RiverGAN: Fluvial Landscapes Generation with Conditional GAN and Physical Simulations

Xun Liu
School of Architecture
University of Virginia
Charlottesville, VA 22904
`xl4xw@virginia.edu`

Runjia Tian
Graduate School of Design
Harvard University
Camrbridge, MA 02138
`runjia_tian@gsd.harvard.edu`

Abstract

Modeling river morphology is essential in the field of landscape architecture. However, most hydromorphological simulation requires sophisticated set up of physical or numerical models. With the increasing use of Neural Networks for Procedural Terrain Generation(PTG), we propose a novel approach that assembles the surrogate modeling data from a hydromorphological machine and use such data for the training of a Generative Adversarial Network (GANs) that procedurally generate fluvial landscapes without simulation.

1 Introduction

Hydromorphological Simulation Physical hydromorphology tables are used in the field of landscape architecture to simulate the fluvial performance and morphology, such as braided and meandering rivers, delta formations, tidal zones, etc [1]. However, the application of the physical simulation has limitations, such as the limited accessibility of physical devices, the time-consuming simulation process, the difficulty to determine a proper spatial and temporal scale, to prepare input data and to interpret output data. Alternatively, numerical simulations based on 2D Cellular Automata and 3D Computational Fluid Dynamics are also used in the field of landscape design [9]. However, these methods require high-accuracy of data input and intense calculation, which significantly limits the usability among designers. As for designers, the uncertainty and indeterminacy of the simulation are important in terms of understanding the complexity and dynamic process of hydromorphology. A prototyping tool which can produce a large amount of possible scenarios for design inspiration is preferred over an accurate and predictive model [17] Thus, the aim of the project is to utilize the inherent uncertainty in GAN models to study the possibilities of the process and patterns produced by the hydromorphology table, in order to inspire design creativity.

Terrain Design and Procedural Generation Terrain design is a significant part of landscape architecture where designers create terrain models in physical or digital mediums. Procedural terrain generation is a subdomain of Procedural Content Generation (PCG) [16] that plays an important role in both game design and design of built environment domains such as landscape architecture [7, 6], urban design [4] and architecture design [12]. Generation algorithms involves erosion algorithms [10], agent-based modeling [3], evolutionary algorithm [11], and more recently, Generative Adversarial Network(GAN) [13], a type of deep neural network [5]. However, existing terrain generation methods has low fidelity and are incapable of generating high-fidelity braided river models which are crucial in landscape architecture design.

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

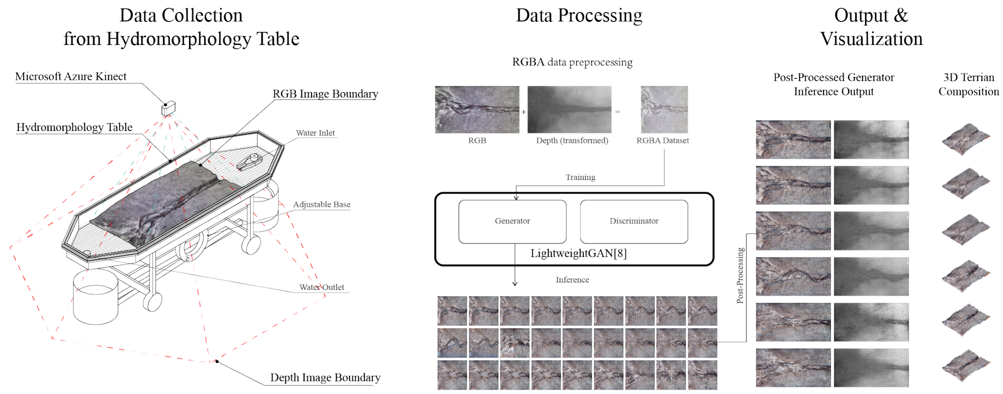

Figure 1: RiverGAN workflow of data collection, processing, training, output and visualization.

2 Methodology and Experiment

Dataset Creation We developed a novel workflow to create a braided river terrain dataset from a hydromorphology table. The table is equipped with a controllable water pump and adjustable tilted base metal bed, filled with synthetic sediment of varying colors, sizes and densities. As the simulation is running, a Microsoft Azure Kinect sensor mounted can capture real-time data of changing material composition and river shape simultaneously from top. We have run 10 times of 1-hour physical simulations with the same initial setup conditions, from which we sampled with a minimum time interval of 1-minute frame to make sure the patterns are diverse. The full training dataset consists of 502 pairs of a top-view color image with its depth image aligned to the same transform with respect to the Kinect sensor, which were further pre-processed into RGBA images where the height texture is used as alpha channel.

