

Boundary-Conditioned Inpainting for Constraint-Consistent Procedural Generation Completion

Zirui Zhang
zhan4192@purdue.edu
Purdue University
West Lafayette, Indiana, USA

Kevin Zhang
zhan4196@purdue.edu
Purdue University
West Lafayette, Indiana, USA

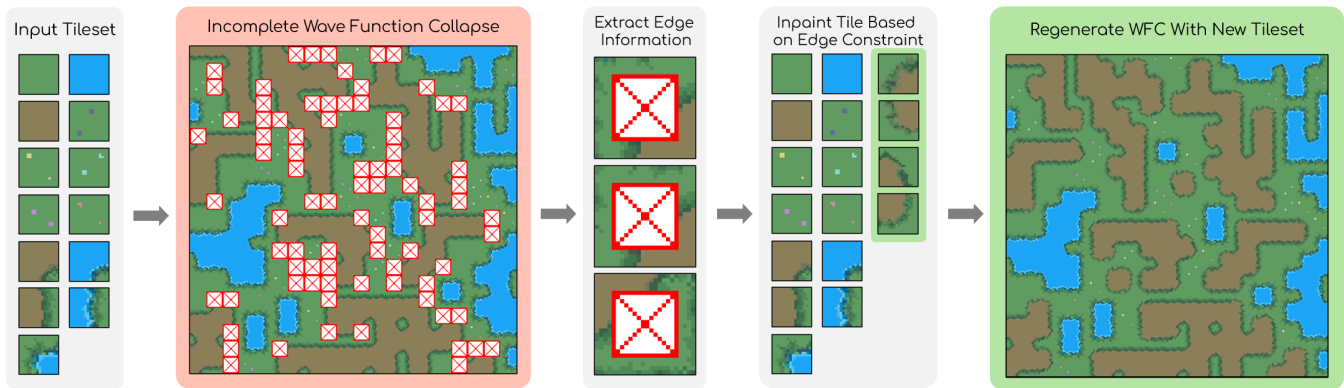


Figure 1: High-level overview of the system pipeline. Given an incomplete input tileset for Wave Function Collapse procedural generation, the boundary condition is extracted from tiles surrounding the missing tile as constraints for diffusion-based inpainting. Additional tiles that satisfy the boundary constraint are generated by diffusion-based inpainting and added to the tileset until all constraints are satisfied, restarting the Wave Function Collapse process.

Abstract

Wave Function Collapse (WFC) is a procedural content generation (PCG) algorithm that assembles large layouts from small tiles by enforcing local adjacency constraints. While simple, it stalls when the rule set or input tileset are incomplete. This work proposes a failsafe that utilizes a boundary-conditioned diffusion inpainting pipeline that is invoked when WFC creates hard constraint conflicts. For any unfilled cell, observed boundary pixels are extracted from its adjacent neighbor cells, masking the interior, and run inpainting diffusion to synthesize a candidate tile, appending it to the input tileset. Tiles generated with this method ensure boundary-consistency with its neighbors without violating any constraints. Cells are prioritized by available context, and accepted tiles are fed back into WFC to continue propagation, repeating until every cell in the grid is filled. Unlike pure-diffusion PCG or continuous texture generation, our method does not directly utilize diffusion to generate content and relies on WFC for PCG, only invoking the pipeline as a local failsafe. While limitations still exist due to reliance on local constraints, we outline some possible extensions for multi-tile inpainting and hierarchical structures.

CCS Concepts

• Computing methodologies → Computer graphics.

Keywords

Inpainting, Diffusion, Procedural Generation, Boundary Conditions, Tileset Completion

ACM Reference Format:

Zirui Zhang and Kevin Zhang. 2025. Boundary-Conditioned Inpainting for Constraint-Consistent Procedural Generation Completion. In *SIGGRAPH Asia 2025 Technical Communications (SA Technical Communications '25)*, December 15–18, 2025, Hong Kong, Hong Kong. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3757376.3771400>

1 Introduction

Tile-based procedural content generation (PCG) aims to create large layouts and structures from small, reusable tiles in a given tileset while enforcing local compatibility. The algorithm is often used to generate content in a multi-cell grid system. Among the various methods, Wave Function Collapse (WFC) has gained popularity as a widely adopted PCG method due to its ability to convert the tile placement problem into a local constraint satisfaction problem (CSP) [Gumin 2016]. Each cell within the grid maintains a candidate set that fits within the adjacency rule set, allowing a lowest-entropy heuristic selection to propagate generation.

