STAG: Smart Tools and Applications in Graphics (2025)
M. Comino Trinidad, C. Mancinelli, D. Cabiddu, and D. Giorgi (Editors)

Structured Pattern Expansion with Diffusion Models

Marzia Riso[1], Giuseppe Vecchio[2] and Fabio Pellacini[3]

[1]Sapienza University of Rome, Italy [2]Independent Researcher, Italy [3]University of Modena and Reggio Emilia, Italy

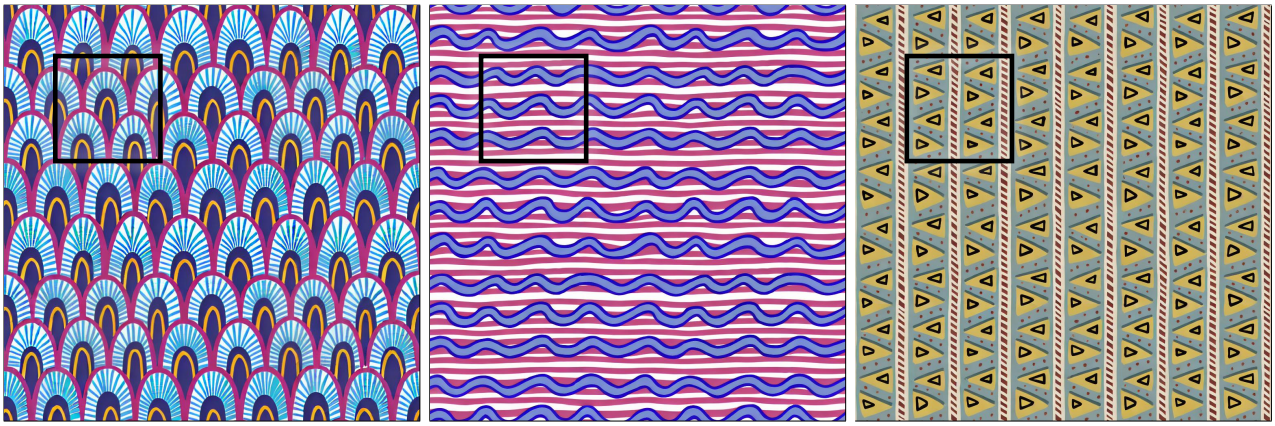

Figure 1: ***Overview.*** *We present a diffusion-based model for structured pattern expansion. Our approach enables the generation of large-scale, high-quality tileable patterns by extending a user-drawn input, shown within the black boxes, to an arbitrarily sized canvas. Our method extends the input pattern while faithfully following the user input and producing coherently structured and yet non-repetitive images.*

Abstract

Recent advances in diffusion models have significantly improved the synthesis of materials, textures, and 3D shapes. By conditioning these models on text or images, users can guide the generation, reducing the time required to create digital assets. In this paper, we address the synthesis of structured, stationary patterns, where diffusion models are generally less reliable and, more importantly, less controllable.

Our approach leverages the generative capabilities of diffusion models specifically adapted to the pattern domain. It enables users to exercise direct control over the synthesis by expanding a partially hand-drawn pattern into a larger design while preserving the structure and details of the input. To enhance pattern quality, we fine-tune an image-pretrained diffusion model on structured patterns using Low-Rank Adaptation (LoRA), apply a noise rolling technique to ensure tileability, and utilize a patch-based approach to facilitate the generation of large-scale assets.

We demonstrate the effectiveness of our method through a comprehensive set of experiments, showing that it outperforms existing models in generating diverse, consistent patterns that respond directly to user input.

CCS Concepts

• ***Computing methodologies*** → ***Texturing;***

1. Introduction

Hand-drawn structured patterns are central to computer graphics, with applications spanning various domains in design and digital art. Creating these patterns remains a complex and time-consuming task that requires specialized expertise. AI-assisted content creation offers the potential to simplify this process. For instance, learning-based image synthesis methods have shown impressive generative capabilities for natural images [RBL*22; BDS19; KALL18; KLA*20; PEL*23]. However, their application to pattern-like synthesis has primarily focused on unstructured, realistic materials [ZZB*18; ZCXH23; ZXL*24; HGZ*23; VSPS24; VMR*24; VD24; LPdC24], leaving the creation of structured patterns an underexplored task. In contrast, the synthesis of highly structured

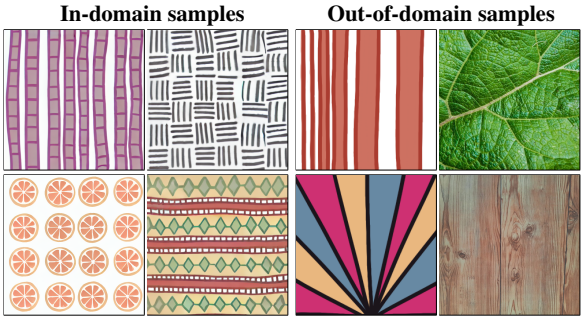

Figure 2: *We focus on structured, stationary, patterns in a hand-drawn style, characterized by repeated recognizable shapes drawn in flat colors (left). Unstructured or aperiodic patterns, as well as photorealistic textures, fall outside the scope of this paper (right).*

vector patterns has been explored by automatically discovering and exploiting their structure, geometry, and topology [TWY*20; RGF*20; TWZ22], or by optimizing the procedural parameters of differentiable vector patterns to match a sketch or a viewport edit [RP23; RSP22].

Our work focuses on structured patterns with a hand-drawn appearance, characterized by the repetition of sketch-like shapes filled with solid colors and defined by sharp, crisp edges. Formally, these structured patterns consist of stationary repetitions of recognizable shapes, each with individual variations, and are drawn with piecewise-constant colors. Examples of these patterns are shown throughout the paper, with Fig. 2 also highlighting examples of textures outside of our scope. We focus on this type of pattern for their importance in design applications and the general lack of learning-based methods addressing its synthesis.

Our approach leverages Latent Diffusion Models [RBL*22] as a foundation for the synthesis. Although these models have achieved significant advances in natural image generation, they are not optimized for generating structured patterns. One key limitation is that the synthesized patterns often lack quality, as these models are typically trained to generate photorealistic images with unstructured, chaotic textures and high-frequency, stochastic color variations. When applied to structured patterns, these methods often fail to maintain the inherent structure, sharpness, and cohesive visual style of the patterns. Furthermore, design applications often require precise pattern control by users, often lacking in generative approaches, generally focusing on text-to-image synthesis. Although high-level conditioning may be sufficient for natural image synthesis, specifying the exact structure and appearance of a pattern is much more challenging. Even when using images as conditioning inputs, existing methods perform inconsistently on structured patterns within our domain. To address this gap in the literature and provide artists with a user-friendly yet controllable content creation tool, we propose a diffusion-based model specifically designed for the synthesis and expansion of structured stationary patterns. In particular, we leverage the extensive knowledge embedded in large-scale models, such as Stable Diffusion [RBL*22; PEL*23], and adapt it to the pattern domain by training a "lightweight" Low-Rank Adaptation (LoRA) [HSW*22]. This approach reduces the computational and data requirements of training a diffusion model from scratch, while retaining the expressive power of models trained on large-scale datasets like LAION [SVB*21]. To this end, we collect a dataset of procedurally designed patterns that we used to train our LoRA.

We base our architecture on an inpainting pipeline, which supports the expansion of a partial, hand-drawn input sketch into a larger pattern while preserving its structural integrity and details. During inference, we leverage noise rolling and patch-based synthesis to produce large-scale, tileable patterns, at high quality in a reliable way. These design choices allow us to generate large-scale, tileable patterns that accurately follow the input sketch, while adding a limited degree of variation and thus avoiding visible repetitions.

We qualitatively evaluate the effectiveness of our approach across a diverse range of input patterns, demonstrating significant improvements over previous state-of-the-art texture synthesis methods. To assess user-perceived quality, we also conduct a user study that captures preferences and perceived fidelity in the synthesized output. In addition, we analyze our architecture through a comprehensive set of experiments and ablation studies, highlighting the benefits of our design choices. The results show that our method consistently generates a wide variety of structure patterns, correctly preserving the structure and visual coherence of the input sketches. In summary, the contributions of our work are as follows:

- we present a new diffusion-based approach for structured pattern synthesis and expansion;
- we introduce a new medium-scale dataset for fine-tuning generative models on the pattern domain;
- we demonstrate the generation capabilities of our model for different types of structured patterns and show its ability to control the generation precisely from input sketches;
- we validate the improvements over other generative methods, non-specifically trained for patterns, underlying the need for a specifically trained model.

