

Frequency-Based Creation and Editing of Virtual Terrain

Gwyneth A. Bradbury^{*}
University College London

Il Choi
University College London

Cristina Amati[†]
University College London

Kenny Mitchell[‡]
Disney Research

Tim Weyrich[§]
University College London

ABSTRACT

Authoring virtual terrains can be a challenging task. Procedural and stochastic methods for automated terrain generation produce plausible results but lack intuitive control of the terrain features, while data driven methods offer more creative control at the cost of a limited feature set, higher storage requirements and blending artefacts. Moreover, artists often prefer a workflow involving varied reference material such as photographs, concept art, elevation maps and satellite images, for the incorporation of which there is little support from commercial content-creation tools. We present a sketch-based toolset for asset-guided creation and intuitive editing of virtual terrains, allowing the manipulation of both elevation maps and 3D meshes, and exploiting a layer-based interface. We employ a frequency-band subdivision of elevation maps to allow using the appropriate editing tool for each level of detail. Using our system, we show that a user can start from various input types: storyboard sketches, photographs or height maps to easily develop and customise a virtual terrain.

Categories and Subject Descriptors

I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism; I.3.8 [Computer Graphics]: Applications.

General Terms

Computer Graphics

Keywords

Terrain, modelling, first-person, silhouette, reference-based

^{*}g.bradbury@cs.ucl.ac.uk

[†]c.amati@cs.ucl.ac.uk

[‡]kenny.mitchell@disneyresearch.com

[§]t.weyrich@cs.ucl.ac.uk

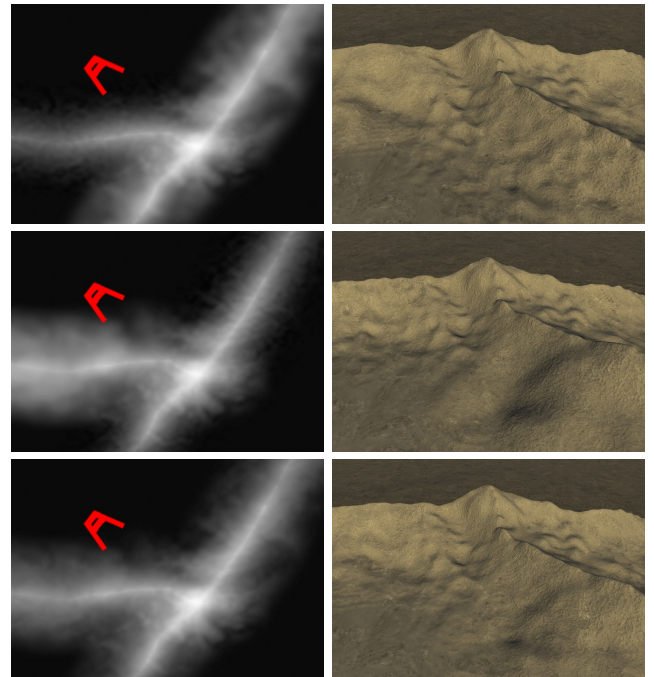


Figure 1: A user adds separate scales of fractal detail to two sketched ridges using our toolset. From top: high frequency noise applied to the left ridge, low frequency to the right; low frequency noise applied to the left ridge, high frequency to the right; lower levels of medium frequency noise applied to both features. Red marker indicates camera position.

1. INTRODUCTION

Applications requiring virtual terrain include films, animation, games, landscape visualisation and flight simulations. Modelling terrain is a challenging task due to the characteristics of natural landscapes. At a macro level, terrains present distinctive geological features such as height, shape, and distribution of roughness, but at micro level they can become increasingly stochastic. This makes manual modelling by simply extruding elevation meshes a difficult task.

Existing solutions for terrain generation and editing can be condensed into 2D methods and 3D methods. 2D methods rely on the representation of terrain as a height map, essentially an image. Thus, common image processing operations such as texture synthesis, blending and compositing can be applied to create a height map that corresponds to user spec-



Figure 2: *Example reference collection for constructing a desert terrain, including photographs and digital elevation map. (Moodboard drawn from Pure, courtesy of Disney.)*

ifications. The shortcomings of these methods consist of: the limitation of terrain features to the ones present in the input height maps (if the method is data-driven); speed and computation time; high storage requirements, and difficulty in editing once the new terrain map has been generated.

3D methods can be procedurally-based or manual. Procedural methods rely on different types of noise to create rich and plausible results, but are difficult to control at a feature level, and any modification requires the whole terrain to be re-generated from scratch. Interactive methods can be found in software packages such as Mudbox [Aut14b], ZBrush [Pix14] or Maya [Aut14a] which offer the user complete freedom to sculpt a mesh to their desire. However, this is a lengthy and daunting process and can often result in implausible terrains due to lack of high frequency detail.

In practice, artists resort to a manual, example-based approach where inspiration and references are collected from the real world in various formats. These can include photography, elevation maps, satellite images or concept sketches - both digital or hand drawn. Figure 2 shows a moodboard used for video game production which gathers all these resources. It is important to note that the majority of resources are in fact 2D as the acquisition of 3D material from nature is non-trivial. Maclean [Mac11] describes traditional artistic workflows for collaborative scene and animation development which depend heavily on sketched scenes. Without mixed-media methods such as ours, which allow 2D reference materials and sketches to be converted into 3D assets, these esteemed artistic workflows are at risk of being lost.

To meet the need of artists to work with multiple input types and also to offer great editing flexibility, we have created a terrain generation and editing framework that seamlessly combines 2D and 3D workflows. Users can start either from 2D artwork, elevation maps or randomly generated fractal terrain. The elevation map and the corresponding 3D mesh are both editable using various sets of operations detailed in Section 4 and offer immediate visual feedback.

Different tools are best applied at different scales: creation of large-scale features can be fully artist-directed using interactive 3D methods, whereas creation of small-scale detail is a tedious task when carried out by hand and is best dealt with by noise generation. We provide the separation into high-, mid- and low-frequency bands of the terrain and allow individual editing of each frequency band.

We borrow from image editing paradigms the concept of layers and apply each new heightfield editing operation in a distinct layer. We extend this layer view to include three sub-layers which show, respectively, the high-, medium- and low-frequency bands of the image. This offers additional flexibility in handling content.