Despite its practicality and popularity, WFC still has problems that may arise in production, from rule sets that contain conflicts, tilesets that are incomplete, or generations that propagate to no



This work is licensed under a Creative Commons Attribution 4.0 International License. *SA Technical Communications '25, Hong Kong, Hong Kong*
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2136-6/2025/12
<https://doi.org/10.1145/3757376.3771400>

legal continuation, causing the algorithm to stall. Existing solutions, such as backtracing or relaxed constraints, do not guarantee a solution, especially with hard constraint conflicts.

This work explores a novel failsafe pipeline based on boundary-conditioned diffusion inpainting. When a cell cannot be filled from the available tiles without violating adjacency constraints, we inpaint a missing through use of a diffusion model missing interior tile. The inpainted tile is appended to the overall available tileset. This exterior-boundary condition preserves WFC's local constraint while allowing learned conditional prior to act as a failsafe. The failsafe is only invoked when a combinatorial search stalls, leaving normal WFC behavior otherwise unchanged.

While effective as a fallback, we recognize that this approach currently remains local. Limitations such as insufficient coverage and local/global validation may arise as a result. We discuss these issues and propose solutions for future work in Sec. 5.

To summarize, the contribution of this work includes:

- The formulation of boundary-conditioned diffusion inpainting as a failsafe for tileset completion when WFC stalls.
- A pipeline that acts as a complement to WFC, allowing it to retain its combinatorial strengths.
- Proposal of a path towards multi-tile and hierarchical variants that incorporates broader context while maintaining locality.

2 Prior Work

This section introduces some of the prior work that forms the primary backbone within our pipeline. We begin with a short introduction to the wave function collapse procedural generation algorithm, along with its current applications. Then, we will explore some of the existing diffusion models for procedural generation. Lastly, we will specifically examine applications of diffusion and inpainting in tile-based generative models.

2.1 Wave Function Collapse

Wave Function Collapse (WFC) is a procedural content generation algorithm that is often utilized for tile-based generation. It frames the tile placement problem as a local constraint satisfaction problem (CSP), where each tile within a given tileset has constraints encoded along its edge describing valid adjacent tile placements. The generation proceeds by repeatedly selecting the lowest entropy cell, collapsing into a single deterministic tile, and propagating the result to neighbors.

While this method performs well given the right conditions, it is quite susceptible to incomplete rule sets or tilesets, such as missing transition pieces or asymmetric allowances, demonstrated by Fig. 2. Propagation based on incomplete input can create an empty set for a cell's candidate list, leading to conflicting edge constraints. Some implementations attempt to handle this with additional features such as backtracking or simply restarting [Christie et al. 2024], but convergence is still not guaranteed.

In this work, we utilize WFC as a first-pass planner. When the algorithm stalls or the tileset does not contain the necessary piece to fill a cell, a border-conditioned inpainting model is invoked to propose a tile that is locally admissible given the surrounding borders. Our proposed approach aims to be complementary to

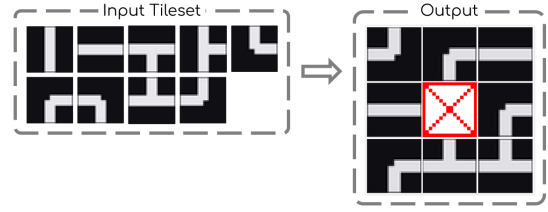


Figure 2: Procedurally generated 3 by 3 grid produced by Wave Function Collapse algorithm using incomplete input tileset, producing incomplete solution with missing center tile.

WFC, preserving its local constraint generation properties while providing a fallback when local CSP cannot be satisfied.

2.2 Diffusion-Based Tile Procedural Generation

Diffusion models have emerged in recent years as an alternative to autoregressive approaches for procedural content generation [Dai et al. 2024; Li et al. 2024; Podell et al. 2023]. For PCG tasks, they offer strong editability through masking and iterative denoising, typically used to synthesize or complete textures or sprite grids. In the inpainting variant specifically, they allow a PCG system to specify where to generate [Dai et al. 2024; Gonzalez and Guzdial 2023], allowing them to be easily integrated into grid-based content generation.

Popular approaches for utilizing diffusion models for PCG include level reconstruction [Dai et al. 2024; Gonzalez and Guzdial 2023; Sarkar and Cooper 2021], texture synthesis [Rodriguez-Pardo et al. 2024, 2019; Sartor and Peers 2024; Zhou et al. 2018], and continuous tiling [Cohen et al. 2003; Madar and Fried 2025; Rodriguez-Pardo and Garces 2023]. Notably, Gonzalez et al. [Gonzalez and Guzdial 2023] suggested in their future work that adapting WFC for level inpainting involves treating the masked region as diffusion regions and the surrounding areas as already generated by WFC. While a viable option, the method outlined do not allow for reusable tiles at cells with the same edge constraint. Our approach follows a similar idea, where inpainting occurs from the exterior boundary when constraint conflicts arise, but we shift resolution to a learned conditional prior and enforce compatibility via an explicit boundary check.