2. Related Work

2.1. Texture Synthesis.

Texture synthesis is important in computer graphics, vision, and image processing. Pixel-based methods generate textures pixel by pixel, either by expanding a seed image [EL99] or modifying noise [WL00] to match a reference. However, they often suffer from artifacts or repetition when generation drifts into the wrong part of the texture space. Patch-based methods synthesize textures using patches from the sample image. [LLX*01] enabled real-time performance through efficient patch sampling, while [EF01] improved quality with Image Quilting, blending overlaps to reduce seams. [KSE*03] further optimized boundaries with graph cuts, enhancing global coherence. Extensions include tiling methods such as automatic seamless pattern generation [KS00] and Wang Tiles for non-repetitive textures [CSHD03]. Quality and efficiency have also been improved through refined feature matching [WZ01] and multi-resolution block sampling [YLC02], which captures both coarse and fine details.

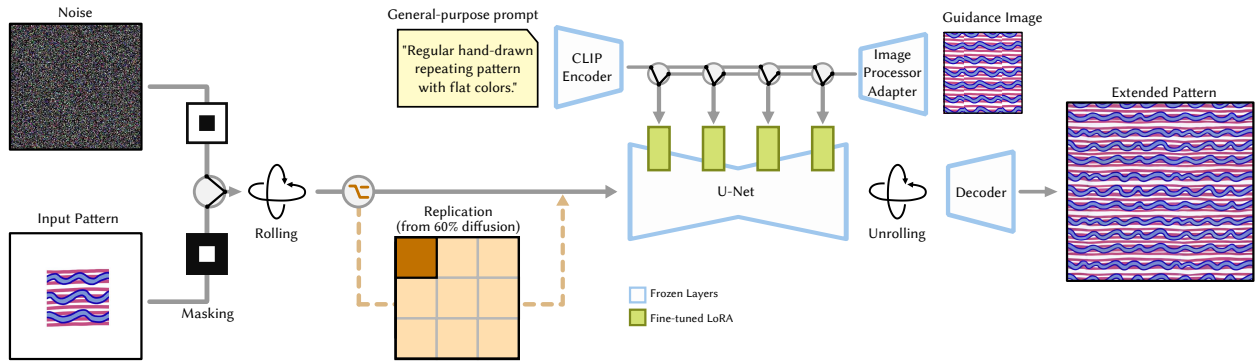

Figure 3: *Starting from a hand-drawn input, we extend it to an arbitrarily large canvas, introducing variations while preserving structure and appearance. The pattern is centered and expanded outward in an "outpainting"-like process. Our method combines text and image conditioning to guide a fine-tuned Latent Diffusion Model [RBL*22] in generating consistent, high-quality patterns. Tileability is ensured by rolling the input tensor at each diffusion step and unrolling afterward. To expand beyond the initial canvas, we replicate the latent tensor after 60% of the diffusion process.*

Relevant to our domain is the generation of near-regular textures, which combine strong regularity with stochastic variation in color and geometry. [LHW*04] compared synthesis algorithms, and [LHW*06] quantitatively evaluated quality. [LLH04] introduced deformation fields for geometry, lighting, and color on a coarse structure, enabling flexible manipulation, while [RHE11] guided random sampling with automatically extracted tiles to produce coherent patterns and handle irregularities.

2.2. Generative models.

Image generation is a long-standing challenge in computer vision due to the complexity of visual data and the diversity of real-world scenes. With the advent of deep learning, the generation task has been increasingly posed as a learning problem, with Generative Adversarial Networks (GAN) [GPM*14] enabling the generation of high-quality images [KALL18; BDS19; KLA*20]. However, GANs are characterized by an unstable adversarial training [ACB17; GAA*17; Mes18], and struggle to model complex data distributions [MPPS17], exhibiting a *mode collapse* behavior and leading to a limited output diversity.

Diffusion Models (DMs) [SWMG15; HJA20; RBL*22] have recently emerged as an alternative to GANs, achieving state-of-the-art results in image generation tasks [DN21] also due to their stable supervised training approach. Furthermore, DMs have enabled a whole new level of classifier-free conditioning [HS22] through cross-attention between latent image representations and conditioning data. More recently, ControlNet [ZRA23] has been proposed to extend generation controllability beyond the typical global-conditioning (e.g.: text prompts) for a fine control over the generation structure. Moreover, approaches like DreamBooth [RLJ*23] and LoRA [HSW*22] allow users to adapt large-scale pre-trained models, to particular tasks or domains, without requiring to fine-tune them and only needing a limited set of training samples.

2.3. Generative models for textures synthesis.

Several works have assessed the synthesis of patterns in the form of natural textures or BRDF materials, with few explicitly focusing on structured patterns. [HVCB21] address the problem of texture synthesis via optimization by introducing a textural loss based on the statistics extracted from the feature activations of a convolutional neural network optimized for object recognition (e.g. VGG-19). [VMR*24] recently introduced ControlMat to perform SVBRDF estimation from input images, and generation when conditioning via text or image prompts. It employs a novel *noise rolling* technique in combination with patched diffusion to achieve tileable high-resolution generation. MatFuse [VSPS24], on the other hand, focuses on extending generation control via multimodal conditioning and editing of existing materials via *volumetric inpainting*, to independently edit different material properties. Although MaterialPalette [LPdC24] focus on extracting PBR materials from a single real-world image, they incorporate a texture extraction module in their proposed pipeline. By fine-tuning a text-to-image diffusion model for each set of material samples, they are able to generate tileable textures at arbitrary resolutions.

Focusing on non-stationary textures, [ZZB*18] proposes an example-based synthesis GAN that is trained to double the spatial extent of crops extracted from an arbitrary texture, using a combination of Style and L_1 losses. After the GAN is trained, its generator can recursively be applied to expand texture samples while coherently maintaining its non-stationary features. [ZCXH23] introduces a new Guided Correspondence Distance metric that can be used as a loss function to optimize the texture synthesis process, improving the similarity measurement of output textures to examples. [ZXL*24], in contrast, leverages a diffusion model backbone combined with a two-step approach and a *"self-rectification"* technique to generate seamless textures, faithfully preserving the distinct visual characteristics of a reference example.

Our method synthesizes a large texture from a small seed provided as a hand-drawn input sketch, offering versatility without requiring further tuning to expand the pattern while preserving the

structure and design properties of the input. In particular, it differs from ControlNet-based methods, which require an additional network and therefore increase computational overhead without significant improvements in performance, in our setting.

3. Method

Our work is based on the Latent Diffusion Model (LDM) architecture [RBL*22], adapted to synthesize high-quality, stationary, structured patterns with a sketched, vector-like appearance. Given a hand-drawn input sample as a seed, we expand it to an arbitrarily sized canvas, introducing subtle variations while preserving overall structure and visual consistency. In particular, we leverage the inpainting capabilities of diffusion models via latent masking, by centrally placing the input pattern on a larger canvas and generating the outer border. Our model extends the design outward in an "outpainting" process, thus appropriately filling the entire frame.

To achieve this, we fine-tune a pre-trained LDM for image generation by training a Low-Rank Adaptation (LoRA) on a dataset of procedurally generated patterns. To further enhance fidelity, we integrate an IP-Adapter [YZL*23] for image-based conditioning. This ensures that the extended design remains visually consistent with the original input, which is loosely replicated to serve as a *guidance image*. We additionally use text prompts to constrain the generation to the structural regularity and solid-color look characteristic of our target domain. To enable seamless extension of patterns to arbitrarily large sizes, we adopt a latent replication strategy, which introduces controlled variations while preserving structural integrity. We also apply the noise rolling technique [VMR*24], to achieve tileable pattern generation. Specifically, latent replication occurs after N iterations, while noise rolling and unrolling are applied before and after each diffusion step, respectively.