Our contributions to terrain generation and editing include: extending the concept of layered image editing by incorporating frequency-controlled sub-layers; support for multiple input types to assist in initial terrain creation and creative guidance, and a flexible workflow with multiple tools applicable both to elevation maps and terrain meshes.

Our prototype system offers a set of representative operators that combine state-of-the-art techniques in virtual terrain manipulation.

2. RELATED WORK

Virtual terrain creation has recently received more attention in the literature. Until lately, procedural techniques had become the default method for terrain generation, allowing a designer to generate vast, or even infinite, landscapes with just a few parameters. These approaches, however, commonly suffer a lack of intuitive control over the output as well as low level of support for editing post-generation.

Whilst procedural and noise-based approaches still have significant application in adding high-frequency (small scale) detail to terrain, new techniques tend towards sketch-based or data-driven methods as well as hybrid approaches. Clone and stamp tools are also gaining in popularity, allowing the designer to intuitively move or stamp certain features onto a terrain. Sketching, similarly, is a highly intuitive means of content creation either in the image domain (the 2D heightfield, in our case) or directly in 3D, as in many 3D modelling software packages such as ZBrush [Pix14], Mudbox [Aut14b], or Maya [Aut14a].

Creating plausible landscapes with sketch and sculpt tools alone can be a demanding task, especially at small scales. This is overcome using data-driven methods which provide accurate replication of features from real-world examples.

Procedural Terrain Generation

Methods exploiting deterministic functions such as fractal noise, Brownian Motion [EVB10] or Perlin noise [Per85], are capable of generating complex, highly detailed terrain on the fly. An alternative to these random methods are mid-point displacement and the diamond square fractal generation algorithms [FFC82], which are seed-based. Other rule-based systems, driven by hydrology [GGG13] and erosion [PM13] models, make significant progress in imitating systems in the natural world but are computationally complex.

Genevaux et. al. [GGG13] present an intuitive sketch-based interface for modelling terrain via hydrology concepts. The user starts with a simple sketch but thereafter must tune parameters for final terrain generation according to preference. Whilst the algorithm is efficient, allowing for real-time use, and produces plausible terrain, parameter control is still high level and the user lacks fine control over the result.

Although many procedural algorithms suffer from unintuitive, parametric control, with little visual correlation between the input values and the output terrain, these methods remain a good starting point for terrain modelling and are exploited in commercial tools such as Terragen [Pla14] and Vue [E-o14]. Smelik et al. [SDKT*09] give a thorough overview of procedural terrain modelling approaches.

Example-Based Terrain Generation

Parberry [Par13] presents an approach to iteratively adapt procedurally generated heightfields to resemble a real-world digital elevation map using histogram matching. This is fast and efficient but does account for the variety of localised terrain features. One further data-driven method is texture synthesis [EL99, EF01, LH05], which is appropriate for content creation and editing of 2D elevation maps.

Example-based texture synthesis solutions exploiting a sketch-based user interface for guidance are also becoming more prominent [ZTR07, TGM12, dPI13]. Texture synthesis is commonly guided by a user sketch depicting where certain features should occur and patches are merged using blending in the gradient domain [PGB03]. This allows the user to focus on low frequency (large scale) features of terrain with finer details obtained from the input patches.

Patch-based methods rely on having the required output sufficiently well represented by the input data. If a feature does not exist in the source data, it cannot be reproduced in the target terrain. High complexity and storage requirements (see [TGM12, BMV*11]) can also be problematic, as well as the need to perform a correspondence search in a high dimensional space.

Sketch-based Landscape Manipulation

An intuitive way to add new landscape features is by sketching. This can be done either in the image domain on the elevation maps or directly in 3D, as in many 3D modelling software packages such as ZBrush [Pix14], Mudbox [Aut14b], or Maya [Aut14a].

Belhadj and Audibert [BA05] model elevation maps based on sketched ridge and river networks. There, ridges are created by modelling a Gaussian cross-section over a user-sketched baseline. The base of the ridge is considered to be flat on the ground plane. Gain et al. [GMS09] show a more detailed user interaction for ridge sketching, based on parametrised curves drawn onto a base plane. The user draws the ridge profile while the system approximates the base with an editable ellipse and then generates geological features to fit the silhouette and boundary.

Other sketch-based methods have also been developed (see [BKST10, Sme11]), allowing sketching of the overall terrain

constraints which is then filled with procedural geometry. While these approaches yield impressive results, once the geometry is generated there are no more possibilities of further editing the output. Furthermore, these are constraint-based approaches and do not support fine-grain user interaction.

Many sketch-based solutions are limited by the unintuitiveness of the user interface. Passos and Igarashi [dPI13] draw attention to this, stating that an artist (either naive or experienced) will have a natural tendency to draw the silhouettes of the landscape, as if painting it from a first-person perspective point of view, rather than the typical birds-eye view many algorithms require. Their algorithm is limited by the quality and scope of input data but gives the user a much more intuitive control over the final terrain. Tasse et. al. [TEC*14] also present a first-person, sketch-based interface which allows easy personalisation of a landscape by deforming the terrain to match the user's sketch exactly.

Our ridge creation tool, described in Section 4.2, is strongly inspired by these sketch-based interfaces, and incorporates further flexibility by allowing the user to set the perspective mapping of the sketch canvas. This allows better support for input references which may use different, and potentially unrealistic, perspectives.

We see this introduced capability as an important bridge to recovering traditional artistic workflows which employ direct transfer of 'flats' or 'plates' to layered scene depiction with studied depth and composition [Mac11], from which we provide a path to fully generated 3D terrain relief edits.

Compositing and Editing

Typical texture synthesis approaches are designed to work well in image applications but require special treatment for terrain generation as they tend to leave seams along patch boundaries. This problem of seamless blending is most poignant in the gradient domain. Dos Passos and Igarashi [dPI13], for example, apply a weighted sum of heights across patch boundaries. This type of method produces artefacts in the gradient domain which are only noticeable as the viewer moves through the scene [TGM12, PGB03, ZTR07].

Common solutions for seamless blending are Poisson image editing [PGB03] and Shepard Interpolation [She68], but they have proven to be too slow for real-time applications [TGM12, ZTR07]. Our work employs Laplacian blending [BEA83] which is faster and ensures continuity across seams.

Gradient blending is also useful when features overlap, which is likely to happen when the user sketches edits onto an existing terrain map. Vemuri et al. [VML97] blend new features into the original terrain by treating them as spline surfaces and applying a fractal to the surface to add terrain-like noise.