Training and Visualization We also used LightweightGAN, a GAN toolkit for fast, high-fidelity and few-shot image synthesis, as the surrogate model [8]. LightweightGAN uses skip-layer channel-wise excitation mechanism (SLE) and self-supervised regularization on the discriminator to boost the performance of image synthesis [8]. With the performance boost, training a GAN with state-of-the-art performance compared to StyleGAN can be achieved on most personal devices. Specifically, we train a LightweightGAN model to generate braided river terrains. We trained the generative model for 100,000 epochs and created latent walk videos from the toolkit.

For procedural content generation with deep learning, inference visualization is crucial to designers' workflow [14]. Therefore, we also developed a user interface for visualizing 3D latent walk. The toolkit can be easily integrated into the creative workflow with landscape architects and will be made open-sourced.

3 Conclusion

In this paper, we present a novel method for the procedural generation of braided river models using Generative Adversarial Network. Our method provides a faster, more accessible and intuitive way for fluvial morphodynamics modelling. Furthermore, through making analogies between the stochasticity of the GAN model and the uncertainties of the landscape system, we think the Machine Learning framework opens more possibilities and creativities for landscape architects to explore. Our dataset, code and web demo will be made public available. For future work, video-based GAN such as MoCoGAN HD[15] and DVDGAN [2] with the same data collection pipeline can be used for surrogate modeling of morphodynamics of landscape process such as the formation of braided rivers and deltas.

References

[1] Bradley Cantrell. Synthetic mudscapes. *Paradigms in Computing: Making, Machines, and Models for Design Agency in Architecture*, pages 357–361, 2015.

[2] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv preprint arXiv:1907.06571*, 2019.

[3] Jonathon Doran and Ian Parberry. Controlled procedural terrain generation using software agents. *IEEE Transactions on Computational Intelligence and AI in Games*, 2(2):111–119, 2010.

[4] M Ghorbanian and F Shariatpour. Procedural modeling as a practical technique for 3d assessment in urban design via cityengine. *Iran University of Science & Technology*, 29(2):255–267, 2019.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[6] Brendan Harmon, Anna Petrasova, Helena Mitasova, and Vaclav Petras. Computational landscape architecture: Procedural, tangible, and open landscapes. In *Innovations in Landscape Architecture*, pages 43–59. Routledge, 2016.

[7] Ilmar Hurkxkens and M Bernard. Computational terrain modeling with distance functions for large scale landscape design. *Journal of Digital Landscape Architecture*, 2019(4):222–230, 2019.

[8] Bingchen Liu, Yizhe Zhu, Kunpeng Song, and Ahmed Elgammal. Towards faster and stabilized gan training for high-fidelity few-shot image synthesis. In *International Conference on Learning Representations*, 2020.

[9] Xun Liu. The third simulation: Augmented reality fluvial modeling tool. *Journal of Digital Landscape Architecture,(5)*, pages 413–421, 2020.

[10] Jacob Olsen. Realtime procedural terrain generation. 2004.

[11] William L Raffe, Fabio Zambetta, and Xiaodong Li. A survey of procedural terrain generation techniques using evolutionary algorithms. In *2012 IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2012.

[12] Francisco Caio Maia Rodrigues, Joaquim Bento Cavalcante Neto, and Creto Augusto Vidal. Split grammar evolution for procedural modeling of virtual buildings. In *2015 XVII Symposium on Virtual and Augmented Reality*, pages 75–83. IEEE, 2015.

[13] ryan rs spick and james walker. Realistic and textured terrain generation using gans. In *European Conference on Visual Media Production*, pages 1–10, 2019.

[14] Runjia Tian. Suggestive site planning with conditional gan and urban gis data. In *The International Conference on Computational Design and Robotic Fabrication*, pages 103–113. Springer, 2020.

[15] Yu Tian, Jian Ren, Menglei Chai, Kyle Olszewski, Xi Peng, Dimitris N. Metaxas, and Sergey Tulyakov. A good image generator is what you need for high-resolution video synthesis. In *International Conference on Learning Representations*, 2021.