An overview of our model architecture is shown in Fig. 3. In the following, we first provide an overview of the latent diffusion architecture for image generation and the approaches to combine text and image conditioning, to then detail our approach and architectural choices specific to the structured pattern domain. We then ablate our design choices and architectural components in Sec. 4.6, demonstrating the benefits of our approach.

3.1. Guided Image Generation

We leverage the Latent Diffusion architecture, consisting of a Variational Autoencoder (VAE) [KW14] and a diffusion U-Net [RBL*22]. The encoder $\mathcal{E}$, compresses an image $\mathbf{x} \in R^{H\times W\times 3}$ into a latent representation $z = \mathcal{E}(\mathbf{x})$, where $z \in R^{h\times w\times c}$, and c is the dimensionality of the encoded image, capturing the essential features in a lower-dimensional space. The decoder $\mathcal{D}$, reconstructs the image from this latent space, adequately projecting it back to the pixel space.

The diffusion process involves a series of transformations that gradually denoise a latent vector, guided by a time-conditional U-Net. During training, noised latent vectors are generated, following the strategy defined in [HJA20], through a deterministic forward diffusion process $q(z_t|z_{t-1})$, transforming the encoding of an input image into an isotropic Gaussian distribution. The diffusion network ϵ_θ is then trained to perform the backward diffusion process $q(z_{t-1}|z_t)$, efficiently learning to denoise the latent vector and reconstruct its original content.

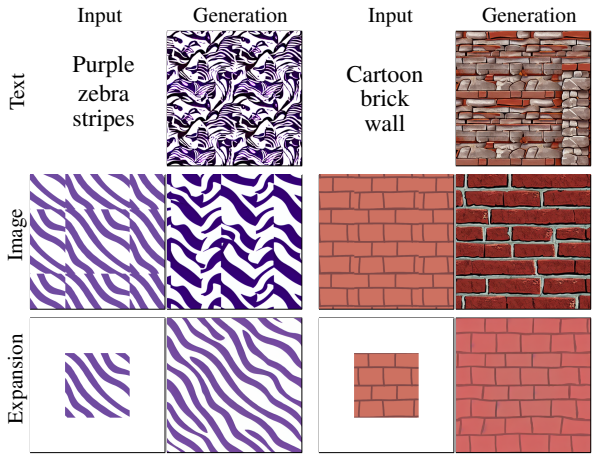

Figure 4: *Generation modes supported by our base architecture. While text or image global conditioning is possible (first two rows), it often yields inconsistent patterns and limited fine-grained control. In contrast, our expansion approach (last row) produces arbitrarily large patterns while preserving input coherence.*

3.1.1. Text conditioning

Latent Diffusion models can typically be globally conditioned with high-level text prompts via cross-attention [VSP*17] between each convolutional block of the denoising U-Net and the embedding of the condition y, extracted by an encoder τ_θ, with the attention defined as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V, \tag{1}$$

where $Q = W_Q^i \cdot \varphi_i(z_t), K = W_K^i \cdot \tau_\theta(y), V = W_V^i \tau_\theta(y)$. Here, $\varphi_i(z_t) \in R^{N\times d_\epsilon^i}$ is the flattened output of the previous convolution block of ϵ_θ, and $W_Q^i \in R^{d\times d_\tau^i}$, $W_K^i \in R^{d\times d_\epsilon^i}$, $W_V^i \in R^{d\times d_\epsilon^i}$, are learnable projection matrices.

The training objective in the conditional setting becomes

$$L_{LDM} := E_{\mathcal{E}(M),y,\epsilon\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t,t,\tau(y))\|_2^2\right]. \tag{2}$$

We use the CLIP [RKH*21] ViT encoder from Stable Diffusion v1.5 as our text encoder τ, with a patch size of 14×14.

Despite the expressive capabilities of text, which has shown remarkable results in the context of natural image synthesis, accurately describing a pattern structure with text is challenging since it would require precise definitions of the pattern shapes, their positions, and symmetries in relation to the other, and their appearance features. For this reason, we mostly use image conditioning as described below. At the same time, we use text prompting to reinforce general pattern features, such as regularity and symmetry, rather than targeting a specific example. To achieve this, we designed a *general purpose prompt*, aimed at infusing the generation process

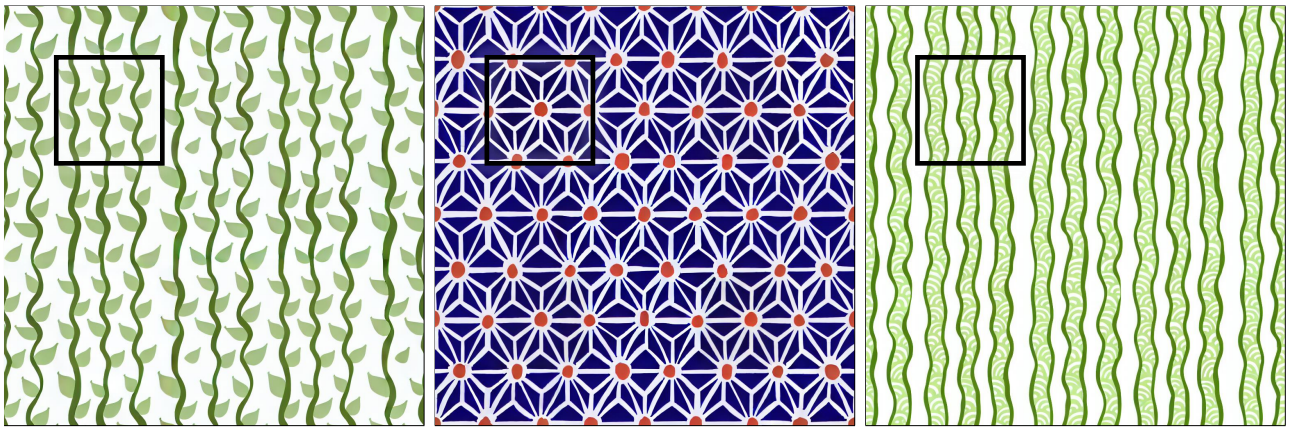

Figure 5: *The figure demonstrates our model's versatility in expanding different pattern types, with inputs enclosed in black boxes. The left and right samples display organic designs, highlighting seamless extrapolation of intricate details while maintaining a natural flow. The central panel features a geometric pattern successfully extended to a larger area.*

with these common characteristics. As illustrated in Fig.3, we employ the prompt *"Regular hand-drawn repeating pattern with flat colors"* for all our results and evaluate its effectiveness against a tailored pattern prompt in Sec.4.6.

3.1.2. Image conditioning

To provide better control of the synthesized pattern, we propose to combine the general purpose prompt, commonly valid for all our patterns, with image conditioning via an IP-Adapter [YZL*23] model. This lightweight adapter achieves image prompting capability, for pre-trained text-to-image diffusion models, through a decoupled cross-attention mechanism that separates cross-attention layers for text features and image features. In particular, the adapter computes separate attention for the text and image embeddings, which are then summed before being fed to the next U-Net layer. The output of the new cross-attention is computed as:

$$\begin{aligned}\text{Atten.}(Q,K_t,V_t,K_i,V_i) = \text{softmax}\left(\frac{QK_t^T}{\sqrt{d}}\right)V_t + \\ + \text{softmax}\left(\frac{QK_i^T}{\sqrt{d}}\right)V_i\end{aligned} \tag{3}$$

with K_t, V_t, K_i, V_i being respectively the keys and values for the text and image embeddings. During the training of the IP-Adapter, only the image cross-attention layers are trained, while the rest of the diffusion model is kept frozen.

This approach has shown remarkable performances in controlling the generation process with image prompts, allowing it to closely follow the reference image. In our experiments, we constructed the guidance image by replicating the original sketch across the entire canvas, resulting in a repetitive pattern that is regular but neither tileable nor fully consistent.

However, global conditioning through text or image prompts alone lacks the level of detail necessary to capture and reproduce the characteristics of our class of patterns, as shown in Fig. 4. We address these limitations by employing the expansion strategy described in Sec. 3.3.