Jenny et. al. [BHL11] present a means of attenuating and amplifying certain spatial frequencies of a terrain using a Laplacian pyramid. Their slider-based interface is used to clean cartographic data for visualisation and generalises terrain whilst maintaining its characteristics. The tool targets only accentuation of geographical features and noise reduction rather than creative control.

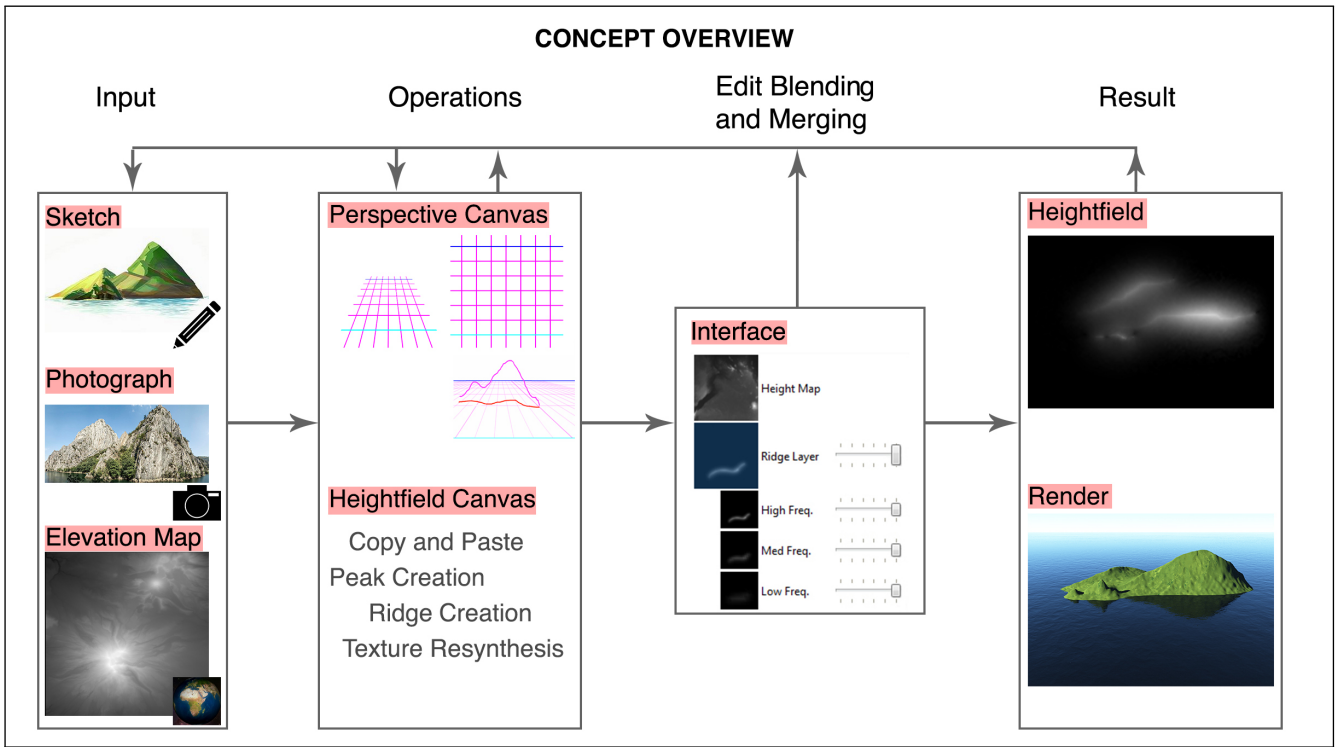


Figure 3: Overview of system workflow. From left: optional inputs (the user may use any combination or none at all); editing operators; edit layer merging interface; resulting heightfield and rendered view. The user has the option at any point to add further input resources, edits and make adjustments to blending parameters.

3. FRAMEWORK AND INTERFACE

Our proposed system consists of sketch-based terrain creation and editing tools with various degrees of automation and manual control which, in combination, facilitate a workflow for custom terrain creation.

The interface to our system provides the user with aerial and perspective views of the developing terrain. Concurrent views update in real time during each edit. The aerial view consists of the 2D elevation map of the terrain and the perspective view shows the corresponding 3D mesh.

Using a 2D elevation map representation for terrain allows our tools to exploit common image processing operations (e.g. cut, copy and paste), as well as layer-based editing.

The layers panel shows the original heightfield as the base layer and each new modification as a sub-layer (see Figure 3). Layers can then be merged or deleted, allowing for a non-destructive editing pipeline as seen commonly in image editing software such as Adobe’s Photoshop [Ado14] and the GNU Image Manipulation Program [The14]. This is further described in Section 3.2.

Rather than combining layers using alpha-blending, our system features gradient-domain blending, applied in real-time. This reduces artefacts in the gradient domain which become apparent in 3D. Further to the simple layer view described above, the user is presented with 3 levels from the Laplacian pyramid decomposition of each edit layer, and is given sliders to control the effect (similar to an alpha value) of each

frequency band. The Laplacian pyramid representation used for merging new features and modifications offers real-time reconstruction, even as the user changes the blending parameters. Combined, these features help to fulfil the need for interactivity.

3.1 Incorporating Assets

Our toolset allows the user to follow a flexible workflow, importing new assets at any point, rather than restricting them to a fixed pipeline (see Figure 3 for an overview).

Users have a choice of starting points for creating terrain - blank canvas, existing elevation maps, photographs or concept art. When starting out from scratch, the framework offers the options of having a random elevation map generated using standard procedural methods. Another option is to import reference material such as photographs or sketches as direct input to the canvas and use a natural first-person perspective view to draw terrain silhouettes directly onto the reference material, which will then be translated into an elevation map and a corresponding 3D mesh. Each feature is created in a new editing layer and the user can, at any point, control the blending of these layers on the canvas (Section 3.2).

New reference material can be brought in at any time during the creation process and integrated with the already existing terrain. External elevation maps can be used for adding geological features and high frequency detail; new ridges and peaks can be sketched directly onto the existing heightmaps and typical image processing operations such as cut, copy,

and paste can help shape the desired terrain (see Section 4 for details).

