps, rendered from manually arranged viewpoints across our heightmap dataset. After aligning MoGe-2's predictions to the ground truth via least-squares scale and shift optimization, the model achieved an Absolute Relative Error (AbsRel) of 0.082 ± 0.034 and a threshold accuracy ($\delta < 1.25$) of $92.06\% \pm 6.58\%$.

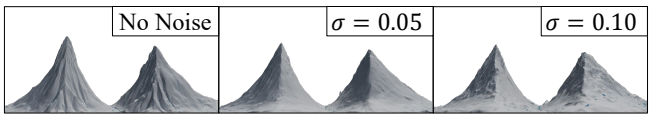


Fig. 9. MoGe-2 robustness to noise. Pointmaps are perturbed with multiplicative Gaussian noise: (Left) $\sigma = 0$, (Center) $\sigma = 0.05$, and (Right) $\sigma = 0.10$. The input image is from Figure 19(b).

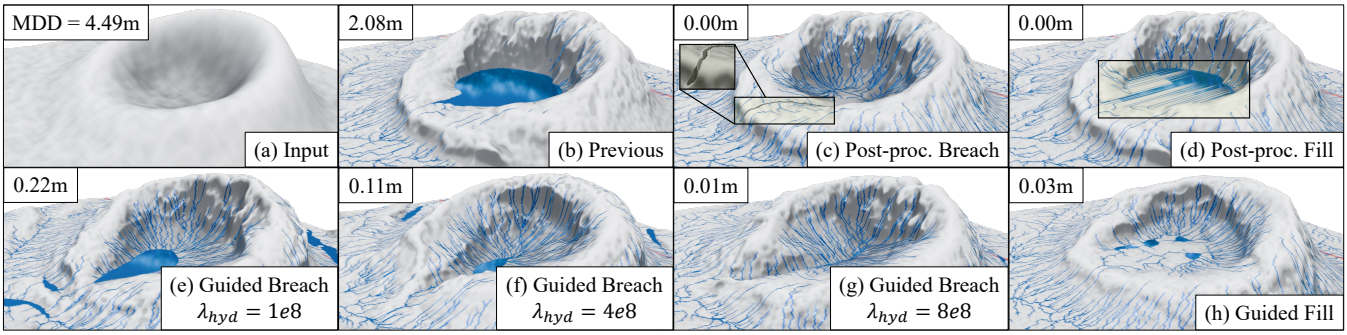


Fig. 10. An evaluation of flow guidance. We apply our diffusion model to a crater-shaped terrain (a). Diffusion without guidance [Hu et al. 2024] leaves a large endorheic basin (b). Post-processed breaching (c) or filling (d) of the depression results in an unnaturally deep and narrow trench or a flat sheet. In contrast, our method (bottom row) forms a valley to accommodate water outflow, controlled by the guidance weight (bottom left and center). As an option, the user can instead apply guidance with filling (bottom right).

MoGe-2 Robustness Validation. To assess robustness to MoGe-2 estimation errors, we perturb the estimated pointmaps with multiplicative Gaussian noise. Specifically, spatial coordinates are scaled by a noise map sampled from $\mathcal{N}(1, \sigma)$ using standard deviations $\sigma \in \{0.00, 0.05, 0.10\}$ (Figure 9) to simulate depth-proportional sensor measurement errors. While this noise inherently degrades high-frequency details, the diffusion model effectively smooths the perturbations to synthesize plausible landscapes that preserve the low-frequency structure of the input image.

VLM tilt validation. As a baseline against which to evaluate camera tilt prediction, we make use of the GeoPose3K dataset [Brejcha and Čadík 2017]. This consists of 3,111 photographs of the Alps, with accurate camera position and orientation metadata derived by refining rough initial camera parameters using alignment with scanned real-world heightmaps. This baseline is compared against three alternatives (see Figure 11): a state-of-the-art horizon-line detector (SOFI) [Janampa and Pattichis 2024], directly querying a VLM for tilt (VLM(t)), and our search procedure for inferring tilt from VLM-predicted slope (VLM(s)). Our VLM(s) (median error = 7.18°) significantly ($p < 0.01$) outperforms both VLM(t) (median error = 9.07°) and SOFI (median error = 13.16°) based on a Kruskal-Wallis test ($H(2) = 610.4$, $p < 0.001$, $\varepsilon^2 = 0.066$) followed by post-hoc Dunn's pairwise testing. The relatively poor performance of SOFI is explained by its reliance on geometric cues, which are often absent in mountainous scenery. Less clear is why slope prediction (VLM(s)) outperforms direct tilt prediction (VLM(t)). We hypothesize that this may be due to a higher frequency of terrain slope labelling in the internet-sourced training set of the VLM.

VLM elevation range validation. For estimating the difference between the minimum and maximum elevation in an image, we compared the accuracy of our VLM approach against MoGe-2 [Wang et al. 2025], a state-of-the-art metric monocular geometry model. On a 16-image dataset derived from Google Earth with known viewpoints and elevation ranges, our VLM (mean error: 27.59% ± 19.08%) proves considerably more accurate than MoGe-2 (91.35% ± 4.57%). For an explanation, it is illustrative to consider the two cases in Figure 12. The Mt. Everest image (left) has a ground truth

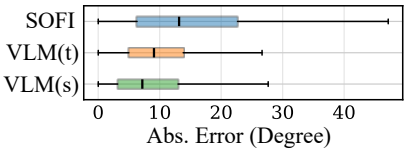


Fig. 11. To evaluate the options for camera tilt prediction, we compare the absolute error in degrees of a horizon line detector [Janampa and Pattichis 2024] (SOFI) against a VLM, instructed to predict tilt directly (VLM(t)) or the terrain slope followed by an optimization search (VLM(s)).

elevation range of ~ 3,000m (from 5,700m to 8,700m) accurately inferred by the VLM (a 3,484m range) compared to MoGe-2 (a 77m range). The VLM's reasoning confirmed that it correctly identified the mountain itself along with the names of the visible glaciers.