3.2. Stable Diffusion Finetuning

We fine-tune the Stable Diffusion 1.5 model [RBL*22] to our specific pattern domain to achieve consistent and visually coherent pattern synthesis and expansion. In particular, we leverage a Low-Rank Adaptation (LoRA) technique [HSW*22] for efficient fine-tuning of large, pre-trained models, limiting the number of training parameters, while also avoiding catastrophic forgetting, which is the tendency of a model to lose previously learned knowledge when fine-tuned on new data.

In particular, we train a low-rank matrix and add it to the transformer layers of the base Latent Diffusion Model (LDM):

$$\theta' = \theta + \Delta\theta, \tag{4}$$

where θ represents the original weights of the transformer in the LDM, and $\Delta\theta$ is the low-rank update, computed as:

$$\Delta\theta = U \cdot V^T, \tag{5}$$

with $U \in R^{r\times d}$ being the trainable matrices, and r much smaller than d, the dimensionality of the layer's parameters.

This fine-tuning step focuses the generation on the pattern domain and is mostly responsible for the model's ability to maintain stylistic consistency and detailed coherence specific to the target patterns. By introducing these low-rank updates, we ensure that the model adapts efficiently to the specific feature of the pattern domain, without losing its expressive capabilities from the training on the image domain.

3.3. Pattern expansion

To ensure aesthetic coherence and seamless tileability during pattern expansion to arbitrary sizes, we use the Stable Diffusion 1.5 model trained for image inpainting [RBL*22]. Exploiting its ability to interpret partial images and generate coherent completions, we combine inpainting with latent replication and noise rolling [VMR*24] to produce tileable high-quality expansions. This ensures that the extended patterns retain consistency with the

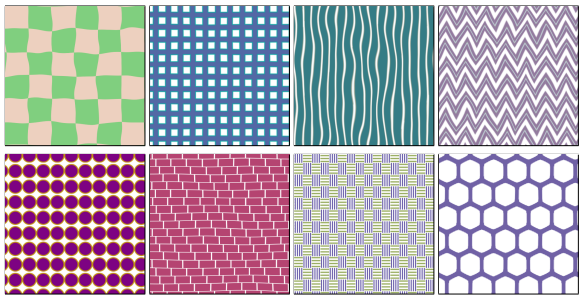

Figure 6: *Our custom pattern dataset includes samples from eight classes. For each pattern, we use the generating procedural parameters to craft captions that accurately describe its design details, creating pattern-text pairs while training the LoRA.*

original input, preserving the overall visual appearance, as demonstrated in Fig. 4 and Fig. 5.

In particular, we start the denoising process at the model's native resolution of 512×512, placing the input at the center of the canvas and leveraging the inpainting capabilities of Stable Diffusion to fill the masked area. However, although inpainting can reconstruct missing parts, it tends to lose long-term dependencies inside the image, leading to inconsistencies in the global structure. This limitation arises from the local receptive field of the model, which makes it challenging to maintain consistency in regions farther from the fixed input pattern. As a result, border artifacts can appear, disrupting the overall structure on the expanded canvas. To address this, we incorporate noise rolling, which cyclically shifts the latent representation z_t at each diffusion step. This technique adequately refocuses the model's receptive field, allowing it to capture a broader spatial context and better preserve structural coherence throughout the pattern. In particular, for each diffusion step, we compute:

$$z'_t = \text{roll}(z_t, \Delta x, \Delta y), \tag{6}$$

where $\text{roll}(\cdot, \Delta_x, \Delta_y)$ denotes the cyclic shift operation along the image's width (Δ_x) and height (Δ_y). After rolling, the model estimates the noise component and performs a denoising step, computing z'_{t-1}. Subsequently, the latent space is unrolled back to its original configuration to maintain the integrity of the global pattern structure:

$$z_{t-1} = \text{roll}(z'_{t-1}, -\Delta x, -\Delta y). \tag{7}$$

By manipulating the latent space in this manner, the model suitably treats the pattern's edges as interconnected, thus intrinsically minimizing the presence of visible seams.

To tailor the expansion process to arbitrarily large canvases, we replicate the latent tensor to fit the target size. As illustrated by the dashed path in Fig. 3, We carry out this phase after N iterations and continue with patched diffusion for the remaining denoising steps. In our experiments, we set N to 60% of the inference steps, resulting in the best compromise between pattern consistency and variation. By combining latent replication and noise rolling, we support a larger expansion while guaranteeing the quality of the generated patterns, as demonstrated in Sec. 4 and in the ablation study in Sec. 4.6.

4. Experimental results

4.1. Datasets

Due to the lack of publicly available pattern datasets, we created a custom dataset consisting of 4000 patterns of 8 classes, namely *grids*, *checkers*, *stripes*, *zigzag*, *dots*, *bricks*, *metal* and *hexagons*, each of which is showcased in Fig. 6. Such classes were specifically designed to expose strong geometrical structures and shape arrangements, to help the LoRA learn the key features of structured pattern domains. For each class, we defined an ad-hoc procedural program capable of generating a diverse set of samples in both design and colors. To simulate a sketched style, we combined our patterns with varying scales of Perlin noise [Per85], introducing the irregularities commonly found in hand-drawn designs.

Our dataset consists of procedurally generated pattern-text caption pairs. We generate each pattern by randomly sampling a value in the proper range for each procedural parameter, including colors. We use these values to build the descriptive caption that highlights the main feature of the pattern. We use a caption template for each pattern class, that is filled with details drawn from the procedural parameter values. As an example, the caption matching the checkered pattern in Fig. 6 (top left) is generated from the base caption of *"A hand-drawn checkered pattern. Checkers are colored in* `<even_color>` *and* `<odd_color>`*, and their size is* `<checker_size>`*. Checkers are surrounded on all four sides by a checker of a different color. Colors are flat and without shading."*, where the free variables are completed by "light green", "wheat" and "big" respectively. For each class, we sample 500 different parameter sets and generate the corresponding pattern-text pairs for training For inference, we manually sketched 35 small pattern samples on a graphics tablet, whose expansion is shown in Fig. 1, Fig. 5, Fig. 11 as well as in all the experiments reported in this paper.

4.2. Technical details

We train our LoRA with a mini-batch gradient descent, using the Adam [KB14] optimizer with a learning rate set to 10^{-4} and a batch size of 8. The training is carried out for 5000 iterations on a single NVIDIA RTX3090 GPU with 24GB of VRAM, using the pre-trained inpainting Stable Diffusion 1.5 checkpoint from [RBL*22].

At inference time, generation is performed by denoising a latent random noise for 50 steps, using the DDIM sampler [SME21] with a fixed seed. Expanding an input sketch takes about 2 seconds at 512×512 and 4 seconds at 1024×1024 and 6GB of VRAM, about 12 seconds at 2048×2048 and 8GB VRAM. Memory requirements can be further reduced by processing fewer patches in parallel, albeit at the cost of increased computation time.

4.3. Results and comparisons

We evaluate our model generation capabilities when conditioning using either text or image. Despite not being the main focus of this

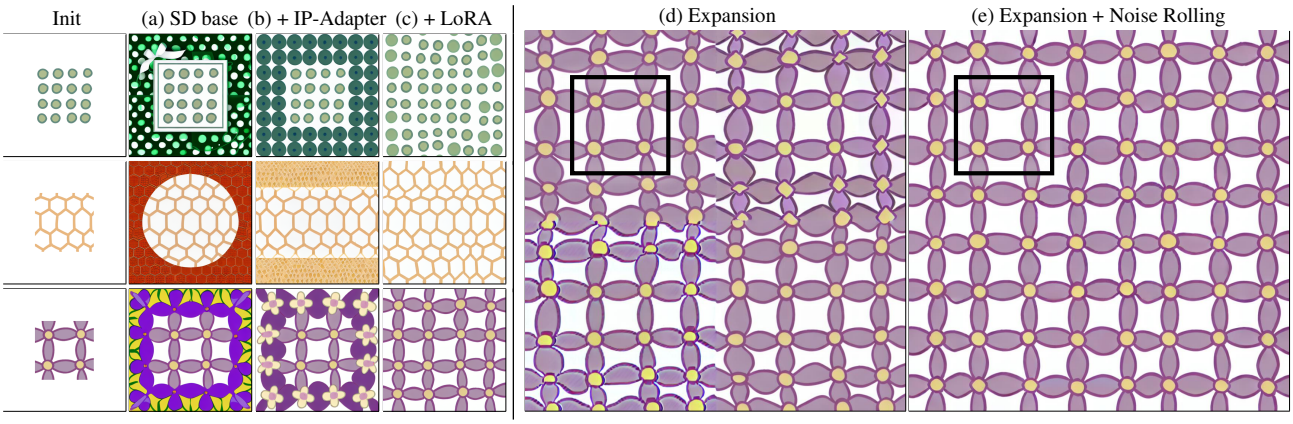

Figure 7: *We evaluate performance improvements from each design choice. The base inpainting model often loses visual coherence during pattern expansion, but adding IP-Adapter conditioning helps align outputs with the prompt. LoRA fine-tuning further strengthens coherence and quality across generations, while noise rolling ensures tileable results by removing seams and artifacts.*

work, we show that our architecture is able to generate pattern-like images when being globally conditioned. However, as shown in Fig. 4, the generation style tends to diverge significantly from the guidance image, highlighting the need for stronger constraints for a specific pattern expansion that closely follows the input sample.