3.2 Laplacian Blending and Editing

Creating a separate layer in the terrain canvas for each modification and decomposing each layer into a Laplacian pyramid allows us to apply gradient domain blending and achieve seamless merging of these modifications with the underlying terrain. Simultaneously, by displaying the Laplacian pyramid to the user in an intuitive form, we can give selective control over the strength of each frequency band’s contribution within its layer. See Figure 3. Each of these frequency bands are presented to the user as sub-layers, with a slider to control blending upon recomposition. Using Laplacian image blending to combine edit layers in this way avoids artefacts in the gradient domain [TGM12].

To create the decomposition, the underlying heightfield and each edit layer are three times successively filtered and subtracted from the original layer. When recomposing this heightfield, each band in each layer is given an alpha value which controls its effect on the result. We direct the reader to [AAB*84] and [PGB03] for a more detailed explanation of Laplacian image decomposition.

We separate the image into just three frequency bands as, in our experience, separating the image into more bands gives no additional advantage to the user – it becomes difficult to see the impact of narrower bands on the landscape, and the increased complexity adds confusion and slows down the workflow.

4. REPRESENTATIVE TOOLBOX

Our system also includes typical image processing tools such as cut, copy and paste as well as more complex tools: a first-person sketching canvas for tracing features from a range of reference types; texture synthesis for adding detail, and peak and ridge creation. Some of these are based on existing techniques and outlined below. Figure 3 puts these tools in the context of our system workflow.

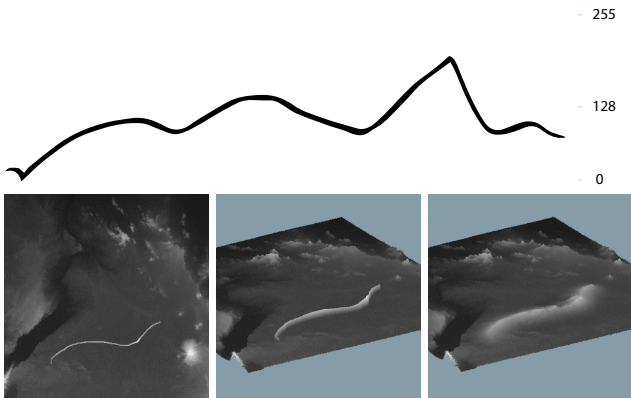


Figure 4: *Ridge creation interface: bottom left, user sketches the base of the new ridge onto the heightfield; top, user sets the profile of the new ridge; bottom centre, before the new ridge base profile is created; bottom right, the new ridge blended into the canvas.*

4.1 Silhouette Sketching from Concept Art

In practice, reference material can come from a host of different sources. Scouts bring back photographic references using different camera set-ups whilst other artists create sketches and virtual mock-ups with a loose (and sometimes unrealistic) interpretation of perspective. Merging the different perspective transformations captured by these sources then becomes non-trivial using a sketch-based interface such as [dPI13].

A further contribution of our system is a means of mitigating the effect of these different perspective settings by allowing the user control over them in the sketching canvas. We provide a first-person perspective canvas, allowing the user to sketch ridge features and see them appear in 3D on the terrain.

In order to deal with a wealth of reference material, we give the user intuitive control over the perspective mapping between first-person canvas and terrain. To aid in setting up the perspective canvas, a grid is overlaid onto both the perspective canvas and the corresponding 2D heightfield canvas, indicating the perspective mapping of the virtual ground plain. The user then adjusts this mapping by dragging the horizon line and outer grid lines until the grid’s perspective corresponds with their interpretation of the image (Figure 3: ‘Perspective Canvas’).

When terrain is sketched in the 3D perspective environment, the new feature’s position is immediately transformed onto the heightfield simply by solving the homography between the adjusted base plane and the 2D heightfield. Its elevation is inversely proportional to the depth of its corresponding position on the ground plane.

4.2 Peak and Ridge Creation

We allow the user to add peaks on top of the existing terrain. Once the base has been outlined, the height of a desired peak is set and a Gaussian-shaped peak of respective height and standard deviation is added to the terrain. This peak can then be further modified, by interactively changing skewness and kurtosis, and by tracing its profile in a cross-sectional view. Finally, it is blended with the surrounding environment using a Laplacian pyramid

As an alternative means of ridge creation, we allow the user to first sketch a ridge line directly onto the heightfield and then sketch the height profile (Figure 4). The new ridge feature is dynamically placed onto the terrain as it is created. The resulting features create a new layer and, by default, affect only the low-frequency content of the terrain. Higher frequencies are taken from the original terrain.

4.3 Texture Transfer

Texture transfer is a broad term encompassing the way our system deals with copy and paste operations. When copying and pasting terrain (either from the current canvas or from an external source), we use gradient blending to merge the two areas, taking the highest gradient pixel from either region. As a result, the output terrain preserves characteristics from both the source and destination and is merged seamlessly. By using the frequency decomposition in the layer view, the user can adjust the effect the pasted region

has on the existing terrain, choosing to discard details at high, medium or low frequency as desired.

This feature is important for adding detail to terrain features created by sketching. The sketched feature corresponds to low frequency features in the terrain but in order for the terrain to appear plausible (and not smooth), high frequency detail must somehow be added. Our tool allows the user to choose detail they like from other sources and incorporate this into the scene. Figure 5 shows an example of pasting detail onto a new feature. Figure 6 shows what happens when low frequencies in the pasted region are discarded.

4.4 Texture Synthesis

We use a modified texture synthesis algorithm for hole filling when a region is cut from the terrain which is based on image quilting. For this purpose, we must not only consider constraints on neighbouring pixels but also enforce initial boundary constraints. Traditional scan-line approaches leave artefacts at the boundaries where residual error from patch matching accumulates. Ideally, this error should be evenly distributed across the synthesised region and, therefore we employ a spiralling technique to achieve this goal.

In informal requirements discussion with artists, we received the request that the system should offer multiple versions of auto-generated terrain, so the artist can pick a suitable instance. This is in line with the need for more variation for enforcing naturalness and plausibility of the terrain. Accordingly, we synthesise the new region three times, and let the user choose their preferred result (see Figure 7). We then allow the user to again resynthesise the region again, using the previously preferred result as a starting point, preserving its structure but providing further variations. The process can be repeated or restarted as desired and the result is then blended into the original heightmap. Figure 7 shows an example result.

