NATIVE GENERATION OF TILEABLE PIXEL-ART TEXTURES FROM A MATERIAL NAME AND A TARGET COLOUR

Anonymous authors
Paper under double-blind review

ABSTRACT

Game and mod textures are authored natively at tiny canvases: a 16×16 tile is not a downscaled photograph but a grid in which every pixel is a design decision, the palette holds at most a handful of slots, and the tile must stay continuous across its own wrap boundary because it is used tiled. We study the task of painting a solid-colour region of an image with a tileable pixel-art texture given only a material name and the region's colour, and we generate it natively at the target resolution rather than rendering large and downsampling. Our model decodes palette-index and luminance-rank tokens in parallel, keeps tileability by construction through a toroidal relative attention bias that adds no parameters, and conditions on palettes retrieved from a memory of artist palettes. Against the strongest render-and-downsample pipeline, annotators prefer our 16×16 tiles in 73.6% of decided pairs; on the region-colouring task the de-biased judge prefers them 60.3% (family interval excludes 0.5), and that judge's 57.3% on the name-to-texture comparison is a lower bound on the same human reading. The tiles also win the majority of judge comparisons against a same-data pixel diffusion model, score distillation, and artist-tile retrieval. Measured end to end, a tile takes 0.51 s, $28\times$ faster than the strongest render pipeline and four orders of magnitude faster than score distillation. Getting these comparisons to mean anything required repairing the rulers: we show that render-and-downsample baselines collect a structural gain purely from a larger canvas, that a widely used tiling score awards a perfect value to a flat fill, and that a pairwise judge's resolvable-rate floor is far below chance.

Keywords: pixel art generation, tileable texture synthesis, masked generative transformer, retrieval-augmented generation, color-conditioned image generation, relative position bias, evaluation of generative models.

Primary Area: generative models.

1 INTRODUCTION

A large share of the textures in voxel games, their mods and many 2D engines are drawn at 16×16. A modder who adds a new block ("mossy cobblestone", "birch planks", "copper ore") needs a tile that reads as that material, that fits the colour scheme of the object it is painted on, and that can be repeated across a wall without a visible seam, and that, over dozens of blocks, must be produced at interactive speed. We study exactly this task: given an image with a solid-colour region, a material name and the region's colour, paint the region with a tileable pixel-art texture at 16×16 (and, secondarily, 24×24 and 32×32). Figure 1 states the task and shows held-out tiles against the strongest render-and-downsample baseline.

The obvious route is to render large and shrink. A 1024px SDXL (Podell et al., 2024) brick wall is structurally sound, yet every kernel we tried turns it to mush at 16×16: the render packs about 25 structural units into 16 cells, where pixel artists place about three. Cropping the render to its dominant period repairs this (the Render pipeline, below) but inherits the generator's colours, seams and cost: every 16px tile pays for several megapixel renders. Score distillation (Binninger & Sorkine-Hornung, 2024) optimises the small image directly, at hours per tile.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We instead *generate natively*. A pixel-art tile is a small grid of indices into a palette of at most sixteen colours, so we model it as such: a masked generative transformer (Chang et al., 2022) that decodes two discrete token streams in parallel (sixteen palette slots and an $N \times N$ grid of luminance ranks) under a text condition, a colour condition and a palette retrieved from a memory of artist palettes. The transformer has no absolute positions at all; its only spatial signal is a relative attention bias defined on the torus, so the network is exactly equivariant to cyclic shifts and has no notion of where the tile's boundary lies. We call the model TRD (Toroidal Rank Decoding). Native generation is also what makes it fast: 272 tokens instead of a megapixel render, half a second per tile. The recipe is not specific to game textures: the palette–rank representation fits any palette-indexed image (sprites, icons, tile maps) and the toroidal bias any periodic one (textile and wallpaper patterns, terrain tiles).

Evaluation at this scale needs care. There is no unique right answer: independent artists drawing the same material agree on 9.8% of cells against 8.6% expected by chance. Distribution metrics reward the method that returns artist originals verbatim; a common tiling score gives a perfect value to a flat fill; and a pairwise VLM judge has to be validated against a human conclusion before it can be trusted. We therefore report every comparison with its noise floor, its judge resolvable-rate floor and a family-level confidence interval, and we call a comparison a win only when it survives those checks.

Contributions.

- **Task and method.** We formulate region-conditioned pixel-art texture generation from a material name and a target colour, and propose TRD: dual-stream discrete decoding of palette indices and luminance ranks, a parameter-light toroidal relative attention bias that makes the model cyclic-shift equivariant, and retrieval-augmented palette conditioning (§4).
- **Results.** On 272 held-out materials, TRD has the best KID and FID of all generators at 16px. Against the strongest render pipeline, annotators prefer TRD+rr in 73.6% of decided pairs; the region-colouring task is family-robust at 60.3%; the judge's 57.3% is a lower bound on that human reading, at $28\times$ the speed. The judge also prefers TRD over same-data diffusion (63.0%), retrieval (58.9%) and SD-πXL ($11/12$) (§6).
- **Efficiency.** A reranked TRD tile takes 0.51 s end to end on one A100, $28\times$ faster than the strongest pipeline and about $30{,}000\times$ faster than SD-πXL (Table 1).
- **Measurement.** To make those comparisons meaningful we repair three rulers: we show that render-and-downsample baselines collect a structural gain purely from a larger canvas, that the raw tiling score awards a perfect value to a flat fill, and that the judge's resolvable-rate floor on identical pairs is 17.8%, far below chance (§5). Parameter-matched ablations show that both the periodic bias and the palette memory are load-bearing: dropping the memory more than triples KID, and swapping the periodic bias for a same-size non-periodic encoding nearly doubles it. A pre-registered 2×2 confirms that the toroidal bias alone makes tiles seamless (§6.4).

2 Related Work

Existing pixel-art methods either start from a finished large image or distil a large latent model onto a small canvas. SD-πXL (Binninger & Sorkine-Hornung, 2024) can write a palette-constrained tile by score distillation (Poole et al., 2023) from a latent diffusion model (Rombach et al., 2022), but it has no region colour, no wrap-around and costs hours per tile. Pixelization (Gerstner et al., 2012; Wu et al., 2022), content-adaptive and perceptual downscaling (Kopf et al., 2013; Öztireli & Gross, 2015), quantization (Heckbert, 1982) and dithering (Floyd & Steinberg, 1976) convert a given image and cannot change how many structural units it contains. We generate the tile natively at the target resolution, with a region colour and a toroidal bias, in half a second.

Tileable synthesis is either unconditioned (exemplar (Efros & Leung, 1999; Efros & Freeman, 2001; Gatys et al., 2015) and procedural (Perlin, 1985; Worley, 1996) methods) or enforces wrap only at sampling time on canvases of ≥ 256px, by circular padding, rolled noise or a tiling constraint (Zhou et al., 2022; Vecchio et al., 2024; Madar & Fried, 2025). TexTile (Rodriguez-Pardo et al., 2024) scores tileability but does not generate. None of these put periodicity into the model at 16px. We put wrap into a parameter-free relative attention bias so every sample is shift-equivariant, and we use the circular-padding recipe only as PixelDiff, a same-data opponent.