4.3.1. Expansion results

We evaluate our model generation capabilities for pattern expansion (Fig. 1, Fig. 5, Fig. 11 and Supplementary Materials). The results demonstrate that our method closely follows the input prompt, highlighting the pattern expansion capabilities of our model both in terms of quality and coherence of the expanded result.

All the result figures, present the original input pattern contained within a black box, while the surrounding part of the canvas is filled during inference time. Our pipeline successfully extends the input pattern, maintaining coherence and preserving the structural integrity of the original design. Each generated pattern flows naturally from the input, ensuring that there are no abrupt transitions or noticeable repetitions. The model keeps the colors consistent in the generated area, matching the original input. Due to the adoption of the *noise rolling* technique, all results are tileable, allowing seamless repetition. All the provided examples use an expansion factor of 2 for both width and height dimensions.

4.3.2. Comparison

We compare our approach against several state-of-the-art methods: [ZZB*18], [HVCB21], GCD [ZCXH23], MatFuse [VSPS24], [ZXL*24] and MaterialPalette [LPdC24]. For each method, we use the official code and weights released by the authors and adapt our input to match the ones required by each method. As MatFuse [VSPS24] is trained to generate PBR materials, we provide the pattern as the diffuse component of the material, initializing the other properties to the default values. Simirarly, we only exploit the tileable texture extraction module of MaterialPalette [LPdC24], leaving the SVBRDF estimation module aside.

As shown in Fig. 8, our method significantly improves on previous approaches in preserving the structural integrity and visual fidelity of patterns. While methods like [HVCB21] and GCD capture the visual features of the patterns, they tend to break the overall structure, introducing unnatural distortions resulting quality degradation. MatFuse fails to capture the appearance of the pattern, mostly due to the training on natural textures, being only able to reproduce the colors and general shape of the pattern but lacking any fine details. [ZXL*24], in contrast, is generally able to reproduce the sharp visual appearance of the pattern, and capture the main features; however, due to its main focus on non-stationary textures, it tends to break the overall structure, resulting in sharp discontinuity edges inside the image and transitioning between different parts of the pattern. Additionally, it struggles with very sparse patterns (e.g., third column in Fig. 8), and introduces a color shift on the original input. Despite being the only method capable of producing tileable results, MaterialPalette [LPdC24] still faces significant challenges in preserving the integrity of patterns, both in terms of geometrical features and scale. Compared to the other approaches, our work is able to capture the visual features of the pattern and extend it seamlessly, introducing slight variations without altering the overall structure. Moreover, all of our expansion results are tileable, thanks to the use of noise rolling at inference time.

For the sake of completeness, we also compared our method to [ZZB*18], which aims to double the spatial size of a texture by leveraging a GAN specifically trained to reconstruct a $2k \times 2k$ texture from a $k \times k$ patch. We trained a GAN model for each of the input patterns, which required approximately 1.5 hours each. As shown, [ZZB*18] is able to reproduce the overall pattern structure while spatially extending the input sketch, but it also introduces several artifacts and discontinuities that degrade the overall quality output. Moreover, tileability is not imposed, resulting in longer training times to scale to higher-resolution images. In contrast, our approach achieves superior results in both structure preservation and image quality, without requiring ad-hoc training for each texture, while preserving the input sketch and naturally producing tileable outputs.

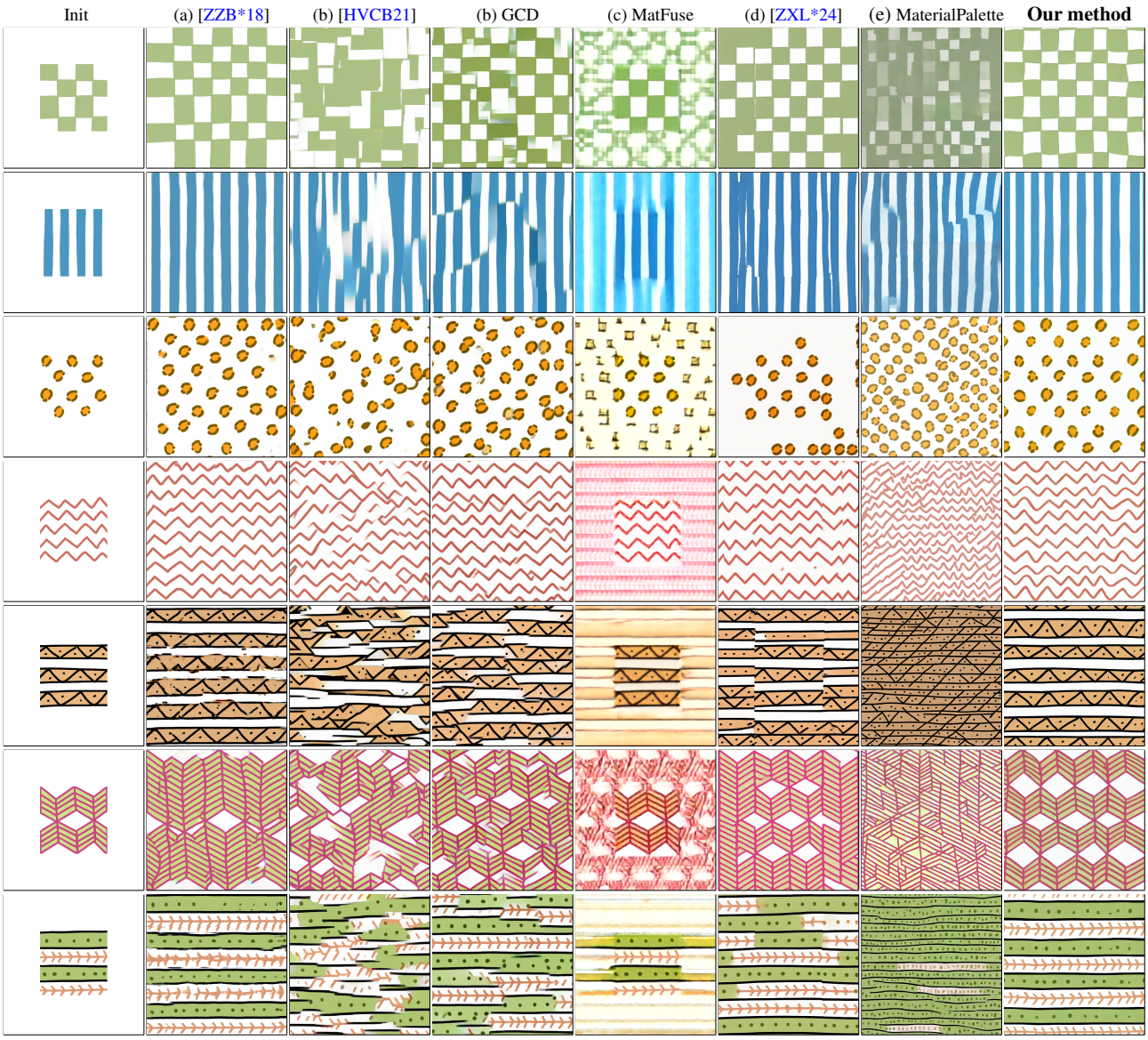

Figure 8: *We compared our method with established texture and material synthesis techniques. As previous work tends to break the overall structure (a, b, c, e, f) or fails at reconstructing the pattern appearance (d), our method consistently expands the input by preserving structural integrity and input coherency.*

It is important to highlight that our pipeline completes the diffusion step in around 2 seconds, whereas the execution times for [HVCB21], GCD [ZCXH23], and [ZXL*24] and [ZZB*18] from at least 2 minutes up to 1.5 hours for each example, where training is involved. MatFuse [VSPS24] has similar timings to our method, due to the similar diffusion backbone, but shows significantly worse generation quality.