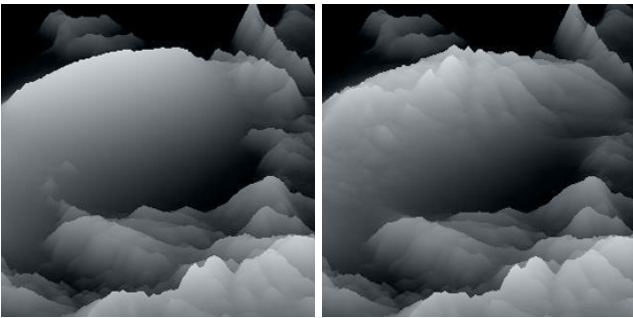


Figure 5: *Adding texture to a new, initially smooth, sketched feature: before (left) and after (right).*

5. RESULTS

Our results demonstrate how the user may successfully turn concept artwork including photographs and digital sketches into virtual terrain. The resulting terrain geometry was created using our system and rendered using Planetside’s Terragen 3 [Pla14].

Frequency band decomposition in the layer view of the scene proves useful for the user for transfer of high frequency detail from an external source onto the terrain. Results using this

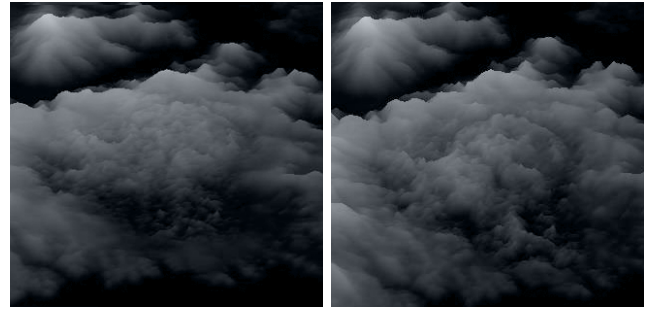


Figure 6: *Changing the frequency of detail transferred. Left, only high frequency detail is transferred; right, all detail transferred.*

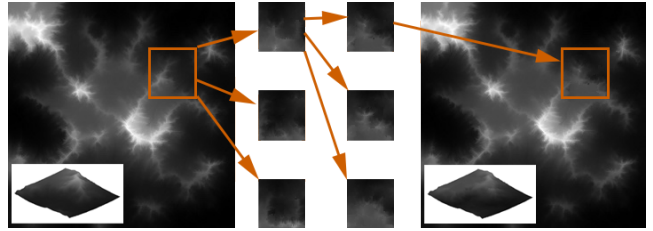


Figure 7: *Using texture synthesis for hole filling. Left, the original terrain, a patch is selected for resynthesis based on the rest of the terrain, center, the user chooses one of three possible results, which is again resynthesised. The final choice is merged into the terrain.*

tool can be seen in Figures 12 and 11, demonstrating the change in terrain as the user adjusts the effect of each of two of our three frequency bands.

Figure 9 (a,b) uses a photograph as input to the system. The user gradually builds up terrain features by setting the perceived perspective of the input and sketching onto the perspective canvas. Figures 9 (c,d,e) demonstrate a similar results, using a digital sketch as input. Figure 10 demonstrates how different viewpoints can be combined using the perspective sketching interface. The user sketches the first ridge (top left in the figure), rotates the aerial canvas, and draws a second (top, centre), intersecting ridge. High frequency detail is added to the resulting terrain (top right). Figure 8 further demonstrates the use of our supporting tools. Features are created on the aerial canvas as well as being copied from external terrain - elevation maps of both the Grand Canyon and Mount Everest are used to provide features. Blending is seamless for all editing operations.

6. CONCLUSION

Results show that our system allows virtual terrain to be designed from concept art using a non-rigid work-flow which incorporates reversible, unordered, and non-destructive editing phases. The resulting heightfields can be edited using a sketch-based interface. We demonstrate a layering system which allows tuning of coarse to fine levels of detail without inhibiting creative direction. Frequency-selective control over the merging of terrain edits allows to tune individual operators after their application.

Using our system, access to these levels of detail is straightforward and displayed to the user in three frequency bands.

The user can mix these bands and apply different edits to each, as desired. To carry out a similar edits in commercial editing tools is a non-trivial process. There is no direct access to frequency bands of an image and so the user would need to manually convolve and subtract image layers.

Our tools support additional reference material which can be incorporated at any point. We handle perspective reference material (e.g. photographs or concept artwork) by allowing the user to set a perspective mapping using a simple guide mesh that maps new edits on the perspective canvas directly onto the terrain's heightfield. This is an important feature, particularly for transforming concept art, which may use unrealistic perspective mapping, into usable terrain.

Our system addresses the need to guide terrain editing with digital assets (terrain maps, photographs, and sketches and concept art), linking conceptual art to terrain generation.

There are several directions in which our work can be extended. Colours could be adapted from the input artwork and projected onto the new terrain. Whilst this would be trivial for a single image, the user may use different concept art at different points in the scene which would need to be blended or merged in an intelligent way. Automatic, or semi-automatic, extraction of the base layer elevation would make the system more usable - currently the system assumes the same base layer elevation for all input but this could vary depending on the scene. Finally, whilst our system is based directly on artist requirements, it would benefit significantly from wider user testing.

User feedback has been encouraging, and showed that such a tool could be useful in the creative pipeline. However, testing has been limited and would benefit from deeper investigation in future work.

Acknowledgements

This work was supported by the UCL EngD VEIV Centre for Doctoral Training.

7. REFERENCES

- [AAB*84] ADELSON E. H., ANDERSON C. H., BERGEN J. R., BURT P. J., OGDEN J. M.: Pyramid methods in image processing. *RCA Engineer* 29, 6 (1984), 33–41.
- [Ado14] ADOBE: Photoshop. www.photoshop.com/, July 2014.
- [Aut14a] AUTODESK: Maya. www.autodesk.co.uk/products/autodesk-maya/overview, July 2014.
- [Aut14b] AUTODESK: Mudbox. www.autodesk.com/products/mudbox/overview, July 2014.
- [BA05] BELHADJ F., AUDIBERT P.: Modeling landscapes with ridges and rivers. In *VRST* (2005), Singh G., Lau R. W. H., Chrysanthou Y., Darken R. P., (Eds.), ACM, pp. 151–154.
- [BEA83] BURT P. J., EDWARD, ADELSON E. H.: The laplacian pyramid as a compact image code. *IEEE Transactions on Communications* 31 (1983), 532–540.
- [BHL11] BERNHARD J., HELEN J., LORENZ H.: Terrain