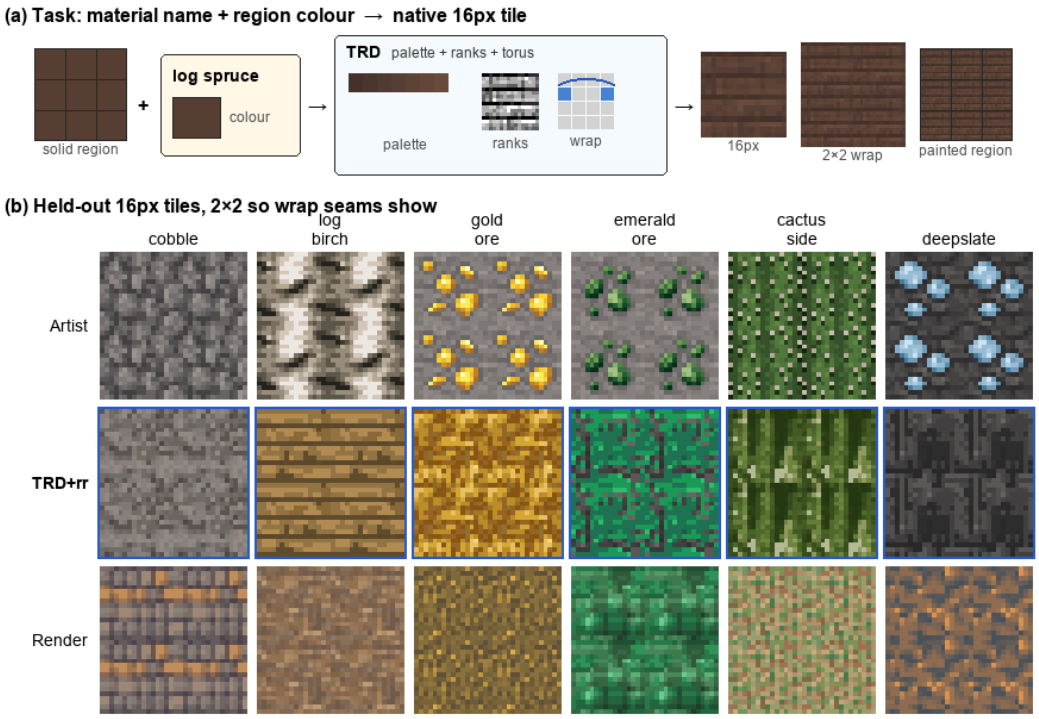


Figure 1: **(a)** The task. A material name and the region's colour condition TRD, which decodes a retrieved palette and an $N \times N$ luminance-rank grid under a toroidal relative bias and writes the tile into the region. **(b)** Held-out 16px name-to-texture samples (2×2 so wrap seams are visible), chosen for variety from the materials on which the judge preferred TRD+rr over both Render and Retrieve. Unselected sheets are in Appendix E.

Colour prompts (Butt et al., 2024), adapters (Zhang et al., 2023) and retrieval-augmented diffusion (Blattmann et al., 2022; Sheynin et al., 2023; Chen et al., 2023) condition large generators by retrieving or steering whole images, which leaks training pixels and does not separate colour from structure. We retrieve only palettes and feed them as known tokens, so the grid is always generated.

Masked transformers (Chang et al., 2022; 2023; Lezama et al., 2022; Li et al., 2023; Weber et al., 2024) decode VQ tokens (van den Oord et al., 2017) of natural images, a form of discrete diffusion (Austin et al., 2021). Pixel art is already discrete, so a tokenizer is a detour; standard relative position biases (Shaw et al., 2018; Raffel et al., 2020; Liu et al., 2021; Wu et al., 2021) are not periodic, so the tile edge is a privileged location. We drop the tokenizer, decode palette slots and luminance ranks, and restrict the relative bias to the torus (Cohen & Welling, 2016).

Pairwise VLM judges (Zheng et al., 2023) have positional bias (Wang et al., 2024), order-swap inconsistency (Li et al., 2026) and unreliable absolute scores (Kumar et al., 2026); FID, KID and DINOv2 (Heusel et al., 2017; Bińkowski et al., 2018; Stein et al., 2023; Oquab et al., 2024) were calibrated on natural images at $\geq$ 224px, and a flat fill can ace a tiling score. We certify the judge against a human conclusion, ask both orders, measure an empty-pair floor, use family-level intervals, and refuse the raw tiling score.

3 Why a 16×16 Tile Is Not a Small Image

Data. We collected every texture pack with a clean licence and an explicit resolution tag from the Luanti (Minetest) ContentDB: 105 packs were scanned, packs whose licence forbids training were excluded, and Minecraft-derived packs were excluded entirely. Because the engine uses a shared file name for each block across packs (e.g. `default_wood.png`), file names are free, artist-aligned

material labels. After dropping tiles with fewer than three native colours the corpus has 5,979 tiles from 76 packs covering 521 materials (4,951 at 16px, 679 at 32px, 349 at 64px). The split is *by pack*: 57 packs for training (4,248 tiles), 7 for validation (460) and 12 for testing (1,271). Because a pack has one visual style, a pack-level split means every test tile is drawn by an artist the model has never seen, while 448 of the 450 test materials occur in training under other artists.

Every pixel is a decision. In the 16px subset the median native colour count is 16 and 2,172 of 4,951 tiles use a full sixteen-slot palette; there is no anti-aliasing, and edges are hard. Dithering, often cited as the characteristic difficulty, is rare: a 50% checkerboard signature appears *less* often than under a within-tile shuffle null, and tiles are dominated by coherent stripes and blocks.

There is no right answer. Because a tile is a grid of palette indices, agreement between two artists who drew the same material is exactly computable. Over 50,753 same-material artist pairs, per-cell agreement is 0.098 against 0.086 under a paired shuffle null that preserves each tile's own marginal. A per-pixel reconstruction target is therefore meaningless for this task, which is why we model a distribution and why evaluation needs the care of §5.

Why not render and shrink? Artist tiles hold a median of 3.2 structural units at both 16 and 32px; a 1024px SDXL render holds about 25, so it must be cropped before any kernel can help (Figure 3, appendix).

4 METHOD

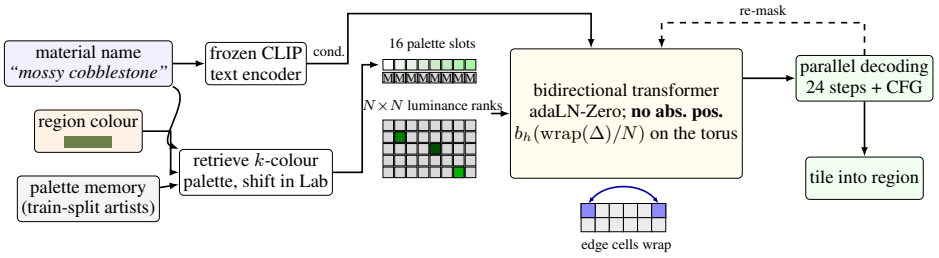


Figure 2: Architecture. Name and colour become a text condition and a retrieved artist palette; palette slots and luminance ranks are unmasked in parallel. The only spatial signal is a relative bias on wrapped offsets.