4.4. User Study

To evaluate our method's performance, we conducted a comparative study involving prior methods, namely [HVCB21], GCD [ZCXH23], MatFuse [VSPS24], and [ZXL*24], excluding those with excessive training times or evident scale issues. The user study involved 80 MS/PhD students in computer science who were tasked with selecting their preferred expanded pattern based on the quality and consistency of the generation. We showed each one of them 20 randomly chosen pattern generations–including both the input crop and the output for each method compared, in a random sequence–from a set of 35 expanded patterns. Our approach received a higher number of votes (**Ours=1471 (i.e.: 91.93%)**, [HVCB21]=0, GCD=7 (i.e.: 0.44%), MatFuse=0, [ZXL*24]=122 (i.e.: 7.63%)), showing a significant general preference of the ex-

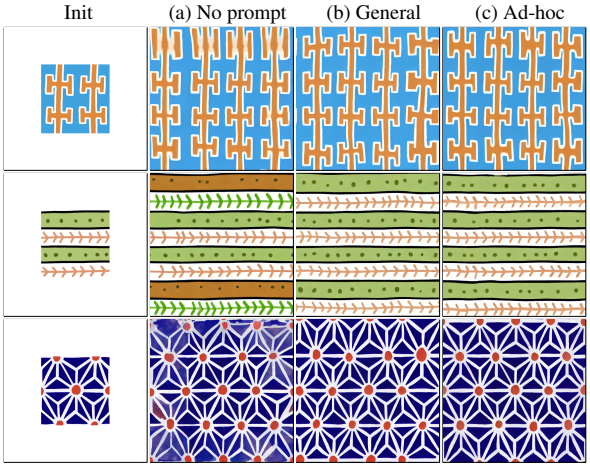

Figure 9: *Rather than tailoring a text prompt to each pattern sample (c), our general-purpose prompt (b) improves generation quality by introducing structural and coherent features that image prompting alone (a) often fails to capture.*

pansions generated from our approach compared to the other methods. These results further support our claims.

4.5. Quantitative Evaluation

We quantitatively evaluate our method by computing the TexTile metric score [RCGL24] for 35 expanded patterns. TexTile provides a differentiable metric that quantifies how likely a texture can be concatenated with itself without introducing artifacts. In our experiments, we achieve a TexTile score of $63.93\% \pm 8.24\%$, which aligns with the score obtained on tileable textures from the dataset ($62.25\% \pm 14.04\%$). However, vertical and horizontal concatenation of expanded patterns yields perfect tileability due to the noise rolling technique. Still, we recommend larger-scale expansion over replicating smaller ones to reduce detail repetition (see Fig. 11).

4.6. Ablation Study

We evaluate our design choices starting from the baseline solution and gradually introducing the different proposed architectural components and diffusion elements–IP-Adapter, LoRA finetuning, and noise rolling–. To systematically assess the impact of each component, we test the different configurations on a series of example patterns. We provide qualitative results of the ablation study in Fig. 7.

We first evaluate the Stable Diffusion base model performing a text-guided inpainting task (Fig. 7a). This sets a performance baseline without being influenced by any of the design choices presented in the paper. Although the model is able to fill in the missing areas, it tends to diverge from the input condition and break the overall structure. Even for simple examples, the text-guided approach is not a natural mean to express pattern structures such as shapes and arrangements, and moreover, it is not versatile enough to perfectly describe the design of the partial input pattern.

To provide control in a more natural way, we include an IP-Adapter [YZL*23] that introduces an image prompt as further guidance for the inpainting process. The guidance image is constructed by simply repeating the image prompt multiple times to fill a 512×512 canvas. As described in Sec. 4.2 this helps the CLIP encoder better capture the visual features and sharpness of the guidance, due to the high sensitivity of CLIP to image resolution [WCL23]. As shown in our results in Fig. 7b, visual guidance allows the model to better follow the input, while still presenting some visual inconsistencies and limitations, mostly due to the training on natural images. In fact, the model is capable of better catching the style and colors provided by the guidance image, but it still fails at reconstructing its geometrical details in both shape features or pattern scale and often provides natural-looking results.

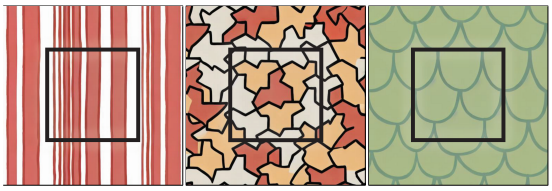

Figure 10: *Our model has both domain-specific and architectural limitations. It cannot expand non-repeating patterns, whether non-stationary or aperiodic [SMKG23] (left, center), and may also distort structured patterns when enforcing consistent scale and tileability (right).*

Since the model is more exposed to photorealistic, natural, and unstructured data during training, we perform a fine-tuning of the structured patterns to better adapt it to this new domain and task. To do so, we trained a LoRA module on our crafted pattern dataset. By combining the LoRA domain knowledge with the Stable Diffusion Model backbone, we noticed that the overall result quality and consistency are significantly improved, thanks to the new adaptation to the pattern domain. In particular, results preserve the same style as the provided input and reconstruct geometrical details and arrangements in a more resilient way (Fig. 7c).

Despite good results could be achieved on small expansions, we notice a deterioration of the output for higher expansion factors. As reported in Fig.7d, the expansion tends to produce a degraded output that influences the style and the structure, in terms of color artifacts and discontinuities in the pattern respectively. The introduction of the noise rolling technique enables us to produce results that correctly integrate the provided image by maintaining both the visual and geometrical aspects Fig.7e. In particular, this addition has a twofold effect: it makes the generated pattern tileable by removing edge discontinuities, as already assessed in [VMR*24], and it helps in better capturing long-range dependencies inside the image, thus allowing us to increase the expansion factor without losing quality.

In Fig. 9 we assess the design choice of having a general purpose text prompt as a support to the guidance image prompt. Our fixed text prompt drives the diffusion to maintain properties like high regularity in terms of both structure and colors, which are common features in the pattern domain we focus on. In our experiments, we fixed the text prompt to be *"Regular hand-drawn repeating pattern with flat colors."*. The absence of a text prompt (Fig. 9a) tends to