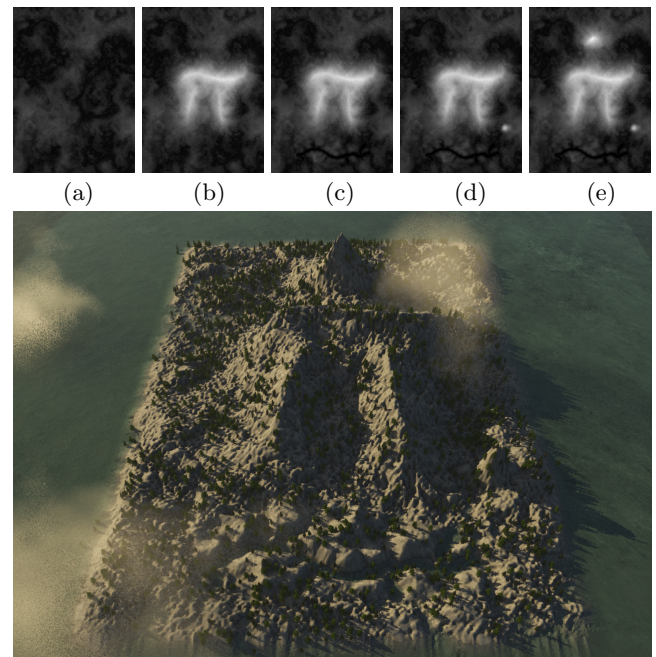


Figure 8: Workflow as the user loads a procedurally generated (fractal) heightfield (a), sketches ridges and peaks (b), and adds valley and peak features from the real world digital elevation maps (c, d, e). All edit operations use our supporting tools.

generalization with multi-scale pyramids constrained by curvature. *Cartography and Geographic Information Science* 38, 1 (2011), 110–116.

- [BKST10] BIDARRA R., KRAKER K. J. D., SMELIK R. M., TUTENEL T.: Integrating semantics and procedural generation: key enabling factors for declarative modeling of virtual worlds. In *Proceedings of FOCUS K3D Conference on Semantic 3D Media and Content* (Sophia Antipolis, France, feb 2010), pp. 51–55.
- [BMV*11] BERNHARDT A., MAXIMO A., VELHO L., HNAIDI H., CANI M.-P.: Real-Time Terrain Modeling Using CPU-GPU Coupled Computation. In *SIBGRAPI* (2011), Lewiner T., da Silva Torres R., (Eds.), IEEE, pp. 64–71.
- [dPI13] DOS PASSOS V. A., IGARASHI T.: Landsketch: a first person point-of-view example-based terrain modeling approach. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling* (2013), ACM, pp. 61–68.
- [E-o14] E-ON SOFTWARE: Vue. www.e-onsoftware.com/products/vue/, July 2014.
- [EF01] EFROS A. A., FREEMAN W. T.: Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*

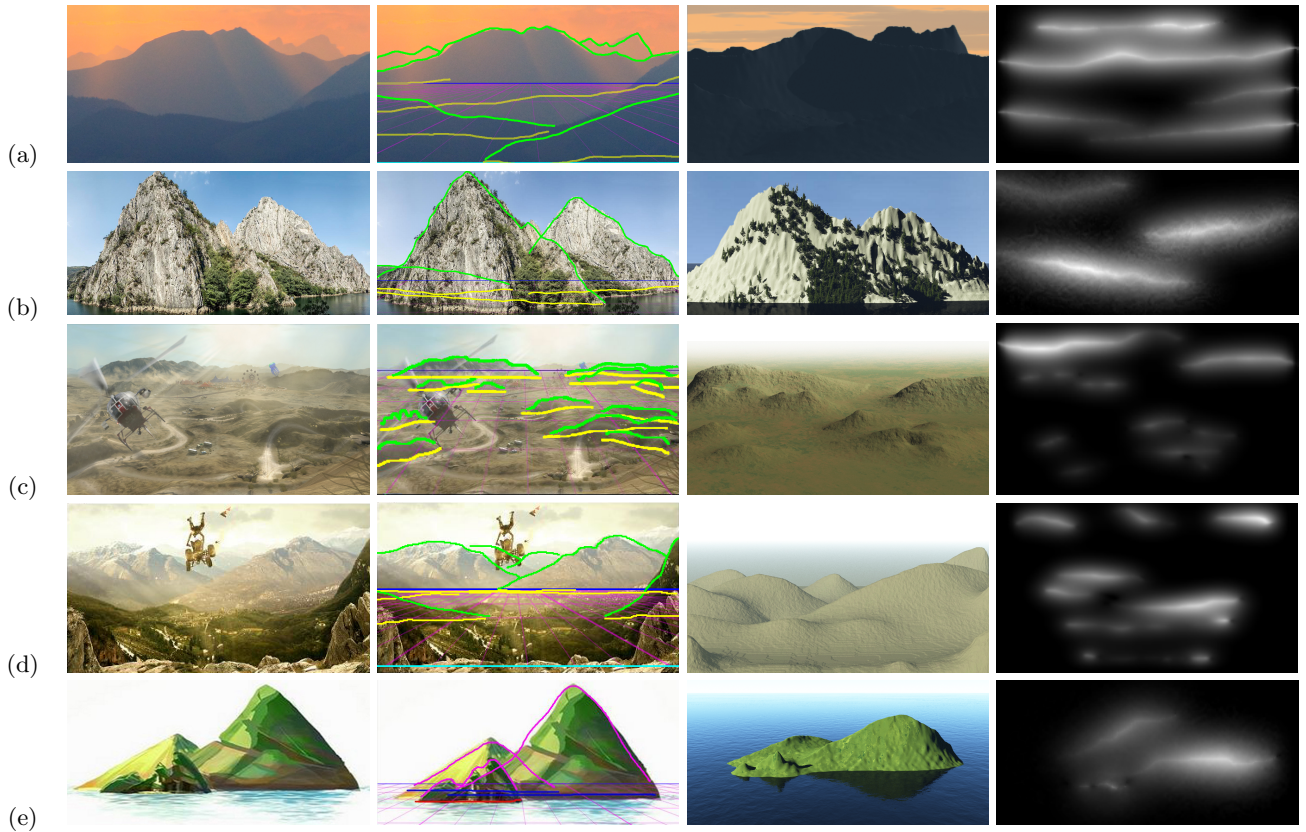


Figure 9: *Example output from our system. The user starts with (a, b) a photograph, or 2D concept art (c, d) and adds noise using our frequency sub-layer interface. From left to right: input image, adding perspective to the canvas and sketching ridges, the generated terrain, and the corresponding heightfield. In example (d), virtual trees are added to the generated terrain for rendering. (Pure in game screenshots (c,d), courtesy of Disney).*