A similar pattern holds for less immediately identifiable cases, such as the mountain in Idaho (right) with a ~1,400m ground truth range, where the VLM (1,605m range) again outperforms MoGe-2 (203m range). While the VLM fails to precisely geolocate this image, it is able to reason by analogy using similar scenery and vegetation found in California's San Gabriel range.

The VLM's reasoning traces highlight that it establishes scale by semantically recognizing the scene, either directly or by analogy, and then accessing its stored geospatial knowledge. Monocular predictors, which infer scale from visual cues, such as atmospheric perspective, are far less reliable for large-scale natural landscapes.

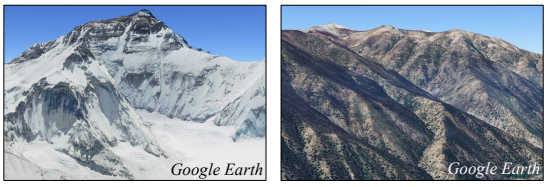


Fig. 12. Two illustrative cases of the landscape images selected for validation of our elevation range prediction against that of a metric geometry model (MoGe-2).

7.3 Validation: guided diffusion

Next, we consider the guided diffusion process, including the pre-process of initialization and the post-process of polishing, from the perspective of our three key consistency attributes (see Section 3).

Hydrological consistency. Endorheic basins with inflow but no outflow are a flaw common to many example-based terrain generation schemes. This gives rise to hydrologically unrealistic features, such as downward sloping valleys that are walled off at the end. In Figure 10, we explore guidance-based correction of one such challenging case: a prominent crater, with a mean depression depth (MDD) of 4.49m (Figure 10(a)). Here, MDD is defined as the average depth of water trapped on the terrain after flow simulation.

In the top row, we consider the performance of an unguided baseline and traditional post-processing. Unfortunately, these strategies are suboptimal: unguided diffusion [Hu et al. 2024] using SDEdit-style partial noising ($t \approx 0.5$) followed by denoising reduces the crater wall somewhat but still leaves an MDD = 2.08m basin behind. Post-hoc corrections fix the endorheic basin completely, but at the expense of realism. Breaching (c) introduces artificially-narrow channels, while filling (d) produces unnaturally flat planar surfaces.

In the bottom row, we integrate guided breaching (e-g) and filling (h) into the diffusion loop, yielding more natural results. Guided breaching is as described in Section 5.2, while for guided depression filling we replace Algorithm 1 with a filling algorithm [Jain et al. 2024b].

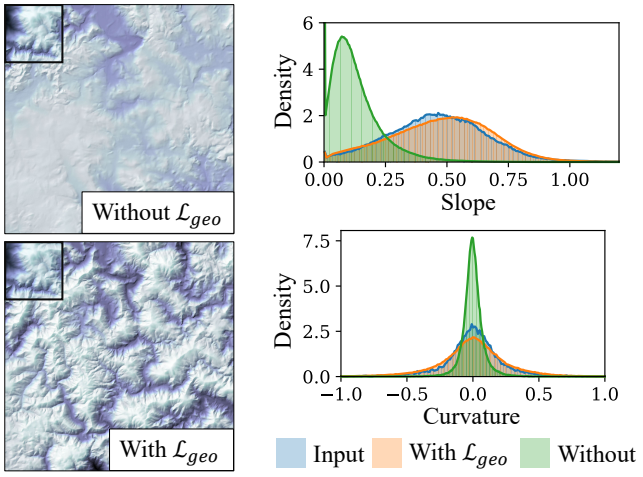


Fig. 13. An evaluation of geomorphological guidance. Better completion of a terrain (inset) is achieved with (bottom left) than without (top left) the $\mathcal{L}_{\text{geo}}$ guidance term. The improvement is further supported by the slope and curvature histograms (right); these compare the input terrain (blue) to completions with (orange) and without (green) guidance.

Geomorphological consistency. To evaluate geomorphological guidance, we perform a terrain completion task on a 2048^2 grid of which the top-left 512^2 tile is provided as a fixed prior (see Figure 13(left)). In the absence of guidance, a clear visual distinction exists between the fixed and synthesized areas. This qualitative observation is substantiated by a quantitative analysis of slope and curvature distributions, which reveals a much stronger statistical match between the two regions when guidance is applied (see Figure 13(right)).

View consistency. In Figure 14, we ablate the view guidance term using the mountain range photograph from Figure 8(d). The impact of view guidance is particularly evident in the foreground and near silhouettes, both of which are visually salient. In quantitative terms, we define the Visible View Error (VVE) as the average over the heightfield of $\mathbf{M}_V \odot |\mathbf{Z} - \mathbf{Z}_V|$, and Occluded View Error (OVE) as the average over the heightfield of $(1 - \mathbf{M}_V) \odot \max(\mathbf{Z} - \mathbf{Z}_{\max}, 0)$. We observe that without view guidance, VVE is 2.36m and OVE is 8.52m, whereas with view guidance, these errors are 2.32m and 0.09m, respectively.