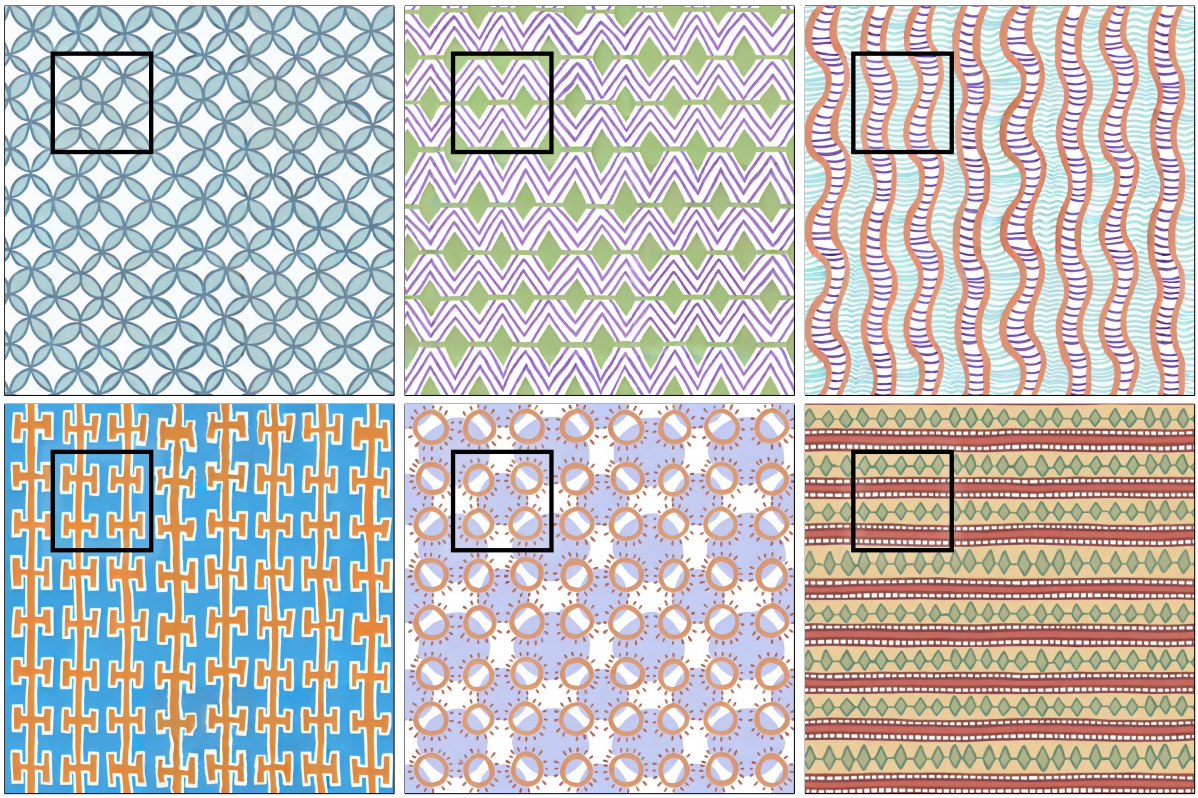

Figure 11: *Our diffusion-based pattern expansion method enables the generation of large-scale, high-quality, and tileable patterns from a small user-drawn input, reported in the black boxes. By being fine-tuned on domain-specific data, it adapts to different structured arrangements of solid-colored shapes, consistently extending the input design features to a larger-scale result. More results in Supplementary.*

include color variations and shape misalignments during the diffusion steps, deteriorating the overall coherency with the input pattern sample. In contrast, using a text prompt that is tailored to the actual input sample (Fig. 9c) does not significantly improve the generation quality while requiring an additional, non trivial, effort by the user. The ad-hoc text prompt used in the middle example is *"The pattern features two geometric motifs repeated horizontally: bright green stripes bordered by dark green with small aligned dots inside, alternating with an orange fishbone pattern on a white background. The hand-drawn style gives the design a slightly irregular appearance."*. This furtherly highlights how prompt complexity increases when attempting to describe geometric features textually. The use of the proposed general purpose text prompt represents a good tradeoff between expansion quality and pipeline generalizability (Fig. 9b).

5. Limitations and Future Work

Our method limitations can divide between architectural limitations and domain ones. Examples of failure cases or unexpected behavior are presented in Fig. 10. As discussed in the paper, our method cannot faithfully expand non-repeating patterns, either non-stationary (Fig. 10 left) or aperiodic. [SMKG23] (Fig. 10 center). This limitation comes from the design choices of our approach, which focus on repeating patterns. While both expansions present plausible patterns, they don't necessarily follow the expected behavior, where the lines in the first figure should keep growing while the tiles should not present a predictable pattern. Future work could focus on tackling non-repeating patterns by injecting, into the generation, additional information in the form of conditioning about the patterns' repetitiveness. The last failure case (Fig. 10 right), on the other hand, shows a design limitation of our approach, which can fail to generate very structured patterns at a consistent scale in the presence of tileability. This is related to the noise rolling, which enforces tileability on the border of the image, thus squeezing the border shingles to make them fit the canvas. Possible improvements could involve an automated solution to find the optimal crop of the pattern [RCGL24] before beginning the expansion.

6. Conclusion

In this paper, we present a diffusion-based architecture for structured pattern expansion, with a focus on the controllability of the generated pattern. We demonstrated the expansion of several hand-drawn patterns samples with distinctly different structures, symmetries, and appearance. Our results show the robustness of the proposed architecture and its controllability, while the comparison with prior work shows that our method is significant with respect to the state of the art.

References

[ACB17] ARJOVSKY, MARTIN, CHINTALA, SOUMITH, and BOTTOU, LÉON. "Wasserstein generative adversarial networks". *International conference on machine learning*. 2017, 214–223 3.

[BDS19] BROCK, ANDREW, DONAHUE, JEFF, and SIMONYAN, KAREN. "Large Scale GAN Training for High Fidelity Natural Image Synthesis". *International Conference on Learning Representations*. 2019 1, 3.

[CSHD03] COHEN, MICHAEL F., SHADE, JONATHAN, HILLER, STEFAN, and DEUSSEN, OLIVER. "Wang Tiles for image and texture generation". *ACM Trans. Graph.* 22.3 (July 2003), 287–294 2.

[DN21] DHARIWAL, PRAFULLA and NICHOL, ALEXANDER. "Diffusion models beat gans on image synthesis". *Advances in Neural Information Processing Systems* 34 (2021) 3.

[EF01] EFROS, ALEXEI A. and FREEMAN, WILLIAM T. "Image quilting for texture synthesis and transfer". *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '01. New York, NY, USA: Association for Computing Machinery, 2001, 341–346. ISBN: 158113374X 2.

[EL99] EFROS, A.A. and LEUNG, T.K. "Texture synthesis by non-parametric sampling". *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2. 1999, 1033–1038 vol.2 2.

[GAA*17] GULRAJANI, ISHAAN, AHMED, FARUK, ARJOVSKY, MARTIN, et al. "Improved training of wasserstein gans". *Advances in neural information processing systems* 30 (2017) 3.

[GPM*14] GOODFELLOW, IAN, POUGET-ABADIE, JEAN, MIRZA, MEHDI, et al. "Generative Adversarial Nets". *Advances in Neural Information Processing Systems*. Vol. 27. 2014 3.

[HGZ*23] HE, ZHEN, GUO, JIE, ZHANG, YAN, et al. "Text2Mat: Generating Materials from Text". *Pacific Graphics Short Papers and Posters*. 2023 1.

[HJA20] HO, JONATHAN, JAIN, AJAY, and ABBEEL, PIETER. "Denoising diffusion probabilistic models". *Advances in Neural Information Processing Systems* 33 (2020) 3, 4.

[HS22] HO, JONATHAN and SALIMANS, TIM. "Classifier-free diffusion guidance". *arXiv preprint arXiv:2207.12598* (2022) 3.

[HSW*22] HU, EDWARD J., SHEN, YELONG, WALLIS, PHILLIP, et al. "LoRA: Low-Rank Adaptation of Large Language Models". *International Conference on Learning Representations*. 2022 2, 3, 5.

[HVCB21] HEITZ, ERIC, VANHOEY, KENNETH, CHAMBON, THOMAS, and BELCOUR, LAURENT. "A sliced wasserstein loss for neural texture synthesis". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021 3, 7, 8.

[KALL18] KARRAS, TERO, AILA, TIMO, LAINE, SAMULI, and LEHTINEN, JAAKKO. "Progressive Growing of GANs for Improved Quality, Stability, and Variation". *International Conference on Learning Representations*. 2018 1, 3.

[KB14] KINGMA, DIEDERIK P. and BA, JIMMY. "Adam: A Method for Stochastic Optimization". *International Conference on Learning Representations* (2014) 6.

[KLA*20] KARRAS, TERO, LAINE, SAMULI, AITTALA, MIIKA, et al. "Analyzing and improving the image quality of stylegan". *IEEE/CVF conference on computer vision and pattern recognition*. 2020 1, 3.

[KS00] KAPLAN, CRAIG S. and SALESIN, DAVID H. "Escherization". *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. USA: ACM Press/Addison-Wesley Publishing Co., 2000, 499–510. ISBN: 1581132085 2.

[KSE*03] KWATRA, VIVEK, SCHÖDL, ARNO, ESSA, IRFAN, et al. "Graphcut textures: image and video synthesis using graph cuts". *ACM Trans. Graph.* 22.3 (July 2003), 277–286 2.