[16] Julian Togelius, Emil Kastbjerg, David Schedl, and Georgios N Yannakakis. What is procedural content generation? mario on the borderline. In *Proceedings of the 2nd international workshop on procedural content generation in games*, pages 1–6, 2011.

[17] Xun LIU Zihao ZHANG. Control and uncertainty: Towards a paradigm of prototyping. *Landscape Architecture Frontiers*, 8(4):10, 2020.

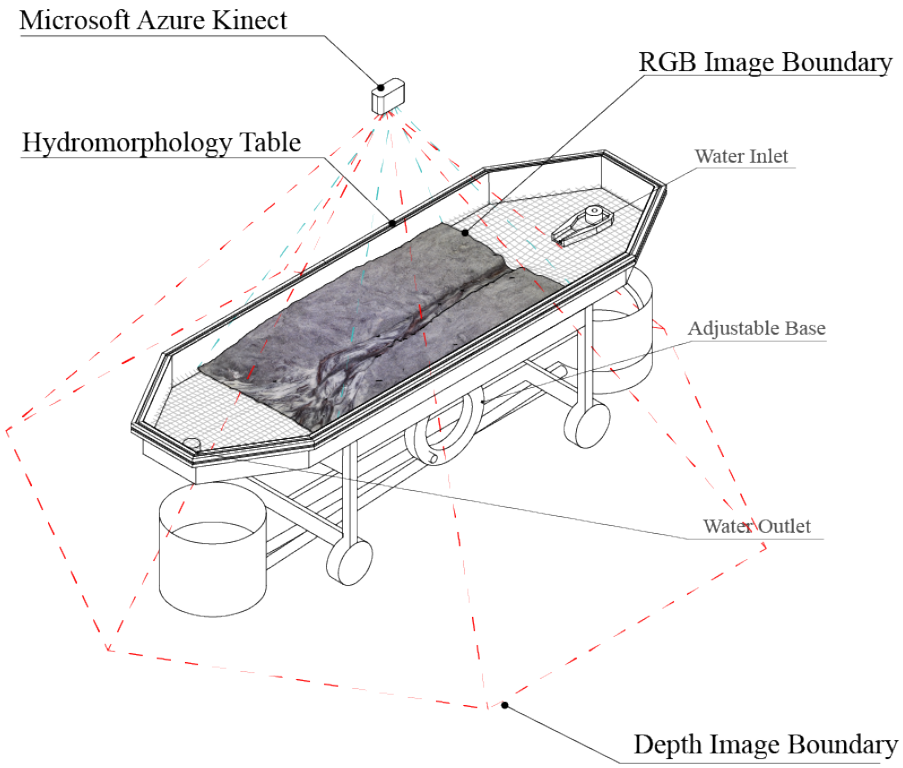

Figure 2: Setup of Hydromorphology Table.

A Hydromorphology Table Setup

We set up the Hydromorphology Table as shown in Figure 2 and perform calibration before data collection, to ensure that the height map and texture map are correctly aligned. The table is equipped with a controllable water pump and adjustable tilted base metal bed, filled with synthetic sediment of varying colors, sizes and densities. As the simulation is running, a Microsoft Azure Kinect sensor mounted can capture real-time data of changing material composition and river shape simultaneously from top. We have run 10 times of 1-hour physical simulations with the same initial setup conditions, from which we sampled with a minimum time interval of 1-minute frame to make sure the patterns are diverse.

B Sample Dataset

The full training dataset consists of 502 pairs of a top-view color image(Figure 3) with its depth image(Figure 4) aligned to the same transform with respect to the Kinect sensor, which were further pre-processed into RGBA images where the height texture is used as alpha channel(Figure 5).

C Latent Walk Visualization

We use the LightweightGAN toolkit for real-time inference of braided river terrain (Figure 6). Then we deploy an inference server with PyTorch and Three.JS for real-time visualization (Figure 7).