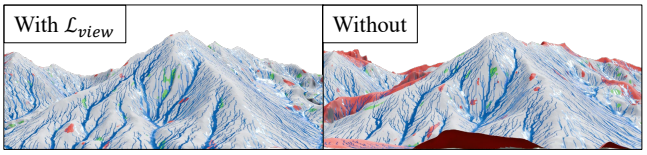


Fig. 14. An evaluation of view guidance. The peak from Figure 8(d), synthesized with (left) and without (right) the $\mathcal{L}_{\text{view}}$ guidance term. Red indicates an over-prediction of the elevation ($\mathbf{Z} > \mathbf{Z}_{\max} + 5$), which does not respect terrain silhouettes, and green shows an under-estimation ($\mathbf{Z} < \mathbf{Z}_{\text{V}} - 5$) of the elevation compared to the heightmap projected from the input image.

Initialization and polishing validation. We demonstrate the efficacy of the initialization (Algorithm 3) and polishing steps (Algorithm 4) in Figure 15, using the text-generated image from Figure 19(b) as input. We consider four configurations: (a) initializing with a pure noise state (diffusion starts at $t = 1$) for occluded regions; (b) initializing with Laplacian diffusion to interpolate visible into occluded regions; (c) using Algorithm 3 for initialization but omitting polishing, and (d) leveraging the complete pipeline.

As with the previous experiment, overestimated and underestimated view errors are coded in red and green, respectively. Hydrological inconsistencies are highlighted as depression maps in the inset plan views. For each configuration, we estimate hydrological consistency with the mean depression depth (MDD), and view consistency with Visible View Error (VVE) and Occluded View Error (OVE) metrics defined earlier. We observe that omitting initialization entirely (a) prevents convergence (MDD = 0.85m, VVE = 2.26, OVE = 186.34m). Initialization with Laplacian diffusion (b-left) produces degraded results even after guidance (b-right, MDD = 0.49m, VVE = 2.74m, OVE = 88.64m). In contrast, a consistent initialization (c) yields coherent terrain with minor inconsistencies (MDD = 0.09m, VVE = 2.20m, OVE = 1.86m), which polishing (d) effectively resolves (MDD = 0.01m, VVE = 2.06m, OVE = 0.14m).

Setting hydrological guidance weighting. To evaluate the impact of the hydrological guidance weight λ_{hyd} on synthesis quality, we undertook a comparison of real and generated terrains at various scales $\lambda_{\text{hyd}} = 0, 10^0, 10^1, \ldots, 10^{11}$ on the basis of Fréchet Inception Distance (FID) [Heusel et al. 2017], Kernel Inception Distance (KID) [Bińkowski et al. 2018], and mean depression depth

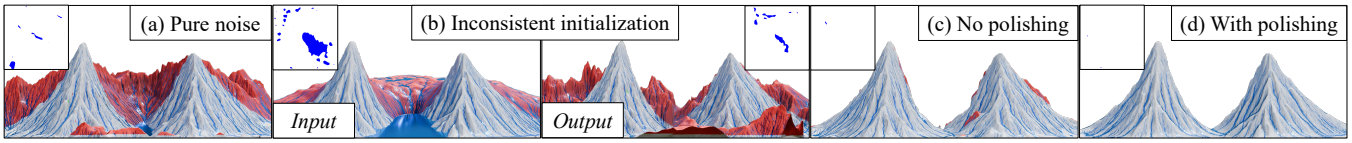


Fig. 15. The impact of initialization and polishing. Even with guidance, inconsistent initialization with noise (a) or Laplacian interpolation of the visible topography (b) fails to preserve view and hydrological consistency. Our initialization leads to a consistent terrain (c), with the resolution of marginal artifacts through polishing (d). The insets show depression maps. Input image from Figure 19(b).

(MDD) metrics (see Figure 16). FID measures the distance between Gaussian approximations of real and generated Inception features, whereas KID is an unbiased alternative that measures their Maximum Mean Discrepancy without distributional assumptions. We generated 10,000 unconditional diffusion samples for each λ_{hyd} scale and collected 10,000 real-world samples as a baseline. Results reveal a clear optimal weighting range of $10^8 < \lambda_{\text{hyd}} < 10^9$ with the lowest FID (11.01–12.99) and KID (0.009–0.011) scores, and MDD stabilized below 0.01m. A plateau of stable but weaker scores appears initially ($\lambda_{\text{hyd}} < 10^6$), likely arising from disconnected water flow paths in independently generated tiles, an artifact not present in real terrain. Past the optimum, FID and KID degrade sharply as the denoising diffusion process is suppressed in favour of flow routing, causing a drift away from the real data distribution. Note that the U-shape is in line with the application of guidance in other image diffusion contexts [Dhariwal and Nichol 2021; Ho and Salimans 2022].

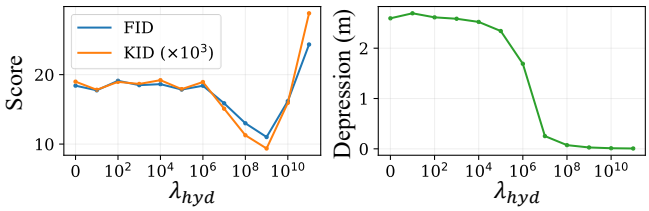


Fig. 16. The influence of hydrology guidance on quality and depression depth. We plot the FID and KID (left), and mean depression depth (right) of generated samples with increasing levels of guidance. Quality increases with sufficient hydrological guidance, reducing unnatural flow patterns, but then decreases when the guidance is overly strict.