4.1 A DISCRETE, COLOUR-FACTORED REPRESENTATION

Each training tile $x \in \{\mathrm{RGB}\}^{N\times N}$ is quantised to at most $K = 16$ colours by median cut and *canonicalised*: the $k \le K$ colours it uses are sorted by luminance, the grid stores each pixel's luminance rank $r_{ij} \in \{0, \dots, k-1\}$, and the palette stores the colours $(c_0, \dots, c_{k-1})$ in the same order as codes in a 512-entry colour codebook. The mapping is exactly invertible. A tile is thus the token sequence

$$s = \big[\, c_0, \dots, c_{k-1}, \underbrace{\texttt{PAD}, \dots}_{K-k} \;\big|\; r_{11}, \dots, r_{NN} \,\big], \qquad |s| = K + N^2, \tag{1}$$

with $|s| = 272$ at 16px. The factorisation separates *structure* (which cell is darker than which) from *colour* (what the levels are), which is what lets the palette be retrieved or shifted to a user's colour without touching the structure, and it matches how pixel artists work: a ramp of shades, placed by hand. Because a rank token means different things for different k (rank 2 is the brightest level of a 3-colour tile and a dark one of a 16-colour tile), each grid token additionally receives an embedding of its normalised level $\mathrm{round}(15\,r/(k-1))$, and logits for ranks $\ge k$ are masked.

4.2 TOROIDAL RELATIVE ATTENTION BIAS

The backbone is a bidirectional transformer (Vaswani et al., 2017) of 12 blocks, width 384 and 6 heads. It has *no absolute position embedding*. Instead every block adds to the attention logits a bias that depends only on the offset between two grid cells taken on the torus. For cells at $p_i, p_j \in \{0, \dots, N-1\}^2$ let $u = \mathrm{wrap}\big((p_i - p_j)/N\big) \in [-\frac{1}{2}, \frac{1}{2})^2$ with $\mathrm{wrap}(t) = ((t + \frac{1}{2}) \bmod 1) - \frac{1}{2}$ applied per axis. The bias is

$$b_h(i,j) = \mathrm{MLP}_h\big(\phi(u)\big), \qquad \phi(u) = \big[\sin 2\pi f u_y,\ \cos 2\pi f u_y,\ \sin 2\pi f u_x,\ \cos 2\pi f u_x\big]_{f=1}^{F}, \quad (2)$$

with $F = 8$ harmonics, which suffices to represent any function on a 16-cell circle (Tancik et al., 2020). Palette tokens have no position; the palette–palette block of the bias is a learned $K \times K$ table per head and the palette–grid blocks are learned scalars. One bias module is shared by all blocks; with a 128-unit hidden layer it has about 6.5k parameters, fewer than a single learned absolute embedding of the 16×16 grid (98k). Its toroidal form costs nothing over an ordinary relative bias of the same shape: the wrap is applied to the input, not learned.

Proposition 1. *Let T_δ cyclically shift the grid tokens of s by $\delta \in \mathbb{Z}_N^2$ and leave palette tokens fixed. If the network uses no absolute positions and its attention bias between grid tokens depends only on $\mathrm{wrap}((p_i - p_j)/N)$, then its logits satisfy $f(T_\delta s) = T_\delta f(s)$ for all δ.*

Proof sketch. Token-wise operations (embeddings, MLPs, adaLN) commute with any permutation of tokens. Attention logits $q_i^\top k_j + b(i,j)$ are permuted by T_δ because $\mathrm{wrap}((p_i + \delta - p_j - \delta)/N) = \mathrm{wrap}((p_i - p_j)/N)$, and the palette–grid terms are position-independent. □The trained model satisfies this to float32 precision: over five shifts at 16 and 32px, grid logits of the shifted input differ from the shifted logits by at most 1.1×10^{-5} (logit s.d. 1.1), whereas the non-periodic and wrong-period variants of §6.4 deviate by 1.4–5.1.

Because masked decoding samples positions with i.i.d. Gumbel perturbations, the sampling distribution inherits the invariance: the model assigns the same probability to a tile and to every cyclic shift of it, so the tile boundary is not a privileged location. This is the precise sense in which tileability is built in. Whether the *periodic* form of the bias is what makes the model work is a separate, empirical question, which we test with parameter-matched replacements in §6.4.

4.3 CONDITIONING AND RETRIEVAL-AUGMENTED PALETTES

Text, colour and palette size. The material name is embedded as "pixel art texture of {name}" by a frozen CLIP text encoder (Radford et al., 2021), which makes the vocabulary open. The region colour enters as $[r, g, b, 1]$ (all zeros when dropped; the flag distinguishes "no colour" from black) and the palette size k as a learned embedding. The three are summed into one vector that modulates every block through adaLN-Zero (Peebles & Xie, 2023). During training the text is dropped for 10% of examples to enable classifier-free guidance (Ho & Salimans, 2021), and the colour is dropped for half of them.

Palette memory. A 512-way palette head trained on a few thousand tiles learns the marginal of colours but not the ramps artists actually use. We therefore keep a memory of every training-split palette with its material, CLIP text embedding, colour count and mean colour, and at inference retrieve instead of generate. For a query name we take the n_{name} nearest training materials in CLIP text space, rank their tiles by cross-modal CLIP similarity between the query text and the tile image, optionally restrict to palettes whose mean colour is close to the target in Lab (ΔE), and sample one of the top 5. The retrieved palette is shifted in Lab so its mean lands on the region colour, and fed to TRD as *known* palette tokens; only the grid is generated. The memory contains palettes, not images, and is built from training packs only, so no test artist's work can be retrieved.

Structural exemplars. Four same-material training tiles by other artists can be appended as position-free, never-predicted tokens; they are optional, and the model is robust to their removal (§6.4).

Table 1: Cost per delivered 16px tile on one A100 (wall clock, exact test-time settings). TRD+rr: all 272 test materials end to end, *including* model loading (138.6 s total). Render: per material, *excluding* model loading. SD-πXL: median over its test subset.

Method	Params	Generated at	Sequential steps	Time / tile	Relative
SD-πXL	2.6B (SDXL)	16^2, optimised	10k optimisation	4.44 h	$\approx 31{,}000\times$
Render	2.6B (SDXL)	4×1024^2	4×28 denoising	14.5 s	$28\times$
TRD+rr (ours)	33M	4×272 tokens	24 parallel	**0.51** s	$1\times$

4.4 Training and parallel decoding

Objective. We train with the MaskGIT objective: a mask ratio $\gamma = \cos(\frac{\pi}{2}u)$, $u \sim \mathcal{U}(0,1)$, is drawn per example, palette and grid tokens are masked independently at that ratio, and the loss is cross-entropy on masked tokens of both streams. Every example is cyclically shifted by a uniformly random offset (all N^2 phases are valid samples of a tileable texture) and horizontally flipped with probability $\frac{1}{2}$. We use AdamW with learning rate 3×10^{-4}, weight decay 0.05, 1,000 warm-up steps and batch 256; the 16px model is trained for 20k steps, and a second checkpoint trained jointly on 16 and 32px is used for the 24 and 32px results. At roughly 0.3 s per step on one A100, a model trains in under two GPU-hours.

Decoding. Sampling starts from a fully masked grid (and the retrieved palette) and runs 24 steps. At each step all masked tokens are sampled in parallel, and the least confident are re-masked according to the cosine schedule, with Gumbel noise on the confidences annealed to zero (choice temperature 20) and classifier-free guidance of weight 1.5. As an optional last stage ("+rr") we draw four samples and keep the one whose CLIP ViT-B/16 image embedding best matches the prompt; evaluation uses a different CLIP (ViT-B/32), so the reranker does not grade its own homework. The tile is then written into the region with its phase aligned to the region's bounding box.