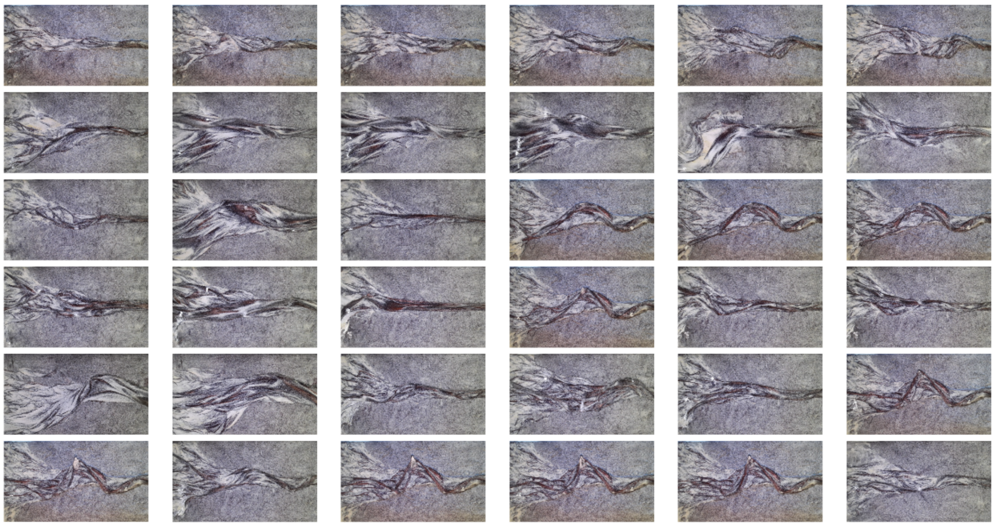

Figure 3: Sample RGB texture data collected with Microsoft Azure Sensor.

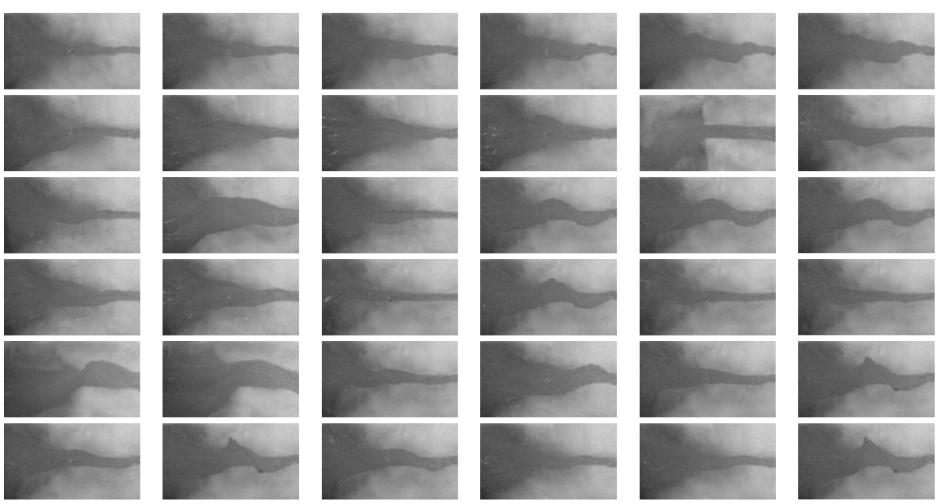

Figure 4: Sample depth map data collected with Microsoft Azure Sensor.

Figure 5: Processed training data by dimsional stacking using OpenCV.