7.4 Applications

Beyond validating hydrological, geomorphological, and view consistency, we further demonstrate that our method extends effectively to non-photographic inputs, in the form of text, paintings, and sketches. This can be achieved by coupling a pre-trained X-to-image model with our image-to-terrain model, effectively bolting on the former as a upstream module to our pipeline, creating an X-to-terrain model, where X is a placeholder for a range of input media.

Figure 19 demonstrates terrain synthesis from various forms of non-photographic input: (1) a painting, Thron der Götter by Werner Hahn (CC), which can be supplied directly to our existing pipeline; (2) an image produced by a text-to-image model [Labs et al. 2025], which we adapt for text-to-terrain synthesis; and (3) an image generated by a sketch-to-image model [Labs et al. 2025], extended for sketch-to-terrain.

Sketch-to-Terrain. Figure 17 provides a side-by-side comparison of our sketch-to-terrain pipeline with the sketching system of Gain *et al.* [2009]. To enable a direct comparison, we traced a sketch from their teaser and processed it through our pipeline (Figure 19(c)). The sketch was first converted to a landscape image using FLUX.1 Kontext [Labs et al. 2025], which was in turn converted to a heightfield terrain by our pipeline. Our method's use of diffusion priors and hydrological guidance leads to a higher-fidelity terrain, with more consistent water flow paths.

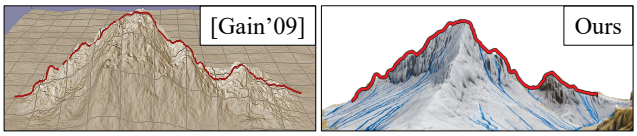


Fig. 17. Comparison with the sketch-based terrain synthesis of Gain *et al.* [2009]. (Left) Terrain generated using their system, with the input sketch shown in red. (Right) The same sketch is replicated and processed through our sketch-to-terrain pipeline (also in Figure 19(c)).

Text-to-Terrain. We compare and contrast our extended text-to-terrain pipeline against MESA [Czerkawski et al. 2025], which uses a Stable Diffusion model trained on terrain data with synthetic prompts. Unfortunately, since MESA's outputs are not on a metric scale, we cannot directly compare quantitative metrics, such as MDD. Instead, we visualize the derived depression maps, which show regions where water accumulates. Figure 18 presents 8 of the 9 samples available on the MESA project page, excluding only the depiction of plains, and should be contrasted with the right-most depression map derived from Figure 19(b). Because MESA does not incorporate an explicit hydrological mechanism, water tends to accumulate. Notice the presence of larger depressions compared to our previous examples.

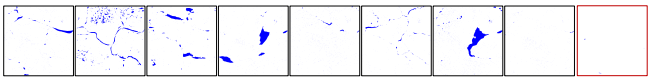


Fig. 18. Depression maps from MESA [Czerkawski et al. 2025] outputs show extensive water accumulation from unrealistic depressions in contrast to the text-to-image sample from Figure 19 (red outline).

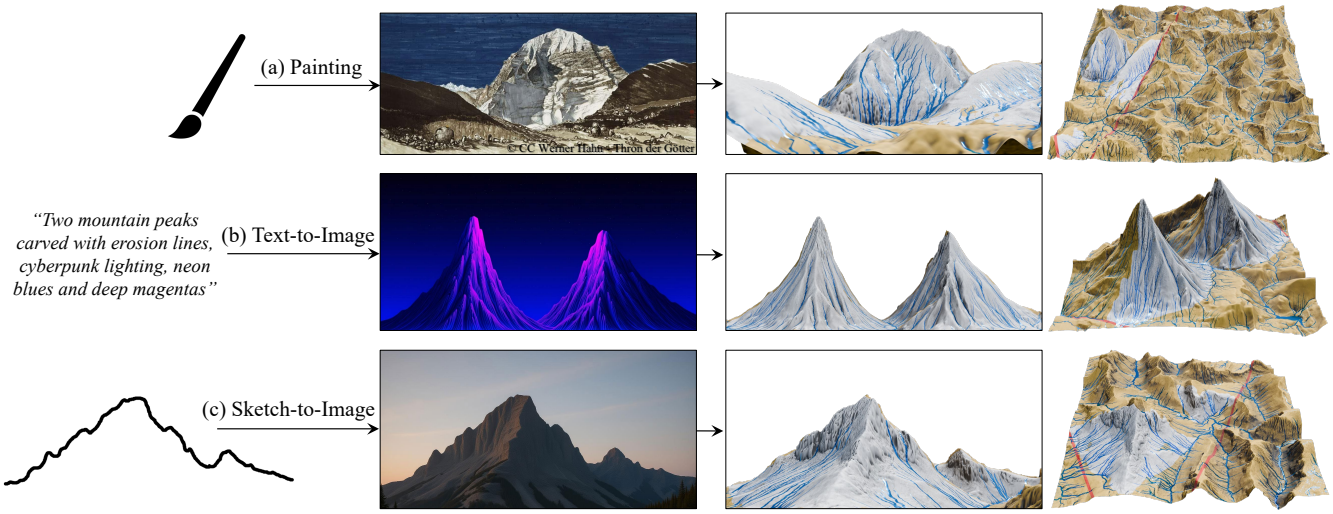


Fig. 19. Diverse input modalities, including painting, text, and sketching (column 1), are used to obtain images (column 2). From these images, we synthesize the corresponding landscapes (column 3: same camera viewpoint; column 4: pulled-back view).