Speed. Table 1 times the exact test-time configurations on one A100. Existing routes are slow for structural reasons: a render pipeline runs a 2.6B-parameter U-Net 28 times on a 128^2 latent per render, four renders per tile, then discards over 99.9% of the pixels; score distillation back-propagates through that network at each of 10k steps per tile. TRD removes each cost by construction: it generates at the target resolution (272 tokens), its discrete tokens *are* the tile (no latent decoder), decoding is parallel (24 steps), palettes are looked up, and the toroidal bias needs no crop, period search or seam repair.

5 Evaluation Protocol: Repairing the Rulers

We report the metrics used by the pixel-art and texture literature and, for each, the evidence a reader needs to know how large a difference has to be.

Distribution and alignment metrics. KID (Bińkowski et al., 2018) ($\times 10^3$), FID (Heusel et al., 2017) with Inception features (Szegedy et al., 2016) and FD-DINOv2 (Stein et al., 2023) against the test-split artist tiles, CLIP score (Hessel et al., 2021) against "pixel art texture of {name}", and LPIPS (Zhang et al., 2018) diversity between samples of the same prompt. Tiles are enlarged with nearest neighbour to the network input size, identically for artists and methods. Two floors calibrate these numbers. Retraining the same recipe with three seeds moves 16px KID by at most 1.09; resampling one checkpoint once per material moves it by up to 4.76 (2.70 with four samples per material), and FID, FD and CLIP by 6.0, 20 and 0.41. We only interpret a difference that exceeds the larger of the two floors.

Ruler 1: a flat fill has a perfect tiling score. A common tiling score sums colour differences across the wrap seam of a 3×3 tiling. We ran it on constructed counterexamples. A tile filled with its own mean colour scores exactly 0.000, better than every real method (the best, LoRA, scores 5.31) and far better than artist tiles (19.9). Blurring or flattening the border likewise improves the raw score. We therefore report the ratio of across-seam to interior neighbour differences, whose ideal is 1: it

returns *undefined* for a flat fill and 0 for a flattened border, so neither can win, and deviations on both sides count as failures. Even this ratio is a property check, not a quality axis: a cyclic shift of the artist tiles changes their median $|r - 1|$ from 0.469 to 0.234.

Ruler 2: a larger canvas buys structure for free. Render crops to 4.5 dominant periods before downsampling, independently of the target size, so at 32px it shows the same structure with twice the cells. Asked to compare Render at 32px with Render at 16px enlarged to the same size, the judge prefers the former $107/138 = 78\%$ ($p = 5.5 \times 10^{-11}$, family CI $[0.69, 0.86]$). The render-and-downsample family therefore improves with the canvas for reasons unrelated to the method, and a curve of win rate against resolution mixes that gain with the method's quality. We compare methods only within a resolution and refuse to read the three resolutions as one curve.

Ruler 3: human ground truth, scaled by a certified judge. Preference is anchored in human judgement. Four annotators labelled a blind study of 60 pairs from an earlier generator-versus-downsample comparison (not the main TRD arms), sides randomised and resolved to a consensus label: one condition was preferred in $49/59 = 83\%$ of decided pairs. To extend this protocol to all 272 test materials and every arm, we admit a pairwise VLM judge only if it reproduces that conclusion: of three frontier judges, only `claude-opus-5` did (73%, $p = 0.008$); another, with identical per-item agreement (66.7%), compressed it to 62% ($p = 0.14$), and the third inverted it. The judge is thus a scaled-up proxy certified against the annotators, not an independent authority. A later human study on the main TRD+rr vs. Render comparison (§6) prefers ours more strongly than this judge does, so the judge is the conservative instrument. Each pair is asked in both orders and a verdict counts only if both orders agree. To measure what this protocol can resolve, we submit 118 *empty* pairs: the same image on both sides. An ideal judge never resolves them; one that answered at random would resolve 50%. The judge resolves $21/118 = 17.8\%$, far below chance, whereas real comparisons are resolved 74–78% of the time, so decided verdicts carry signal. We report wins over decided pairs, a two-sided binomial p, and a *family-level* 95% interval obtained by resampling material families, which accounts for the fact that related materials (all the "planks") are not independent (Cameron & Miller, 2015). A comparison is called robust only if that interval excludes 0.5.

6 Experiments

6.1 Setup and baselines

The formal test set (E_mat) has 272 materials with 857 artist tiles from the 12 held-out packs; it was generated once, under a pre-registered protocol, after all design choices were frozen on the validation packs. Each baseline blocks one objection. **Shrink** ("just shrink a big model"): SDXL at 1024px, box downsample, quantise to 16 colours. **Render** (the industrial answer): our own render pipeline, best of four SDXL renders, crop to the detected dominant period, seam-aligned window, then downsample and quantise (Appendix, Figure 3). **SD-πXL** (a published pixel-art method): the released code, on 12 materials, as each tile takes over four hours (Table 1). **LoRA** ("add a pixel-art LoRA"): SDXL with the public `pixel-art-xl` LoRA (Hu et al., 2022), then Shrink's downsampling. **Retrieve** ("retrieval has the best FID"): an artist tile retrieved from the training split by prompt co-occurrence. **PixelDiff** ("native generation is enough" and "why not circular padding"): a 29.9M-parameter pixel-space diffusion U-Net (Ronneberger et al., 2015; Ho et al., 2020) trained from scratch on *our* data with text and colour conditioning, circular padding, classifier-free guidance, 50 DDIM steps and 16-colour quantisation; it removes the data advantage as an explanation. TRD denotes the single-sample model and TRD+rr the four-to-one reranked variant.

6.2 Main results at 16px

Distribution metrics. Table 2 shows that TRD has the lowest KID and FID of all generators. Against the SDXL family the margins are far outside the noise floor: KID drops from 28.9 (Render) and 52.2 (Shrink) to 8.8, FID from 65.4 to 43.2, and FD-DINOv2 from 217 to 82. Reranking brings CLIP level with Render (35.19 vs 35.21, well within the 0.41 floor) at a KID that is still three times lower. Against PixelDiff, trained on the same data, TRD has lower KID (8.8 vs 10.0) and FID (43.2 vs 45.5) and is clearly better aligned with the material name (CLIP $+1.43$, beyond the floor); where

Table 2: Name-to-texture at 16px on the held-out test set (272 materials, 857 artist tiles, unseen artists). KID is $\times 10^3$. Seam: median $|r-1|$ of the seam ratio (ideal 0, a property check only). Retrieve returns artist originals and is listed as a reference, not a generator. Noise floors: KID 4.76, FID 6.0, FD 20, CLIP 0.41.

Method	KID ↓	FID ↓	FD-DINOv2 ↓	CLIP ↑	LPIPS div. ↑	Seam $\lvert r-1\rvert$
Shrink	52.2	85.9	206.8	34.51	0.217	0.173
Render	28.9	65.4	217.4	**35.21**	–	0.530
LoRA	76.7	104.6	253.7	34.97	0.138	0.190
PixelDiff	10.0	45.5	**59.3**	33.17	**0.355**	0.232
TRD (ours)	**8.8**	**43.2**	82.3	34.60	0.303	0.201
TRD+rr (ours)	9.7	44.2	77.5	35.19	–	0.178
Retrieve (reference)	2.7	43.0	45.3	32.95	0.382	0.614

Table 3: Pairwise preference at 16px. Judge: `claude-opus-5`, both orders, inconsistent pairs discarded; family CI unless noted. Human vs Render: Jeffreys (App. C). **robust** = excludes 0.5. Identical-pair floor 17.8%. Colour task: same Lab residual recolouring on every method.