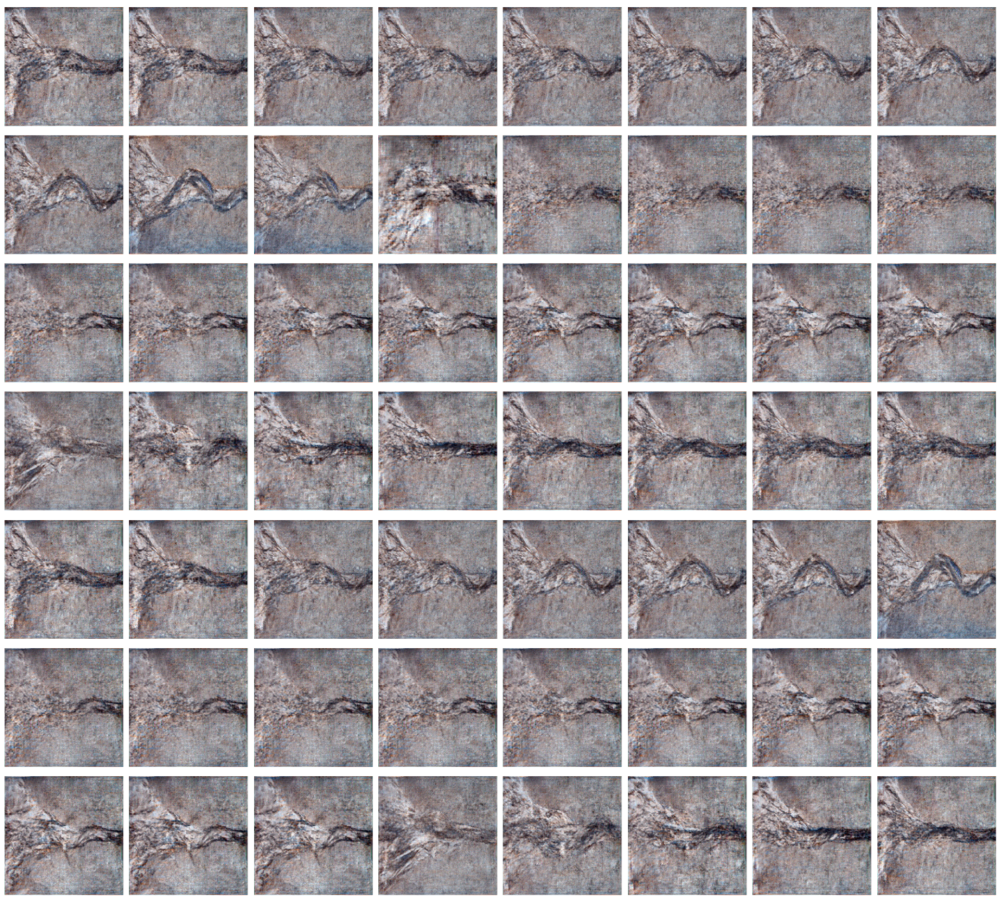

Figure 6: Processed training data by dimsional stacking using OpenCV.

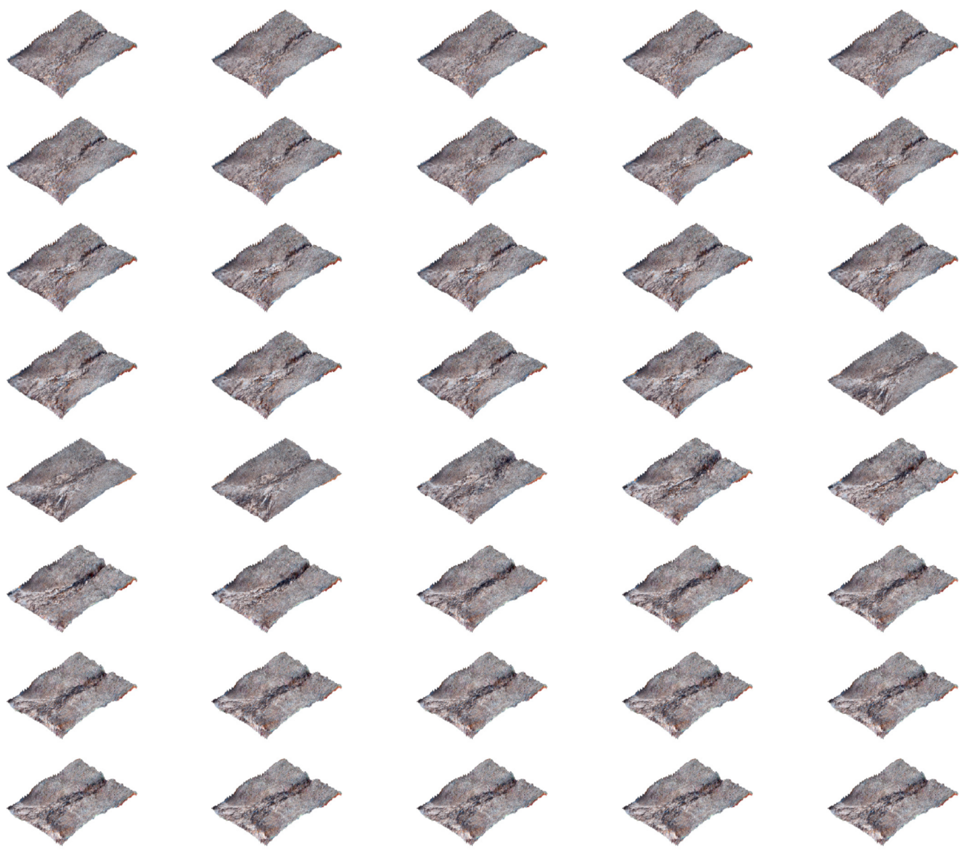

Figure 7: 3D Visualization.