2.3 Continuous Tile Diffusion

On the other hand, the continuous texture synthesis approach for PCG aims to extend an existing texture field by sampling pixels or regions along its boundary so the extended texture remains visually consistent [Rodriguez-Pardo et al. 2024, 2019; Sartor and Peers 2024; Zhou et al. 2018]. This approach often operates on a single canvas, optimizing for local continuity and deliberately avoiding discrete boundaries. The issue with the continuous texture synthesis approach is that it lacks compatibility guarantees with new neighbors and is strictly limited to its local context. In contrast to this approach, our work aims to produce a discrete reusable tile that is placed into cells within a grid system under hard adjacency constraints.

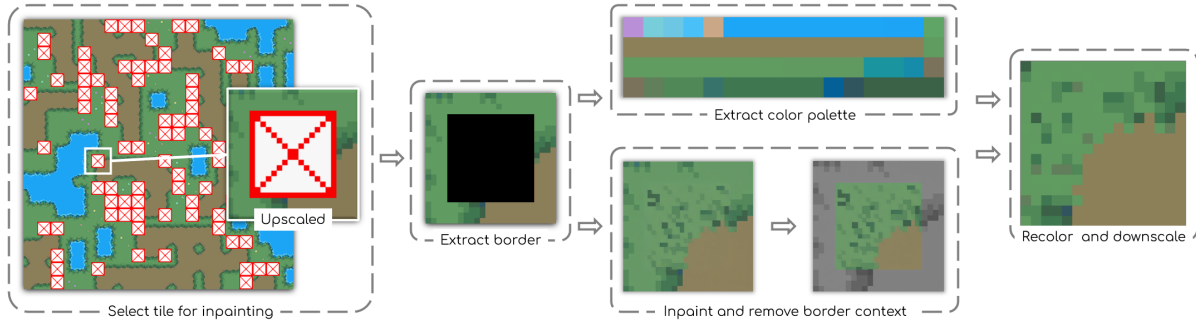


Figure 3: Pipeline overview for context extraction, inpainting, and resample. An unfilled tile is selected and upscaled for inpainting, prioritizing tiles with more filled adjacent neighbors. Edge mask and canvas is created using adjacent context. New tile is extracted from the inpainting output, downsampled to original tile size, and recolored based on existing color palette from adjacent tiles.

Approaches that generate tile via exterior-boundary inpainting follow a similar idea, utilizing Wang-style edge ruleset to synthesize an interior that respects surrounding edges [Cohen et al. 2003; Madar and Fried 2025; Rodriguez-Pardo and Garces 2023]. However, this line of work assumes a fully specified image canvas with the goal of extending outward by repeat-delay inpainting its edges, effectively performing content-aware outpainting with tile structure. While Sartor et al. [Sartor and Peers 2024] have also demonstrated self-generation that inpaints interior regions of the image, it makes no specific assumption about the input, as it directly extracts template tiles from existing images. Our work extends upon this idea with a partially collapsed WFC state with fixed neighboring tiles, invoking inpainting only as a failsafe when the initial input tileset or rule set creates constraint conflicts. Our outputs remain reusable across placements, making the method a local completion mechanism for a CSP-driven pipeline rather than a general outpainting system.

3 Pipeline Structure

This section introduces the inpainting pipeline as a modification to the standard wave function collapse generation algorithm. Our approach combines WFC with boundary-based diffusion generation process, which makes use of a diffusion inpainting model to generate new tiles for use in solving the CSP. This allows for full convergence from a single initial solution, even after an initial solution has reached a “dead-end” using its input tileset. Additionally, we provide specification for the model, training, and diffusion process.

3.1 Tile Selection and Inpainting

Given a partially completed tileset and its associated constraint graph, the algorithm begins by running WFC to generate a preliminary layout. This approach continues until it is impossible to add a new variable based on solution constraints.

All unresolved cells are identified after the first pass. Tiles are then selected to be completed, with higher priority given to those incomplete tiles with more resolved adjacent neighbors, ensuring that tiles with more surrounding structural information are addressed first.