Furthermore, MESA relies on structured prompts and struggles with free-form inputs. As shown in Figure 19(b), our method successfully generates "two mountain peaks", while MESA produces an out-of-distribution canyon-like structure for the same prompt (see inset). Conversely, prompts like *"broadleaf forests and hills in Germany in August"* (their second sample) are a challenge for our pipeline, as text-to-image models often introduce large, undesired foreground elements in such cases.

Image-to-3D. To demonstrate the necessity of a domain-specific approach, we compare our method against a general-purpose image-to-3D model, Hunyuan3D-2.1 [Tencent 2025], using 667 low-tilt ($< 2°$) landscape images from the GeoPose3K dataset [Brejcha and Čadík 2017]. Of these, we excluded 246 samples (37%) where the baseline generated geomorphologically implausible artifacts, such as floating islands or hollow cubes. To enable direct quantitative comparison of the remaining 421 samples, we converted Hunyuan3D-2.1's meshes to heightmaps via top-down orthographic projection and then applied a common VLM-derived scaling factor.

To assess hydrology, we computed mean depression depth (MDD) averaged over all generated samples. Because Hunyuan3D-2.1 is a generalist model lacking terrain-specific adaptation, it introduces unnatural sinks (MDD = 9.64m) not present in our model (MDD = 0.11m).

To assess realism, we measure the feature space distance between generated terrains and a reference set of 167,370 real-world multi-scale (dx from 1m to 42m) terrain feature patches from our dataset. Using DINOv2 [Oquab et al. 2023] embeddings, we compute the mean cosine distance to the 5 nearest real-world neighbors, where a lower score indicates higher realism. By this metric, our pipeline (mean = 0.221, SD = 0.052) outperforms Hunyuan3D-2.1 (mean = 0.312, SD = 0.104). We further confirmed this using Kernel Inception Distance (KID) against 16,000 ground-truth samples spanning dx from 1m to 42m, where our method achieved a better KID (0.0187) than the baseline (0.1304).

Qualitatively, Hunyuan3D-2.1's lack of terrain-specific training results in outputs lacking natural landscape features, as demonstrated by the Hunyuan3D-2.1 generated samples in Figure 20.

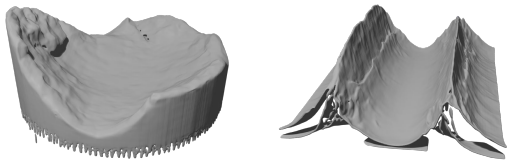


Fig. 20. Terrain samples generated by Hunyuan3D-2.1, lacking natural landscape features. For a comparison with the input images and our outputs, contrast the left sample against Figure 8(b) and the right sample against Figure 19(b).

7.5 Limitations

Our method has several limitations that open up avenues for future research. The workflow is specialized for high-relief landscapes and struggles with out-of-distribution planar topographies (Figure 21, left), such as flatlands, desertscapes and meandering rivers. Performance also degrades in challenging photographic conditions, such as heavy fog, which can corrupt the initial geometry.

Furthermore, our sequential pipeline is susceptible to error propagation. For instance, an incorrect initial slope estimation can cascade into an implausibly steep heightmap (Figure 21, right), or undesirable foreground elements may be introduced during the text-to-terrain process. Tiled diffusion remains computationally expensive,

limiting scalability for very large terrains. Tiling can also introduce minor but visible seams, though the tile offsetting strategy reduces this significantly. Finally, small inconsistencies such as unanticipated depressions can stem from inherent trade-offs between guidance terms in the diffusion denoising process.

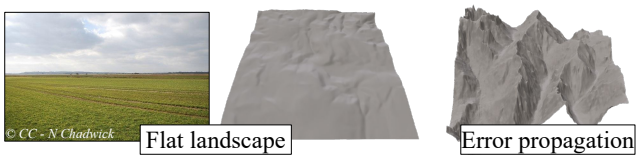


Fig. 21. Failure cases. Left: Out-of-distribution flat landscapes fail to reconstruct as perfectly planar, instead producing unintended elevation artifacts. Right: Incorrect VLM slope estimation cascades into inaccurate tilt and unnaturally steep terrain that mismatches the input (Figure 22, row 5). To demonstrate this, the VLM slope was manually forced to 60°.

8 Conclusion

We have presented Pixels2Peaks, a novel method for converting a photographic image of a landscape into a physically plausible 3D bare-earth heightmap. Our work addresses the fundamentally ill-posed problem of reconstructing unseen geometry by introducing a framework guided by three critical principles: geomorphological, hydrological, and view consistency. This ensures that the generated terrain not only conforms visually to the input image but also adheres to the natural laws that govern landscape formation.

Our technical contribution is a multi-stage pipeline that first resolves camera and scale ambiguities, producing a sparse but correctly-oriented initial heightmap of the visible terrain. We then employ a guided diffusion model, trained on a new, large-scale 1.2 TB dataset of high-resolution real-world terrain, to generate the occluded regions. The core of our method lies in the guidance terms that steer the diffusion process, ensuring the preservation of ridge-line silhouettes, the formation of coherent drainage networks, and the statistical alignment of landform features between visible and synthesized areas.