Task	Comparison (first wins)	Wins / decided	Rate	p	95% CI
Name→texture	annotators: TRD+rr vs Render	67 / 91	73.6%	7.4×10^{-6}	**[0.64, 0.82]**
	TRD vs PixelDiff	131 / 208	63.0%	2.2×10^{-4}	**[0.540, 0.719]**
	TRD+rr vs Retrieve	119 / 202	58.9%	0.014	**[0.502, 0.668]**
	TRD vs SD-πXL (12 mat.)	11 / 12	91.7%	0.0064	n/a
	judge: TRD+rr vs Render	114 / 199	57.3%	0.047	[0.500, 0.641]
Region colour	TRD vs Render	114 / 189	60.3%	0.0056	**[0.525, 0.680]**
	TRD vs PixelDiff	117 / 193	60.6%	0.0039	**[0.524, 0.674]**
	TRD vs Retrieve (+recolour)	111 / 208	53.4%	0.37	[0.434, 0.627]

the metrics split, the judge decides (63.0%, below). Retrieve dominates every distribution metric simply by returning artist originals; the next paragraph shows that this does not make it better.

Pairwise judgements. Table 3 reports the main 16px arms (forest and remaining arms: App. D; samples: App. E and Figure 1). The claim we stand on against Render is the human one (App. C): four retained annotators prefer TRD+rr in $67/91 = 73.6\%$ of decided pairs ($p = 7.4 \times 10^{-6}$, Jeffreys $[0.64, 0.82]$; per person 54.5%, 78.3%, 78.3%, 82.6%; $7/10$ shared materials unanimous). The region-colouring task is family-robust at 60.3% ($[0.525, 0.680]$). The judge's name-to-texture reading is 57.3% (family CI $[0.500, 0.641]$), a lower bound on that human comparison, at $28\times$ the speed (Table 1). TRD is preferred over PixelDiff in 63.0% of decided pairs (family CI $[0.54, 0.72]$), over Retrieve in 58.9% ($[0.50, 0.67]$) despite Retrieve's best KID/FID/FD, and over SD-πXL on $11/12$ materials ($p = 0.0064$).

6.3 The region-colouring task

The full task paints a region of a given colour. Here every method receives the target colour: TRD through its colour condition and colour-filtered retrieval, the baselines through recolouring. To keep the comparison fair, *every* method, ours included, then receives the same Lab residual shift that moves its mean colour onto the target. The shift makes colour error negligible for all methods ($\Delta E \leq 0.32$; Table 8, appendix), so the comparison is about texture quality under a colour constraint. Natively, without the shift, TRD already lands within $\Delta E = 3.05$ of the target, close to the just-noticeable range (Sharma et al., 2005), whereas recolouring the SDXL renders by palette mapping leaves $\Delta E \approx 15$. TRD has the best FID and CLIP score of all generators on this task, and the judge prefers it over Render (60.3%) and PixelDiff (60.6%), both robust (Table 3). Against recoloured artist originals (Retrieve), a generator is being compared with the artists themselves; TRD holds its own (53.4%) while producing new tiles rather than copies.

Table 4: Ablations at 16px. Top: variants of TRD+rr on the test set (KID $\times 10^3$; judge = full model vs. variant). Bottom: toroidal-bias arms retrained from a common checkpoint, validation materials (125); seam = median per-material $|r-1|$, lower is closer to ideal. Floors: KID 4.76, FID 6.0, CLIP 0.41.

Variant	KID	FID	CLIP	Seam	Judge (full wins)
TRD+rr (full)	9.7	44.2	35.19	–	–
– palette memory (generate palette)	35.1	68.2	34.42	–	**126/190 = 66.3%**
– structural exemplars	10.3	44.2	35.18	–	87/175 = 49.7%
choice temperature $20 \to 4.5$	10.8	44.8	35.09	–	86/154 = 55.8%
– 4→1 reranking	12.0	44.9	34.66	–	–
C: toroidal harmonic bias (control)	8.4	–	–	0.225	–
A1: bias zeroed	26.9	–	–	0.093	–
A2: non-periodic features, same #params	15.9	–	–	0.148	–
A3: wrap at wrong period	7.3	–	–	0.219	–

6.4 Ablations

All ablations are read against the noise floors of §5; Table 4 collects them.

Artist palettes are the colour interface. Pixel artists pick a ramp first and then place light and dark. Replacing the retrieved palette by the model's own palette head raises KID from 9.7 to 35.1 and FID from 44.2 to 68.2 and lowers CLIP by 0.76, the only ablation that clears all three floors, and the judge prefers the full model in 66.3% of pairs ($p = 8 \times 10^{-6}$, family CI $[0.57, 0.75]$). That is the intended split: colour comes from artist ramps, structure is generated. With artist palettes substituted, the grid stream alone reaches KID 4.5 against an artist-versus-artist floor of 3.8.

The periodic bias is what makes a position-free model work. We ablated the bias in three parameter-matched ways, each retrained for 12k steps from a common checkpoint with the same recipe and seed on the validation materials: (A1) zero the grid–grid bias, leaving the model with no spatial signal at all; (A2) replace the harmonic features of Eq. 2 by non-periodic power features of identical dimension, a drop-in with exactly the same parameter count; (A3) wrap at a wrong period ($1.3N$). Quality collapses without the periodic form: KID rises from 8.4 to 26.9 without the bias and to 15.9 with the same-size non-periodic encoding, both far above the 4.76 floor, while the wrong-period variant, still periodic, is indistinguishable (7.3). The periodic bias is what lets the model express "a mortar line every four rows" without absolute coordinates, and it does so while keeping the exact shift equivariance of §4.2. (A1's low seam deviation reflects a structureless bag of tokens, not better tiling.) A pre-registered 2×2 trained from scratch (periodic or non-periodic bias $\times$ with or without random cyclic-shift augmentation, 20k steps each) shows that this equivariance delivers tileability by itself: with the toroidal bias, removing the augmentation leaves seam deviation unchanged (0.173 vs 0.185, $p = 0.72$), whereas removing both raises it to 0.249 (worse on $81/125$ materials, $p = 0.001$), at no cost in KID. The remaining components (exemplars, judge 49.7%; choice temperature 55.8%; reranking, ΔKID 2.3) stay within noise, so the gains come from the two design choices above.

6.5 Other resolutions

At 24px (unseen in training; App. D), TRD+rr matches Render (50.0%) and beats PixelDiff (69.2%, robust); at 32px it beats PixelDiff (61.7%, robust) and SD-πXL ($10/12$), while Render is preferred 58.9% ($p = 0.014$), the canvas gain of Ruler 2.

7 Conclusion

Pixel-art textures are small, discrete and toroidal, and TRD is built to match. At 16px annotators prefer it over the strongest render pipeline in 73.6% of decided pairs at $28\times$ the speed; the colour task is family-robust at 60.3%, and the judge's 57.3% is a lower bound. The same recipe applies to sprites, tile maps and textiles.

AI USE STATEMENT

Large language models were used as coding assistants for experiment scripts and analysis code, to help draft and edit the text of this paper, and as the pairwise judge described in §5, whose selection and protocol are reported there. All experimental designs, numbers and claims were checked by the authors against the logged outputs.