For a selected tile, we extract a border context of up to eight tiles from neighboring tiles at the tile’s native resolution. A copy of the edge band of 4 pixels is copied from each filled neighbor into a small conditioning canvas covering the target tile. The canvas is then upscaled to 512 by 512 pixels, with a binary mask applied to the interior. A diffusion inpainting model is applied on the canvas, denoising the masked interior region while treating the copied border as hard context, as seen in Fig. 3. It should be noted however, while we used 4 pixels for border context in our inpainting pipeline implementation, it can easily be adjusted using different masks, including non-uniform shapes.

After inpainting, the initial hard context border region is cropped from the diffusion output image, and the completed tile is downsampled back to 16×16 pixels and recolored according to the existing palette, ensuring visual consistency with the original tileset. The generated tiles are appended to the initial tileset and treated as new valid tile options for the WFC. The partially filled solution then continues from its current solution with the updated tileset, further propagating constraints and identifying the next incomplete tile where we begin diffusion inpainting again. This process is repeated until all tiles are filled. It should be noted that, it is entirely possible for a given input tileset to contain all necessary tiles to generate the an output, skipping the diffusion generation step entirely.

3.2 Model Training

We implemented the pipeline using *SD-XL Inpainting 0.1* [AI 2023], which is an inpainting model based on Stable Diffusion model [Blattmann et al. 2022]. Training was performed on 50 pixel art style texture tiles upscaled to 512 by 512 pixels, with fp16 and an effective batch of 4, for 2000 steps. For all of the inpainting process in this work, we used an Euler scheduler with an inference step of 24 and target resolution of 512 by 512.

4 Results and Evaluation

Our pipeline successfully generates constraint satisfying tiles given surrounding boundary context. For every conflicting edge constraint variants, only one tile needs to be generated to continue propagating the WFC algorithm. Notably, while we recommend using 25% of surrounding tiles as boundary context for enforcing

adjacency constraint, the pipeline can also work with masks of varying sizes and shapes.

5 Limitations and Future Work

While our pipeline solves a specific problem with the constraint-based WFC, We acknowledge that many limitations still exist. Furthermore, we believe there exists many approaches to improve our method for specific procedural generation tasks.

5.1 Insufficient Coverage

Our method relies on a conditional distribution over surrounding tiles as a local border signature. While it ensures edge consistency, it may lead to insufficient coverage when surrounding tiles do not provide adequate samples of the full tileset. In such cases, while it will still satisfy local edge constraints, it will likely produce a stylistically incorrect result, as it defaults to priors dominated by frequent textures.

This occurs most commonly during underspecific vocabulary, where the tileset simply lacks the necessary transition pieces, leading to certain border signatures never occurring during generation. Furthermore, this issue rarely presents a hard constraint violation, as the generated tile is still based on surrounding edge constraints.

While our selection heuristic prioritizes empty tiles with 8 surrounding tiles already completed, sometimes it will inevitably have to select a tile without sufficient surrounding tiles, such as a map edge or corner tile. This issue in practice can lead to diffused tiles having generic-looking interiors or semantically incorrect features.

5.2 Local and Global Validation

Similarly, it is possible to generate a resulting tileset that satisfies all local border constraints yet violates the global design constraint. In an edge-conditioned inpainting setup, the diffuser optimizes a conditional prior over the missing interior given its immediate neighbors, and WFC enforces local adjacency consistency one cell at a time. Both are fundamentally local, which means they have no persistent state for long-range or multi-tile structures. This means while it will generate a sequence of individually valid choices, it may accumulate into a configuration that is globally incorrect.

5.3 Multi-Tile Structures

Our current approach only considers single-tile inpainting conditioned on immediate neighbors. An extension of this work can consider a generalized border consistency check that includes tiles surrounding a small patch, likely consisting of multiple smaller square tiles. It is also possible to utilize the hierarchical generation that organizes varying-sized tiles as tree structures, as demonstrated by many prior works [Alaka and Bidarra 2023; Beukman et al. 2023], for more complex generation. The root would encode macro-structures such as biome, rivers, or buildings, with the intermediate nodes corresponding to subregions, and leaves as corresponding tiles.

6 Conclusion

This work introduced a hybrid pipeline that acts as a failsafe for the Wave Function Collapse algorithm, using exterior-boundary-conditioned diffusion inpainting. When WFC encounters constraint conflicts or lacks the necessary constraint-satisfying tile, the pipeline

selects the local neighboring tile, treating the target cell as a masked region. It then inpaints the interior based on the observed border pixels to create a constraint-satisfying tile for appending to the tileset. This pipeline uses strictly local constraints to preserve the WFC's efficiency while providing a failsafe to allow continued generation without relaxing any global constraint. Despite some of the stylistic limitations, notably multi-tile generation and hierarchical structures, we believe it can easily be extended to cover most variations of Wave Function Collapse while maintaining the practical advantage of boundary-conditioned PCG.