Through qualitative and quantitative evaluation, we have demonstrated that our approach reliably generates diverse landscapes that are consistent with their source images. Our ablation studies validate the necessity of each guidance component, and our comparisons show a significant improvement in physical plausibility and hydrological correctness over general-purpose models. Furthermore, we have demonstrated the flexibility of our core method by incorporating it into text-to-terrain and sketch-to-terrain pipelines, allowing additional forms of creative control.

While Pixels2Peaks represents a step forward, limitations remain that suggest promising directions for future research, such as exploring end-to-end architectures to improve robustness, expanding the model to cater to a wider range of topographies, and investigating more efficient synthesis strategies. Ultimately, our work helps bridge the gap between 2D inspiration and navigable 3D worlds, providing a tool for automating a key bottleneck in the creation of digital environments.

Acknowledgments

This project was sponsored by the Agence Nationale de la Recherche project Invterra ANR-22-CE33-0012-01 and research and software donations from Adobe Inc.

References

Oscar Argudo, Antoni Chica, and Carlos Andujar. 2018. Terrain Super-resolution through Aerial Imagery and Fully Convolutional Networks. *Computer Graphics Forum* 37, 2 (2018), 101–110.

O. Argudo, E. Guérin, H. Schott, and E. Galin. 2025. Terrain descriptors for landscape synthesis, analysis and simulation. *Computer Graphics Forum* 44, 2 (2025), e70080.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. 2018. Demystifying mmd gans. *arXiv:1801.01401* (2018).

Jan Brejcha and Martin Čadík. 2017. GeoPose3K: Mountain landscape dataset for camera pose estimation in outdoor environments. *Image and Vision Computing* 66 (2017), 1–14.

Xingquan Cai, Mengyao Xi, Nu Yu, Zhe Yang, and Haiyan Sun. 2022. A Terrain Elevation Map Generation Method Based on Self-Attention Mechanism and Multifeature Sketch. *Computational Intelligence and Neuroscience* 2022, 1 (2022), 9481445. doi:10.1155/2022/9481445

Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 2016. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision*. 628–644.

Jonathan M Cohen, John F Hughes, and Robert C Zeleznik. 2000. Harold: A world made of drawings. In *Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*. 83–90.

Mikolaj Czerkawski, Rosalie Martin, Romain Rouffet, et al. 2025. MESA: Text-driven terrain generation using latent diffusion and global copernicus data. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 3067–3075.

Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems* 34 (2021), 8780–8794.

Vladimir Alves dos Passos and Takeo Igarashi. 2013. LandSketch: a first person point-of-view example-based terrain modeling approach. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*. 61–68. doi:10.1145/2487381.2487382

David S Ebert, F Kenton Musgrave, Darwyn Peachey, Ken Perlin, and Steve Worley. 2002. *Texturing and modeling: a procedural approach*. Elsevier.

A.A. Efros and T.K. Leung. 1999. Texture synthesis by non-parametric sampling. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Vol. 2. 1033–1038 vol.2. doi:10.1109/ICCV.1999.790383

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, Vol. 96. 226–231.

Haoqiang Fan, Hao Su, and Leonidas J Guibas. 2017. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 605–613.

Stanislav Frolov, Brian B Moser, and Andreas Dengel. 2025. Spotdiffusion: A fast approach for seamless panorama generation over time. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2073–2081.

James Gain, Patrick Marais, and Wolfgang Strasser. 2009. Terrain Sketching. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games (I3D '09)*. 31––38. doi:10.1145/1507149.1507155

J. Gain, B. Merry, and P. Marais. 2015. Parallel, Realistic and Controllable Terrain Synthesis. *Computer Graphics Forum* 34, 2 (2015), 105–116. doi:10.1111/cgf.12545

Eric Galin, Eric Guérin, Adrien Peytavie, Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, and James Gain. 2019. A Review of Digital Terrain Modeling. *Computer Graphics Forum* 38, 2 (2019), 553–577.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2015. A neural algorithm of artistic style. *arXiv:1508.06576* (2015).

Google. 2025. Gemma 3. (2025). https://goo.gle/Gemma3Report

Éric Guérin, Julie Digne, Eric Galin, Adrien Peytavie, Christian Wolf, Bedrich Benes, and Benoît Martinez. 2017. Interactive example-based terrain authoring with conditional generative adversarial networks. *Acm Transactions on Graphics* 36, 6 (2017), 1–13.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* 30 (2017).