REPRODUCIBILITY STATEMENT

The dataset construction and split are described in §3, the architecture and training recipe in §4 and Appendix A, and the evaluation protocol, floors and judge procedure in §5. Code for training, sampling, all baselines, metrics and the judge protocol, together with the per-arm judge outputs, will be released with the paper.

REFERENCES

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, 2021.

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018.

Alexandre Binninger and Olga Sorkine-Hornung. SD-πXL: Generating low-resolution quantized imagery via score distillation. In *SIGGRAPH Asia 2024 Conference Papers*, 2024. doi: 10.1145/3680528.3687570.

Andreas Blattmann, Robin Rombach, Kaan Oktay, Jonas Müller, and Björn Ommer. Retrieval-augmented diffusion models. In *Advances in Neural Information Processing Systems*, 2022.

Muhammad Atif Butt, Kai Wang, Javier Vazquez-Corral, and Joost van de Weijer. ColorPeel: Color prompt learning with diffusion models via color and shape disentanglement. In *European Conference on Computer Vision*, 2024.

A. Colin Cameron and Douglas L. Miller. A practitioner's guide to cluster-robust inference. *Journal of Human Resources*, 50(2):317–372, 2015.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. In *International Conference on Machine Learning*, 2023.

Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W. Cohen. Re-imagen: Retrieval-augmented text-to-image generator. In *International Conference on Learning Representations*, 2023.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning*, 2016.

Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of SIGGRAPH*, pp. 341–346, 2001.

Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision*, 1999.

Robert W. Floyd and Louis Steinberg. An adaptive algorithm for spatial greyscale. *Proceedings of the Society for Information Display*, 17(2):75–77, 1976.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2015.

Timothy Gerstner, Doug DeCarlo, Marc Alexa, Adam Finkelstein, Yotam Gingold, and Andrew Nealen. Pixelated image abstraction. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering (NPAR)*, 2012.

Paul Heckbert. Color image quantization for frame buffer display. In *Proceedings of SIGGRAPH*, pp. 297–307, 1982.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of EMNLP*, 2021.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Johannes Kopf, Ariel Shamir, and Pieter Peers. Content-adaptive image downscaling. *ACM Transactions on Graphics*, 32(6), 2013. doi: 10.1145/2508363.2508370.

Divake Kumar, Sina Tayebati, Devashri Naik, Ranganath Krishnan, and Amit Ranjan Trivedi. VLM judges can rank but cannot score: Task-dependent uncertainty in multimodal evaluation. *arXiv preprint arXiv:2604.25235*, 2026.

José Lezama, Huiwen Chang, Lu Jiang, and Irfan Essa. Improved masked image generation with token-critic. In *European Conference on Computer Vision*, 2022.

Chunyang Li, Yilun Zheng, Xinting Huang, Tianqing Fang, Jiahao Xu, Lihui Chen, Yangqiu Song, and Han Hu. WebDevJudge: Evaluating (M)LLMs as critiques for web development quality. In *International Conference on Learning Representations*, 2026.

Tianhong Li, Huiwen Chang, Shlok Kumar Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. MAGE: MAsked generative encoder to unify representation learning and image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision*, 2021.

Or Madar and Ohad Fried. Tiled diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

A. Cengiz Öztireli and Markus Gross. Perceptually based downscaling of images. *ACM Transactions on Graphics*, 34(4), 2015. doi: 10.1145/2766891.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Ken Perlin. An image synthesizer. *ACM SIGGRAPH Computer Graphics*, 19(3):287–296, 1985.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, 2024.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3d using 2D diffusion. In *International Conference on Learning Representations*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Carlos Rodriguez-Pardo, Dan Casas, Elena Garces, and Jorge Lopez-Moreno. TexTile: A differentiable metric for texture tileability. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, 2015.

Gaurav Sharma, Wencheng Wu, and Edul N. Dalal. The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*, 30(1):21–30, 2005.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of NAACL-HLT*, 2018.

Shelly Sheynin, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. kNN-Diffusion: Image generation via large-scale retrieval. In *International Conference on Learning Representations*, 2023.

George Stein, Jesse C. Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L. Caterini, J. Eric T. Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Advances in Neural Information Processing Systems*, 2023.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, 2020.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Giuseppe Vecchio, Rosalie Martin, Arthur Roullier, Adrien Kaiser, Romain Rouffet, Valentin Deschaintre, and Tamy Boubekeur. ControlMat: A controlled generative approach to material capture. *ACM Transactions on Graphics*, 43(5), 2024. doi: 10.1145/3688830.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Mark Weber, Lijun Yu, Qihang Yu, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. MaskBit: Embedding-free image generation via bit tokens. *Transactions on Machine Learning Research*, 2024.

Steven Worley. A cellular texture basis function. In *Proceedings of SIGGRAPH*, pp. 291–294, 1996.

Kan Wu, Houwen Peng, Minghao Chen, Jianlong Fu, and Hongyang Chao. Rethinking and improving relative position encoding for vision transformer. In *IEEE/CVF International Conference on Computer Vision*, 2021.

Zongwei Wu, Liangyu Chai, Nanxuan Zhao, Bailin Deng, Yongtuo Liu, Qiang Wen, Junle Wang, and Shengfeng He. Make your own sprites: Aliasing-aware and cell-controllable pixelization. *ACM Transactions on Graphics*, 41(6), 2022. doi: 10.1145/3550454.3555482.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, 2023.

Xilong Zhou, Miloš Hašan, Valentin Deschaintre, Paul Guerrero, Kalyan Sunkavalli, and Nima Khademi Kalantari. TileGen: Tileable, controllable material generation and capture. In *SIGGRAPH Asia 2022 Conference Papers*, 2022.

A Implementation Details

Table 5: TRD hyper-parameters (16px model).

Backbone	12 blocks, width 384, 6 heads, MLP ratio 4, dropout 0.1, adaLN-Zero
Parameters	33M (bias module ≈6.5k)
Tokens	16 palette slots (512-code colour codebook) + N^2 rank tokens
Toroidal bias	$F = 8$ harmonics per axis, MLP hidden 128, shared across blocks
Text encoder	frozen CLIP ViT-B/32, prompt "pixel art texture of {name}"
Training	AdamW, lr 3×10^{-4}, wd 0.05, warm-up 1k, batch 256, 20k steps
Augmentation	uniform cyclic shift (all N^2 phases), horizontal flip
Masking	cosine schedule, palette and grid masked at the same ratio
Decoding	24 steps, choice temperature 20, CFG 1.5, palette nucleus 0.9
Retrieval	$n_{\text{name}} = 30$ (100 for +rr), top-5, Lab mean shift
Reranking (+rr)	4 samples, CLIP ViT-B/16 image–text similarity

Judge prompt and bookkeeping. Each pair is shown as two nearest-neighbour enlargements of equal size, with the material name, and the judge is asked which is the better pixel-art texture of that material. Every pair is asked twice with positions swapped; API failures void an arm rather than being dropped silently. Family intervals resample families of related material names; an arm is additionally checked by leaving out one family at a time, and arms with too few families (the SD-πXL subset) receive no family interval.