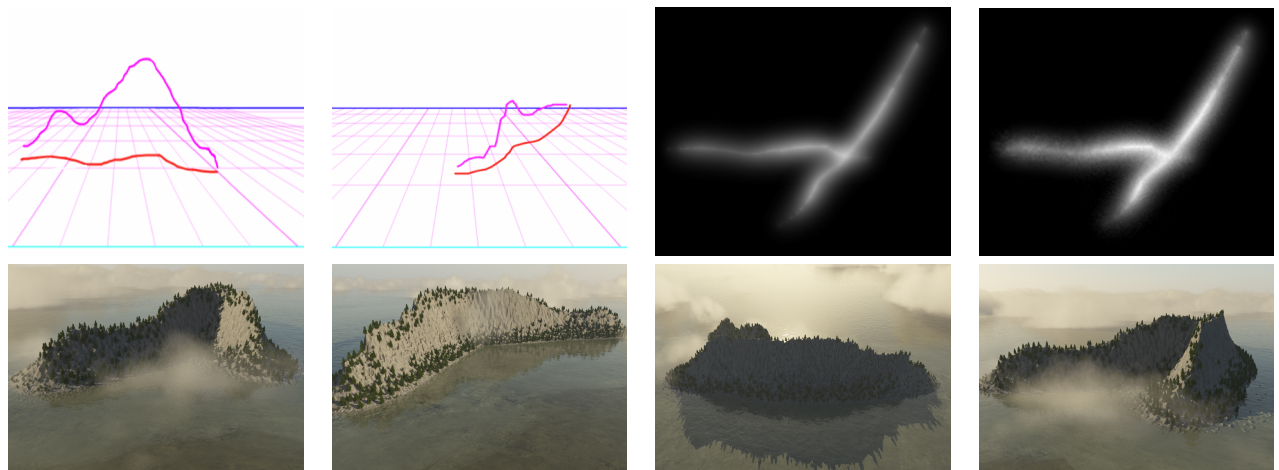


Figure 10: *Example output from our system. Top: the user starts with a blank perspective canvas and sketches from two different viewpoints (left, centre left), resulting heightfield (centre right). The user then transfers detail from an external elevation map, resulting heightfield far right. Bottom: the final scene, rendered from four surrounding viewpoints. Rendered with added vegetation models and atmospheric effects.*

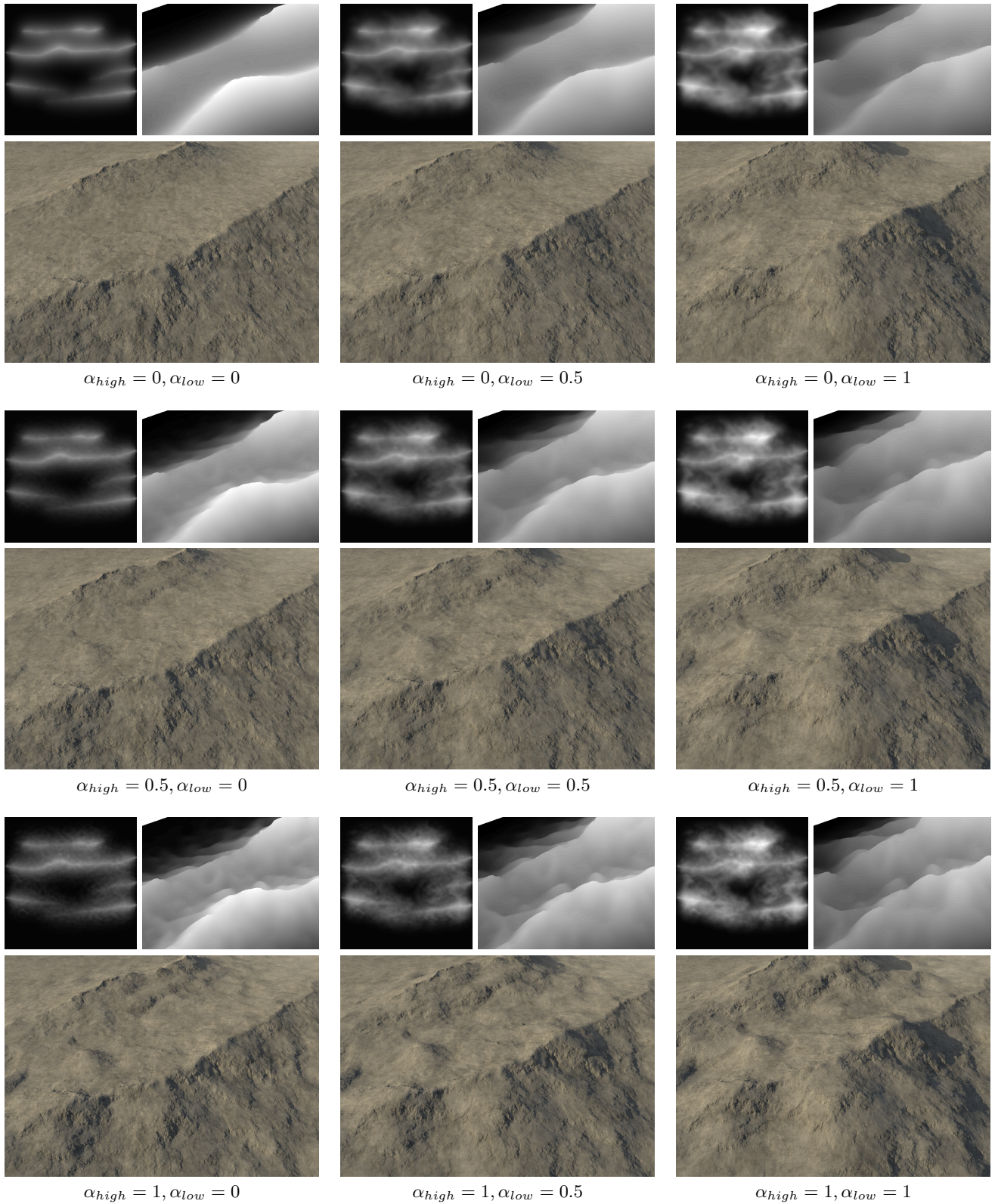
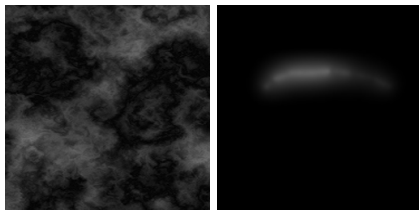
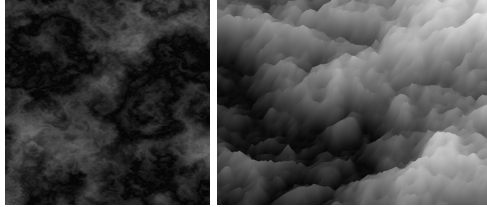