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

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

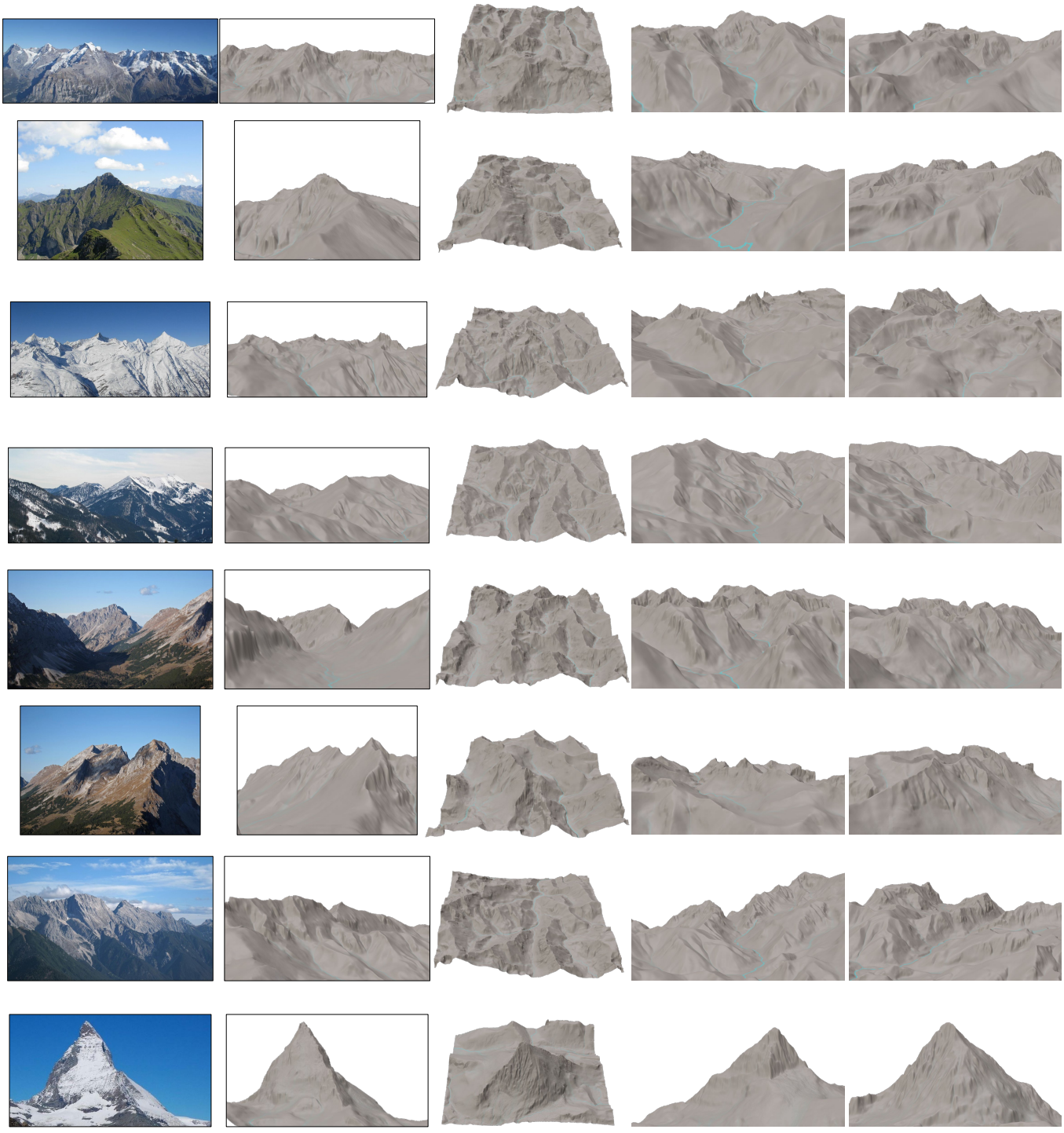


Fig. 22. Qualitative results of Pixels2Peaks applied to eight sample images from the GeoPose3K dataset. From left to right, the columns display: the original input photograph, the reconstructed heightmap rendered from the same camera viewpoint, a pulled-back view revealing the complete terrain, and two novel random viewpoints of occluded regions.

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. 2023. Lrm: Large reconstruction model for single image to 3d. *arXiv:2311.04400* (2023).

Berthold Horn. 1986. *Robot vision*. MIT press.

Zexin Hu, Kun Hu, Clinton Mo, Lei Pan, and Zhiyong Wang. 2024. Terrain Diffusion Network: Climatic-Aware Terrain Generation with Geological Sketch Guidance. *Proceedings of the AAAI Conference on Artificial Intelligence* 38, 11 (2024), 12565–12573. doi:10.1609/aaai.v38i11.29150

Aryamaan Jain, Bedrich Benes, and Guillaume Cordonnier. 2024a. Efficient Debris-flow Simulation for Steep Terrain Erosion. *ACM Transactions on Graphics* 43, 4, Article 58 (July 2024), 11 pages. doi:10.1145/3658213

Aryamaan Jain, Bernhard Kerbl, James Gain, Brandon Finley, and Guillaume Cordonnier. 2024b. FastFlow: GPU Acceleration of Flow and Depression Routing for Landscape Simulation. In *Computer Graphics Forum*, Vol. 43. Wiley Online Library, e15243.

Aryamaan Jain, Avinash Sharma, and KS Rajan. 2022. Adaptive & multi-resolution procedural infinite terrain generation with diffusion models and Perlin noise. In *Proceedings of the Thirteenth Indian Conference on Computer Vision, Graphics and Image Processing*. 1–9.

Aryamaan Jain, Avinash Sharma, and KS Rajan. 2024c. Learning based infinite terrain generation with level of detailing. In *2024 International Conference on 3D Vision (3DV)*. IEEE, 1048–1058.

Sebastian Janampa and Marios Pattichis. 2024. SOFI: Multi-Scale Deformable Transformer for Camera Calibration with Enhanced Line Queries. In *35th British Machine Vision Conference 2025, BMVC 2025*.

Jarosław Jasiewicz and Tomasz F Stepinski. 2013. Geomorphons—a pattern recognition approach to classification and mapping of landforms. *Geomorphology* 182 (2013), 147–156.

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

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems* 35 (2022), 22199–22213.