References

- Stability AI. 2023. stable-diffusion-xl-1.0-inpainting-0.1. <https://huggingface.co/diffusers/stable-diffusion-xl-1.0-inpainting-0.1>.
- Shaad Alaka and Rafael Bidarra. 2023. Hierarchical semantic wave function collapse. In *Proceedings of the 18th International Conference on the Foundations of Digital Games*. 1–10.
- Michael Beukman, Branden Ingram, Ireton Liu, and Benjamin Rosman. 2023. Hierarchical wavefunction collapse. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 19. 23–33.
- Andreas Blattmann, Robin Rombach, Kaan Oktay, and Björn Ommer. 2022. Retrieval-Augmented Diffusion Models. doi:10.48550/ARXIV.2204.11824
- Robert Christie, Brian Kitchen, Wiktor Tumilowicz, Steffan Hooper, and Burkhard C Wünsche. 2024. Procedurally Generating Large Synthetic Worlds: Chunked Hierarchical Wave Function Collapse. In *2024 39th International Conference on Image and Vision Computing New Zealand (IVCNZ)*. IEEE, 1–6.
- Michael F Cohen, Jonathan Shade, Stefan Hiller, and Oliver Deussen. 2003. Wang tiles for image and texture generation. *ACM transactions on graphics (TOG)* 22, 3 (2003), 287–294.
- Shiqi Dai, Xuanyu Zhu, Naiqi Li, Tao Dai, and Zhi Wang. 2024. Procedural level generation with diffusion models from a single example. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 10021–10029.
- Johor Jara Gonzalez and Mathew Guzdial. 2023. Reconstructing existing levels through level inpainting. In *Proceedings of the Nineteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (Salt Lake City) (AIIDE '23)*. AAAI Press, Article 28, 8 pages. doi:10.1609/aiide.v19i1.27523
- Maxim Gumin. 2016. *Wave Function Collapse Algorithm*. <https://github.com/mxgmn/WaveFunctionCollapse>
- Beichen Li, Yivwei Hu, Paul Guerrero, Milos Hasan, Liang Shi, Valentin Deschaintre, and Wojciech Matusik. 2024. Procedural material generation with reinforcement learning. *ACM Transactions on Graphics (TOG)* 43, 6 (2024), 1–14.
- Or Madar and Ohad Fried. 2025. Tiled Diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2023. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. arXiv:2307.01952 [cs.CV] <https://arxiv.org/abs/2307.01952>
- Carlos Rodriguez-Pardo, Dan Casas, Elena Garces, and Jorge Lopez-Moreno. 2024. TexTile: A Differentiable Metric for Texture Tileability. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4439–4449. doi:10.1109/CVPR52733.2024.00425
- Carlos Rodriguez-Pardo and Elena Garces. 2023. SeamlessGAN: Self-Supervised Synthesis of Tileable Texture Maps. *IEEE Transactions on Visualization and Computer Graphics* 29, 6 (2023), 2914–2925. doi:10.1109/TVCG.2022.3143615
- Carlos Rodriguez-Pardo, Sergio Suja, David Pascual, Jorge Lopez-Moreno, and Elena Garces. 2019. Automatic extraction and synthesis of regular repeatable patterns. *Comput. Graph.* 83, C (Oct. 2019), 33–41. doi:10.1016/j.cag.2019.06.010
- Anurag Sarkar and Seth Cooper. 2021. Generating and Blending Game Levels via Quality-Diversity in the Latent Space of a Variational Autoencoder. In *Proceedings of the 16th International Conference on the Foundations of Digital Games (Montreal, QC, Canada) (FDG '21)*. Association for Computing Machinery, New York, NY, USA, Article 3, 11 pages. doi:10.1145/3472538.3472545
- Sam Sartor and Pieter Peers. 2024. Content-aware Tile Generation using Exterior Boundary Inpainting. *ACM Trans. Graph.* 43, 6, Article 202 (Nov. 2024), 12 pages. doi:10.1145/3687981
- Yang Zhou, Zhen Zhu, Xiang Bai, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. 2018. Non-stationary texture synthesis by adversarial expansion. *ACM Trans. Graph.* 37, 4, Article 49 (July 2018), 13 pages. doi:10.1145/3197517.3201285