B THE RENDER-AND-DOWNSAMPLE BASELINE

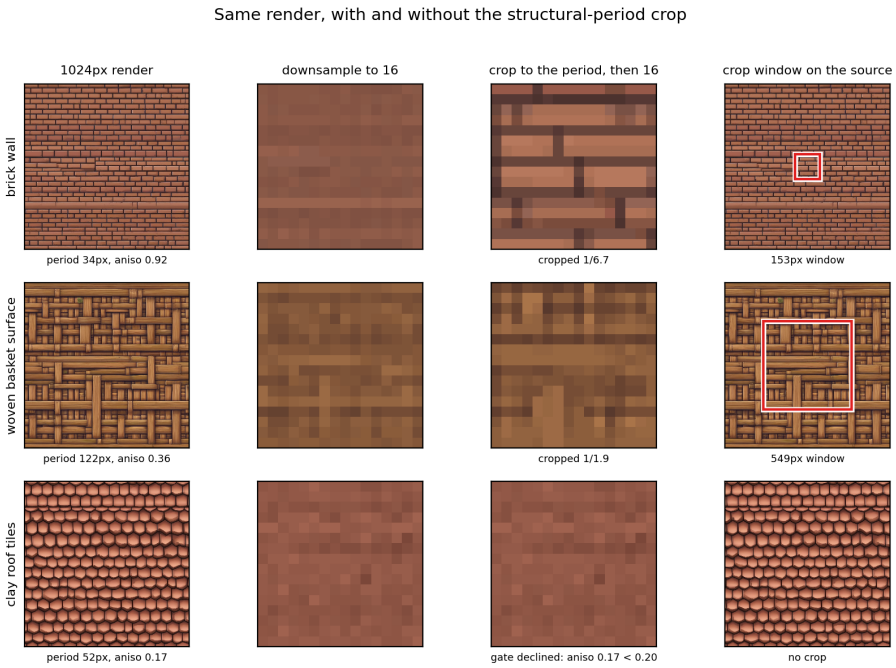


Figure 3: The strongest render-and-downsample baseline (Render). A 1024px render contains far more structural units than a 16px tile can hold, so plain downsampling (second column) loses the structure. Cropping to the detected period first (third column) repairs it when a period exists; an anisotropy gate declines to crop isotropic materials (bottom row). Because the crop is a fixed multiple of the period, the same window yields more detail at a larger target size, which is the canvas gain of §5.

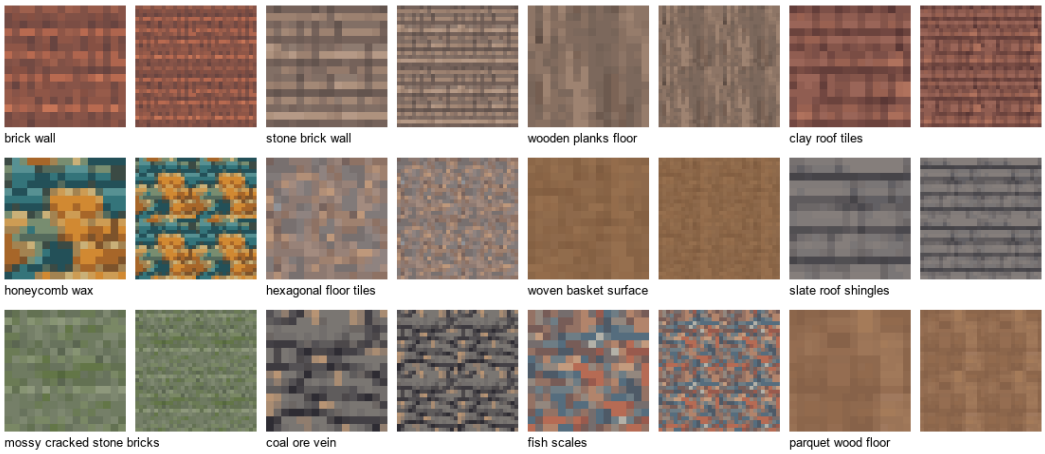


Figure 4: Best-of-4 outputs of Render at 16px for twelve generic material prompts (not test-set materials; the test-set Render samples are in Appendix E). Each pair shows the tile (left) and a 2×2 repetition (right), so seams and repeat artefacts are visible. For strongly periodic materials (bricks, planks, roof tiles, honeycomb) the crop yields clean, tileable structure, which is why Render is the strongest reference at this size (§6); others, such as woven basket or parquet, lose most of their structure at 16px.

C HUMAN STUDY: TRD+RR VS RENDER AT 16PX

Five annotators labelled a pre-registered split of the 16px name-to-texture comparison (TRD+rr vs Render). Ten materials were shared; the rest were partitioned at random before any labels were collected. Each pair appeared in both orders; inconsistent pairs are discarded. Attention checks (a sharp tile versus a blur) and same-image checks (identical tiles on both sides, the correct answer being "cannot tell") void a whole booklet if failed. One annotator failed the attention checks and is excluded. The remaining four yield $67/91 = 73.6\%$ for TRD+rr (Table 6). A per-material majority vote on unique materials gives the same direction ($49/64 = 76.6\%$). After labelling, annotators reported that they had not noticed the "cannot tell" option on the same-image checks and two spruce pairs; those items were revised to "cannot tell" as they stated, and the unrevised CSVs are in the repository.

Table 6: Human blind study at 16px, TRD+rr vs Render. Pooled = all two-order-consistent pairs from booklets that pass the checks. Jeffreys interval on the pooled rate: $[0.64, 0.82]$.

Booklet	Decided	TRD+rr wins	Rate	p
A	22	12	54.5%	0.83
B	23	18	78.3%	0.011
C	23	18	78.3%	0.011
D	23	19	82.6%	0.0026
E (void: 2 failed checks)	–	–	–	–
Pooled A–D	91	67	73.6%	7.4×10^{-6}

D ADDITIONAL JUDGE ARMS AND RESULTS AT 24 AND 32PX

Table 7: Additional judge arms on the test set: single-sample TRD at 16px, and all arms with CLIP scores at 24 and 32px (distribution metrics are not reported there because the test packs contain too few artist tiles).

Size	Comparison	Wins / decided	Rate	p	Family 95% CI
16	TRD (single sample) vs Retrieve	113 / 213	52.6%	0.41	[0.447, 0.627]
16	TRD (single sample) vs Render	90 / 190	47.4%	0.51	[0.395, 0.553]
24	TRD+rr vs Render	94 / 188	50.0%	1.0	[0.421, 0.575]
24	TRD+rr vs PixelDiff	144 / 208	69.2%	2.9×10^{-8}	**[0.631, 0.753]**
32	TRD+rr vs PixelDiff	116 / 188	61.7%	0.0016	**[0.535, 0.698]**
32	TRD+rr vs Shrink	105 / 186	56.5%	0.091	[0.486, 0.642]
32	TRD+rr vs SD-πXL (12 mat.)	10 / 12	83.3%	0.039	n/a
32	TRD+rr vs Render	83 / 202	41.1%	0.014	[0.315, 0.502]

CLIP at 24px: Shrink 35.17, Render 35.73, LoRA **35.86**, PixelDiff 33.83, TRD+rr 35.66
CLIP at 32px: Shrink 34.63, Render **35.52**, LoRA 35.43, PixelDiff 33.73, TRD+rr 34.70

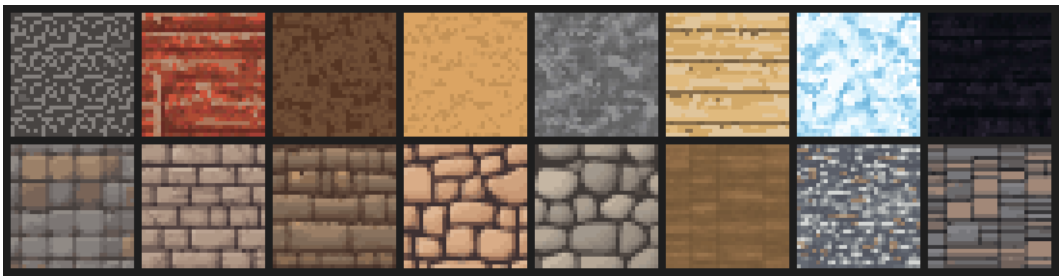


Figure 5: 32px samples for eight materials, TRD+rr (top row) and Render (bottom row), nearest-neighbour enlarged. The two families scale differently with the canvas. Render crops a fixed number of periods, so a larger canvas shows the same block-level units with more cells (the canvas gain of Ruler 2). TRD keeps the per-pixel density of authored 16px art: its edge density at 32px relative to 16px is 0.954, against 0.840 for Render (paired $p = 4.2 \times 10^{-10}$).