[KW14] KINGMA, DIEDERIK P. and WELLING, MAX. "Auto-Encoding Variational Bayes". *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 2014 4.

[LHW*04] LIN, WEN-CHIEH, HAYS, JAMES, WU, CHENYU, et al. "A comparison study of four texture synthesis algorithms on near-regular textures". *ACM SIGGRAPH 2004 Posters*. SIGGRAPH '04. Los Angeles, California: Association for Computing Machinery, 2004, 16. ISBN: 1581138962 3.

[LHW*06] LIN, WEN-CHIEH, HAYS, J., WU, CHENYU, et al. "Quantitative Evaluation of Near Regular Texture Synthesis Algorithms". *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 1. 2006, 427–434 3.

[LLH04] LIU, YANXI, LIN, WEN-CHIEH, and HAYS, JAMES. "Near-regular texture analysis and manipulation". *ACM Trans. Graph.* 23.3 (Aug. 2004), 368–376 3.

[LLX*01] LIANG, LIN, LIU, CE, XU, YING-QING, et al. "Real-time texture synthesis by patch-based sampling". *ACM Trans. Graph.* 20.3 (July 2001), 127–150 2.

[LPdC24] LOPES, IVAN, PIZZATI, FABIO, and de CHARETTE, RAOUL. "Material Palette: Extraction of Materials from a Single Image". *CVPR*. 2024 1, 3, 7.

[Mes18] MESCHEDER, LARS. "On the convergence properties of gan training". *arXiv preprint arXiv:1801.04406* 1 (2018) 3.

[MPPS17] METZ, LUKE, POOLE, BEN, PFAU, DAVID, and SOHL-DICKSTEIN, JASCHA. "Unrolled Generative Adversarial Networks". *International Conference on Learning Representations*. 2017 3.

[PEL*23] PODELL, DUSTIN, ENGLISH, ZION, LACEY, KYLE, et al. "SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis". *International Conference on Learning Representations*. 2023 1, 2.

[Per85] PERLIN, KEN. "An image synthesizer". *SIGGRAPH*. 1985 6.

[RBL*22] ROMBACH, ROBIN, BLATTMANN, ANDREAS, LORENZ, DOMINIK, et al. "High-resolution image synthesis with latent diffusion models". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022 1–6.

[RCGL24] RODRIGUEZ-PARDO, CARLOS, CASAS, DAN, GARCES, ELENA, and LOPEZ-MORENO, JORGE. "TexTile: A Differentiable Metric for Texture Tileability". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024 9, 10.

[RGF*20] REDDY, PRADYUMNA, GUERRERO, PAUL, FISHER, MATT, et al. "Discovering pattern structure using differentiable compositing". *ACM Trans. Graph.* 39.6 (2020) 2.

[RHE11] RECAS, DIEGO LOPEZ, HILSMANN, ANNA, and EISERT, PETER. "Near-Regular Texture Synthesis by Random Sampling and Gap Filling". *Vision, Modeling, and Visualization (2011)*. The Eurographics Association, 2011 3.

[RKH*21] RADFORD, ALEC, KIM, JONG WOOK, HALLACY, CHRIS, et al. "Learning transferable visual models from natural language supervision". *International conference on machine learning*. 2021 4.

[RLJ*23] RUIZ, NATANIEL, LI, YUANZHEN, JAMPANI, VARUN, et al. "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023 3.

[RP23] RISO, MARZIA and PELLACINI, FABIO. "pEt: Direct Manipulation of Differentiable Vector Patterns". *Eurographics Symposium on Rendering*. 2023 2.

[RSP22] RISO, MARZIA, SFORZA, DAVIDE, and PELLACINI, FABIO. "pOp: Parameter Optimization of Differentiable Vector Patterns". *Computer Graphics Forum* 41 (2022) 2.

[SME21] SONG, JIAMING, MENG, CHENLIN, and ERMON, STEFANO. "Denoising Diffusion Implicit Models". *9th International Conference on Learning Representations*. 2021 6.

[SMKG23] SMITH, DAVID, MYERS, JOSEPH SAMUEL, KAPLAN, CRAIG S., and GOODMAN-STRAUSS, CHAIM. *An aperiodic monotile*. 2023. arXiv: 2303.10798 9, 10.

[SVB*21] SCHUHMANN, CRISTOPH, VENCU, RICHARD, BEAUMONT, ROMAIN, et al. "LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs". *NEURIPS Data-Centric AI Workshop*. 2021 2.

[SWMG15] SOHL-DICKSTEIN, JASCHA, WEISS, ERIC, MAHESWARANATHAN, NIRU, and GANGULI, SURYA. "Deep unsupervised learning using nonequilibrium thermodynamics". *International Conference on Machine Learning*. 2015 3.

[TWY*20] TU, PEIHAN, WEI, LI-YI, YATANI, KOJI, et al. "Continuous curve textures". *ACM Trans. Graph.* 39.6 (2020) 2.

[TWZ22] TU, PEIHAN, WEI, LI-YI, and ZWICKER, MATTHIAS. "Clustered Vector Textures". *ACM Trans. Graph.* 41.4 (2022) 2.

[VD24] VECCHIO, GIUSEPPE and DESCHAINTRE, VALENTIN. "MatSynth: A Modern PBR Materials Dataset". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024 1.

[VMR*24] VECCHIO, GIUSEPPE, MARTIN, ROSALIE, ROULLIER, ARTHUR, et al. "ControlMat: A Controlled Generative Approach to Material Capture". *ACM Trans. Graph.* 43.5 (2024) 1, 3–5, 9.

[VSP*17] VASWANI, ASHISH, SHAZEER, NOAM, PARMAR, NIKI, et al. "Attention is all you need". *Advances in neural information processing systems* 30 (2017) 4.

[VSPS24] VECCHIO, GIUSEPPE, SORTINO, RENATO, PALAZZO, SIMONE, and SPAMPINATO, CONCETTO. "MatFuse: Controllable Material Generation with Diffusion Models". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024 1, 3, 7, 8.

[WCL23] WANG, JIANYI, CHAN, KELVIN CK, and LOY, CHEN CHANGE. "Exploring clip for assessing the look and feel of images". *AAAI Conference on Artificial Intelligence*. Vol. 37. 2023 9.

[WL00] WEI, LI-YI and LEVOY, MARC. "Fast texture synthesis using tree-structured vector quantization". *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. USA: ACM Press/Addison-Wesley Publishing Co., 2000, 479–488. ISBN: 1581132085 2.

[WZ01] WILLIAM, W.L.L. and ZENG, BING. "Fast texture synthesis by feature matching". *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*. Vol. 2. 2001, 614–617 vol.2 2.

[YLC02] YU, YUE, LUO, JIEBO, and CHEN, CHANG WEN. "Multiresolution block sampling-based method for texture synthesis". *2002 International Conference on Pattern Recognition*. Vol. 1. 2002, 239–242 vol.1 2.

[YZL*23] YE, HU, ZHANG, JUN, LIU, SIBO, et al. "IP-Adapter: Text Compatible Image Prompt Adapter for Text-to-Image Diffusion Models". *arxiv:2308.06721* (2023) 4, 5, 9.

[ZCXH23] ZHOU, YANG, CHEN, KAIJIAN, XIAO, RONGJUN, and HUANG, HUI. "Neural Texture Synthesis with Guided Correspondence". *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023 1, 3, 7, 8.

[ZRA23] ZHANG, LVMIN, RAO, ANYI, and AGRAWALA, MANEESH. "Adding conditional control to text-to-image diffusion models". *IEEE/CVF International Conference on Computer Vision*. 2023 3.

[ZXL*24] ZHOU, YANG, XIAO, RONGJUN, LISCHINSKI, DANI, et al. "Generating Non-Stationary Textures using Self-Rectification". *CVPR*. 2024 1, 3, 7, 8.

[ZZB*18] ZHOU, YANG, ZHU, ZHEN, BAI, XIANG, et al. "Non-stationary texture synthesis by adversarial expansion". *ACM Trans. Graph.* 37.4 (2018) 1, 3, 7, 8.