Ashish A Kubade, Avinash Sharma, and K S Rajan. 2020. Feedback Neural Network Based Super-Resolution of DEM for Generating High Fidelity Features. In *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*. 1671–1674.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. 2025. FLUX.1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. arXiv:2506.15742

Sijin Li, Ke Li, Liyang Xiong, and Guoan Tang. 2022. Generating Terrain Data for Geomorphological Analysis by Integrating Topographical Features and Conditional Generative Adversarial Networks. *Remote Sensing* 14, 5 (2022), 1166.

Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. 2025. Craftsman3d: High-fidelity mesh generation with 3d native diffusion and interactive geometry refiner. In *Proceedings of the Computer Vision and Pattern Recognition Conference*.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. 2023a. Zero-1-to-3: Zero-shot One Image to 3D Object. arXiv:2303.11328

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023b. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv:2303.05499* (2023).

J. Lochner, J. Gain, S. Perche, A. Peytavie, E. Galin, and E. Guérin. 2023. Interactive Authoring of Terrain using Diffusion Models. *Computer Graphics Forum* 42, 7 (2023), e14941. doi:10.1111/cgf.14941

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv:1711.05101* (2017).

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11461–11471.

Lawrence W Martz and Jurgen Garbrecht. 1998. The treatment of flat areas and depressions in automated drainage analysis of raster digital elevation models. *Hydrological processes* 12, 6 (1998), 843–855.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. 2021. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv:2108.01073* (2021).

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4460–4470.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2021. NeRF: representing scenes as neural radiance fields for view synthesis. 65, 1 (2021), 99–106. doi:10.1145/3503250

Ian Donald Moore, RB Grayson, and AR Ladson. 1991. Digital terrain modelling: a review of hydrological, geomorphological, and biological applications. *Hydrological processes* 5, 1 (1991), 3–30.

Shanthika Naik, Aryamaan Jain, Avinash Sharma, and KS Rajan. 2022. Deep generative framework for interactive 3d terrain authoring and manipulation. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 6410–6413.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv:2304.07193* (2023).

Simon Perche, Adrien Peytavie, Bedrich Benes, Eric Galin, and Eric Guérin. 2023. Authoring Terrains with Spatialised Style. *Computer Graphics Forum* 42, 7 (2023), e14936. doi:10.1111/cgf.14936

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv:2209.14988* (2022).

Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. 2023. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv:2306.17843* (2023).

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. 234–241.

Joshua J. Scott and Neil A. Dodgson. 2021. Example-based terrain synthesis with pit removal. *Computers & Graphics* 99 (2021), 43–53.

Ryan John Spick and James Alfred Walker. 2019. Realistic and textured terrain generation using GANs. In *European Conference on Visual Media Production*. 1–10.

Haruka Takahashi, Yoshihiro Kanamori, and Yuki Endo. 2022. 3D terrain estimation from a single landscape image. *Computer Animation and Virtual Worlds* 33, 6 (2022), e2119. doi:10.1002/cav.2119

Flora Ponjou Tasse, Arnaud Emilien, Marie-Paule Cani, Stefanie Hahmann, and Adrien Bernhardt. 2014a. First person sketch-based terrain editing. In *Graphics Interface 2014*. 217–224.

Flora Ponjou Tasse, Arnaud Emilien, Marie-Paule Cani, Stefanie Hahmann, and Neil Dodgson. 2014b. Feature-based terrain editing from complex sketches. *Computers & Graphics* 45 (2014), 101–115. doi:10.1016/j.cag.2014.09.001

Tencent. 2025. Hunyuan3D 2.1: From Images to High-Fidelity 3D Assets with Production-Ready PBR Material. arXiv:2506.15442

James T Todd. 2004. The visual perception of 3D shape. *Trends in cognitive sciences* 8, 3 (2004), 115–121.

U.S. Geological Survey. 2024. 3D Elevation Program (3DEP). Accessed: 2024-11-11. 1-meter resolution Digital Elevation Model tiles.

Luis Oswaldo Valencia-Rosado, Zobeida J Guzman-Zavaleta, and Oleg Starostenko. 2020. Generation of Synthetic Elevation Models and Realistic Surface Images of River Deltas and Coastal Terrains Using cGANs. *IEEE Access* 9 (2020), 2975–2985.

Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. 2018. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European conference on computer vision (ECCV)*. 52–67.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. 2025. MoGe-2: Accurate Monocular Geometry with Metric Scale and Sharp Details. *arXiv:2507.02546* (2025).

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv:2203.11171* (2022).

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.

M.J. Westoby, J. Brasington, N.F. Glasser, M.J. Hambrey, and J.M. Reynolds. 2012. 'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology* 179 (2012). doi:10.1016/j.geomorph.2012.08.021

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4578–4587.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. 2024. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)* 43, 4 (2024).

Ruo Zhang, Ping-Sing Tsai, J.E. Cryer, and M. Shah. 1999. Shape-from-shading: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, 8 (1999), 690–706. doi:10.1109/34.784284

Yiwei Zhao, Han Liu, Igor Borovikov, Ahmad Beirami, Maziar Sanjabi, and Kazi Zaman. 2019. Multi-Theme Generative Adversarial Terrain Amplification. *ACM Transactions on Graphics* 38, 6, Article 200 (2019).

Howard Zhou, Jie Sun, Greg Turk, and James M. Rehg. 2007. Terrain Synthesis from Digital Elevation Models. *IEEE Transactions on Visualization and Computer Graphics* 13, 4 (2007), 834–848. doi:10.1109/TVCG.2007.1027