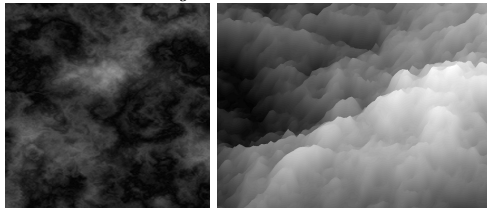
Figure 12: Adding fractal detail to a heightfield output from our perspective sketching tool (Figure 9a). Horizontally, effect as the ‘low frequency’ slider is incremented. Vertically, effect as the ‘high frequency’ slider is incremented. For each, we show the elevation map (left) and 3D interface view (right). For each, we show the resulting elevation map, 3D view from our interface and the same view rendered using Terragen 3 [Pla14].



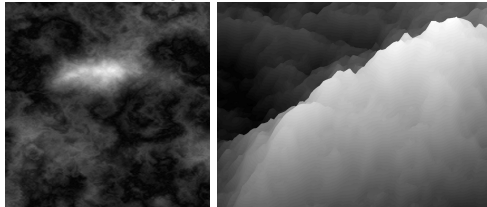
Initial heightfield (left) and added ridge feature (right)



$\alpha_{high} = 0, \alpha_{low} = 0$



$\alpha_{high} = 0, \alpha_{low} = 0.5$



$\alpha_{high} = 0, \alpha_{low} = 1$

Figure 11: Adding a new ridge (top right) to an existing heightfield (top left). Results show the effect as the 'low frequency' slider is incremented (there is no high frequency detail in the new, sketched ridge feature).

- (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 341–346.
- [EL99] EFROS A., LEUNG T.: Texture synthesis by non-parametric sampling. In *International Conference on Computer Vision* (1999), pp. 1033–1038.
- [EVB10] ECHELARD A., VÉHEL J., BARRIÈRE O.: Terrain modeling with multifractional brownian motion and self-regulating processes. *Computer Vision and Graphics* (2010), 342–351.
- [FFC82] FOURNIER A., FUSSELL D., CARPENTER L.: Computer rendering of stochastic models. *Commun. ACM* 25, 6 (June 1982), 371–384.
- [GGG13] GÉNEVAUX J., GALIN E., GUÉRIN E.: Terrain generation using procedural models based on hydrology. *ACM Transactions on Graphics* 32, July (2013).
- [GMS09] GAIN J. E., MARAIS P., STRASSER W.: Terrain sketching. In *SI3D* (2009), Haines E., McGuire M., Aliaga D. G., Oliveira M. M., Spencer S. N., (Eds.), ACM, pp. 31–38.
- [LH05] LEFEBVRE S., HOPPE H.: Parallel controllable texture synthesis. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, ACM, pp. 777–786.
- [Mac11] MACLEAN F.: *Setting the Scene*. Chronicle Books LLC, 2011.
- [Par13] PARBERRY I.: *Designer Worlds : Procedural Generation of Infinite Terrain from USGS Elevation Data*. Tech. rep., University of Northern Texas, 2013.
- [Per85] PERLIN K.: An image synthesizer. *SIGGRAPH Comput. Graph.* 19, 3 (July 1985), 287–296.
- [PGB03] PÉREZ P., GANGNET M., BLAKE A.: Poisson image editing. In *ACM SIGGRAPH 2003 Papers* (New York, NY, USA, 2003), SIGGRAPH '03, ACM, pp. 313–318.
- [Pix14] PIXLOGIC: ZBrush. www.autodesk.com/products/mudbox/overview, July 2014.
- [Pla14] PLANETSIDE: Terragen 3. www.planetside.co.uk/products/terrigen3, July 2014.
- [PM13] PYTEL A., MANN S.: Self-organized approach to modeling hydraulic erosion features. *Computers & Graphics* 37, 4 (June 2013), 280–292.
- [SDKT*09] SMELIK R. M., DE KRAKER K. J., TUTENEL T., BIDARRA R., GROENEWEGEN S. A.: A survey of procedural methods for terrain modelling. In *Proceedings of the CASA Workshop on 3D Advanced Media In Gaming And Simulation (3AMIGAS)* (2009), pp. 25–34.
- [She68] SHEPARD D.: A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference* (New York, NY, USA, 1968), ACM '68, ACM, pp. 517–524.
- [Sme11] SMELIK R. M.: *A Declarative Approach to Procedural Generation of Virtual Worlds*. PhD thesis, TU Delft, 2011.
- [TEC*14] TASSE F. P., EMILIE A., CANI M.-P., HAHMANN S., BERNHARDT A.: First person sketch-based terrain editing. In *Proceedings of the 2014 Graphics Interface Conference* (2014), Canadian Information Processing Society, pp. 217–224.
- [TGM12] TASSE F., GAIN J., MARAIS P.: Enhanced texture-based terrain synthesis on graphics hardware. *Comp. Graph. Forum* 31, 6 (sep 2012), 1959–1972.
- [The14] THE GIMP TEAM: The GNU Image Manipulation Program. www.gimp.org/, July 2014.
- [VML97] VEMURI B. C., MANDAL C., LAI S.-H.: A fast gibbs sampler for synthesizing constrained fractals. *IEEE Trans. Vis. Comput. Graph.* 3, 4 (1997), 337–351.
- [ZTR07] ZHOU H., 0004 J. S., TURK G., REHG J. M.: Terrain synthesis from digital elevation models. *IEEE Trans. Vis. Comput. Graph.* 13, 4 (2007), 834–848.