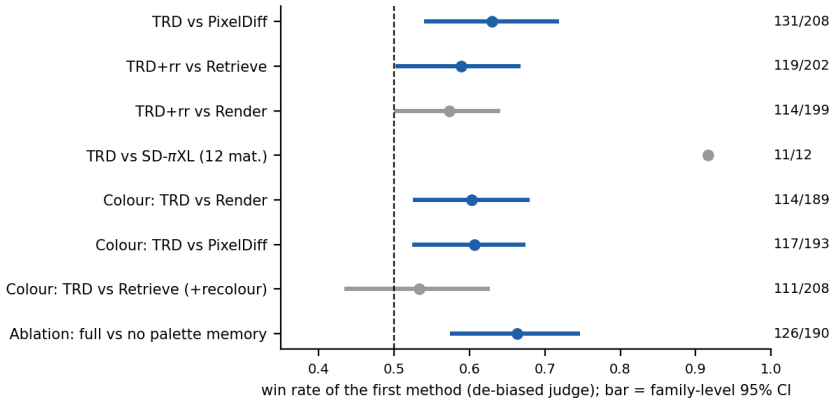


Figure 6: All 16px judge arms with family-level intervals. Blue: interval excludes 0.5. The SD-πXL arm ($n = 12$) has only a binomial test. The last row is the palette-memory ablation of §6.4.

Table 8: Region-colouring task at 16px (857 colour targets). "+shift": the same Lab residual correction applied to every method. KID $\times 10^3$.

Method	$\Delta E \downarrow$	KID $\downarrow$	FID $\downarrow$	FD-DINOv2 $\downarrow$	CLIP $\uparrow$
Shrink + shift / palette recolour	0.16 / 15.2	41.4 / 39.3	63.9 / 66.6	126.7 / 182.1	33.75 / 33.83
Render + shift / palette recolour	0.17 / 15.3	24.6 / 26.9	55.1 / 58.6	151.2 / 179.5	34.21 / 34.08
PixelDiff + shift	0.32	**8.6**	34.3	**52.1**	33.26
TRD native / + shift	3.05 / 0.30	10.2 / 9.9	31.4 / **31.2**	53.5 / 54.2	**34.30** / 34.29
Retrieve + shift (reference)	0.23	1.6	29.7	27.6	33.68

E Qualitative Results at 16px

All samples below are the test-set outputs scored in §6 (first sample per material; TRD+rr is the reranked sample). Every cell shows a 2×2 repetition of the 16px tile, nearest-neighbour enlarged, so seams and repeat artefacts are visible. The artist tile is the held-out reference for that material; no method saw it, and Retrieve draws only from the training split. Figure 7 covers the region-colouring task and Figure 8 the name-only task, whose panel (a) is selected and panel (b) is not.

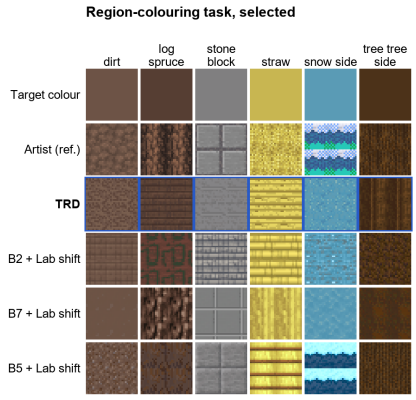


Figure 7: Region-colouring task, *selected* from the 32 test materials on which the judge preferred TRD over all three recoloured baselines. The target is the artist tile's mean colour; baselines are Lab-shifted to it afterwards (TRD gets the same shift as a residual correction), so differences lie in structure and palette spread. TRD follows the target natively with material-appropriate structure, whereas a post-hoc shift keeps whatever the baseline drew, including structure foreign to the material (Render on log spruce and stone block).

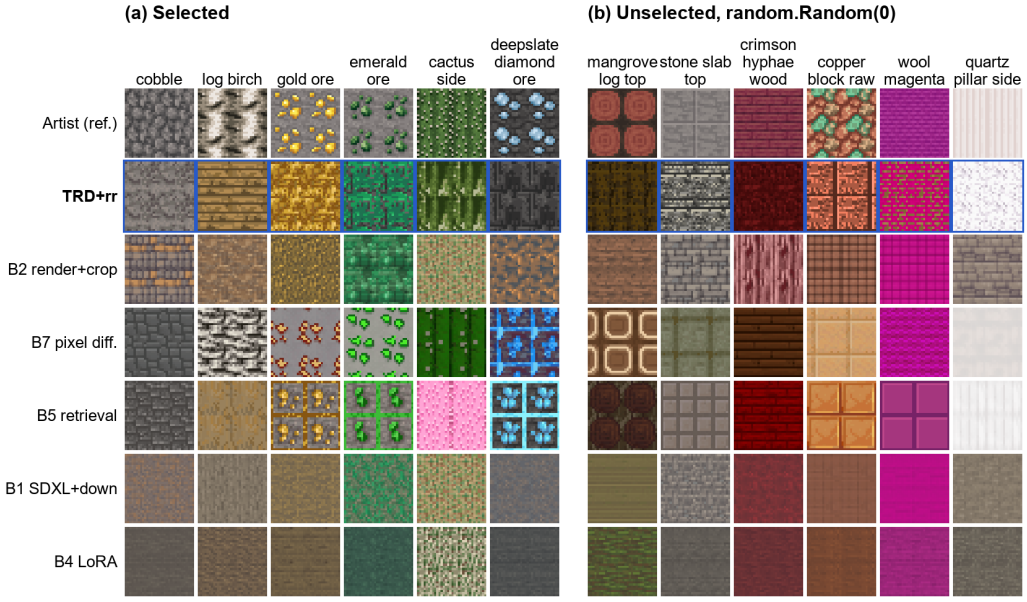


Figure 8: Name-only task; rows are methods, columns are test materials. **(a) Selected:** chosen by us for variety from the 70 of 272 test materials on which the judge preferred TRD+rr over both Render and Retrieve (both presentation orders agreeing). TRD keeps material-appropriate palettes and repeats seamlessly, whereas Shrink and LoRA wash out to near-uniform fills and Render's crop can pick the wrong structure (cobble, cactus side). **(b) Unselected:** six of the 272 test materials drawn with a fixed seed (`random.Random(0)`) before looking at the samples, shown so that panel (a) can be read against typical outputs.