# UNPAINTING: CHARACTERIZING BRUSHSTROKE ORDER RECOVERY FROM FINISHED PAINTINGS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

When a painting is finished, the order in which its strokes were laid down is no longer recorded; only the final surface survives. We ask how much of that hidden order a painting can still reveal. We call this *unpainting*: recovering the stroke sequence behind a finished canvas. We introduce `Pentimento`, a conditional diffusion model, and SYNBRUSH, the synthetic corpus of known-order paintings we train it on; rather than guessing one answer, it proposes many plausible orderings and reports how much they agree, with pairwise probabilities well calibrated on held-out paintings. Recovery is real but limited: far better than chance, yet well short of certainty. Targeted interventions trace it predominantly to the learned process prior: hand the model a stranger's canvas, or none at all, and it does just as well; a model trained without a canvas reaches Kendall's $\tau = 0.257$ against the conditional model's $0.272$. Meanwhile, a simple rule asking only which stroke covers which recovers twice as much of the order, though it is handed the strokes the model must find. Conditioning closes $5\%$ of that gap: it says substantially more about the stroke parameters than about their order. The finished canvas remembers more about how it was made than the model takes from it. What `Pentimento` does offer is calibrated confidence in its own samples: answer only the fifth it is surest about and it is right $97\%$ of the time. That confidence is prior-driven too, barely better than a stroke-width heuristic, uncorrelated with what a given canvas determines. We offer *unpainting* as a controlled study of prior dominance in amortized inverse inference, not a tool for art historians.

## 1 INTRODUCTION

In art history, reconstructing the order of brush strokes in a painting is crucial to study attribution of the work, dating, and workshop practice: for instance, an underpainting laid before a final glaze tells a different story than the reverse (Sabetsarvestani et al., 2019), and detecting an overpaint layer can help conservators distinguish original marks from later interventions (Van Vijle, 2026). Physical imaging captures only part of this information. X-ray and infrared reflectography reveal a painting's buried *layers*, but not the order in which the brush moved within and between them, often the very thing in dispute when telling a master's hand from a workshop copy, or dating a reworking. Stroke order is indeed a temporal dimension that the finished surface preserves only in part. In this paper, we question how much of such temporal dimension can be read back from the image alone. Existing methods touch this only obliquely. Stroke-based rendering (Huang et al., 2019; Zou et al., 2021; Liu et al., 2021; Tong et al., 2022; Tang et al., 2025; Sawada & Katsurai, 2024) breaks an image into strokes that re-create it, but the order is whatever the method emits, never checked against the artist's. Time-lapse methods (Zhao et al., 2020; Song et al., 2024; Chen et al., 2024; Pobitzer et al., 2025) synthesize plausible videos of a painting forming, but pixel by pixel, not stroke by stroke. Existing work (Prudviraj & Jamwal, 2025) also orders traced curves by proximity, but ignores uncertainty.

In this work, we hence introduce the task of *unpainting* — a controlled setting for these questions rather than a tool ready for them — which we treat as inferring a hidden process from a single image: given a finished canvas, which ordered stroke sequences could have produced it? This task is not only relevant in the art historical domain, but it also contributes to art pedagogy, where instructors often decompose a work into steps, either by hand or through tools like Procreate and Krita.[1] In

[1] Widely used digital painting applications that can record stroke-by-stroke history.



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.

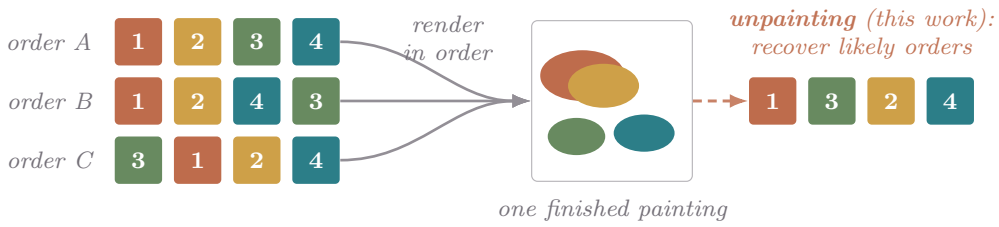



Figure 1: Illustrating the problem we tackle: stroke orders A, B and C composite to one image. While overlap records order, disjoint strokes do not. *Unpainting* inverts that map.

addition, in generative modeling, the task of *unpainting* allows considering artistic production from its *procedural* perspective, rather than only from the final static image. This creates a step toward models that reason about how pictures come to be, not only how they look.

The problem that *unpainting* addresses is fundamentally ambiguous since many different answers are plausible (Fig. 1). For this reason, we propose `Pentimento`[2], a conditional diffusion model that draws on action-sequence models in robotics, namely Diffusion Policy (Chi et al., 2025) and Diffuser (Janner et al., 2022), replacing motor actions with parametric strokes and a real-time observation stream with a single terminal image. As an output, `Pentimento` proposes many plausible orderings at different confidence levels (Fig. 2). In this work, we train on procedurally generated paintings with known stroke order (shared in our dataset SYNBRUSH), so recovery is measured directly against ground truth.

**Contributions.** We **(i)** cast brushstroke-order recovery as inferring a distribution over stroke sequences from a finished canvas, and adapt action-sequence diffusion to the painting setting as `Pentimento` (Section 3); **(ii)** release SYNBRUSH, a synthetic benchmark of paintings with known construction order, and use it to characterize how recoverable order is (Section 4), placing the model between non-learned references (random and proximity near chance, a width-sort of the true stroke set far higher); and **(iii)** show that this recovery is predominantly prior, not evidence (§5). The canvas reaches and perturbs the model's output and informs the stroke parameters, yet contributes almost nothing to their order: training with no canvas at all costs only $0.015\ \tau$, while a non-learned occlusion sort, handed the true stroke set, recovers about twice as much. This separates three quantities the task conflates: the order a process prior predicts, the order the canvas evidences, and the order the model extracts.

## 2 RELATED WORK

The existing tradition in art history (Tan et al., 2015; Sabetsarvestani et al., 2019) recovers *layer* structure from material evidence of paintings (*e.g.*, macro X-ray fluorescence, infrared, or X-ray), but not stroke order. In parallel, the dominant line of work on parametric strokes treats stroke decomposition as a tool for stylized *rendering*. Reinforcement-learning (Huang et al., 2019), direct-optimization (Zou et al., 2021), and feed-forward Transformer (Liu et al., 2021) variants decompose an image into brushstroke parameters that approximate it when composited; stroke order is set by the rendering schedule, not recovered. Extensions add adaptive anchor sampling (Tong et al., 2022), region-compositional rendering (Hu et al., 2023), single-pass prediction (Tang et al., 2025), and state-space backbones (Sawada & Katsurai, 2024); AttentionPainter's stroke-parameter diffusion module (Tang et al., 2025) targets generative editing, not sequence recovery.

A related thread casts painting as sequential decision-making: agents such as SPIRAL (Mellor et al., 2019) learn to reproduce an image stroke by stroke, but again the discovered order serves faithful rendering and is never checked against the true sequence. Close to ours, Sketch & Paint (Prudviraj & Jamwal, 2025) vectorizes a painting and orders the curves by clustering and a traveling-salesman tour over centroids; a deterministic heuristic evaluated only qualitatively, with no model of the posterior uncertainty the inverse problem carries.

[2]after the art-historical term for an earlier composition showing through the surface of a finished painting.

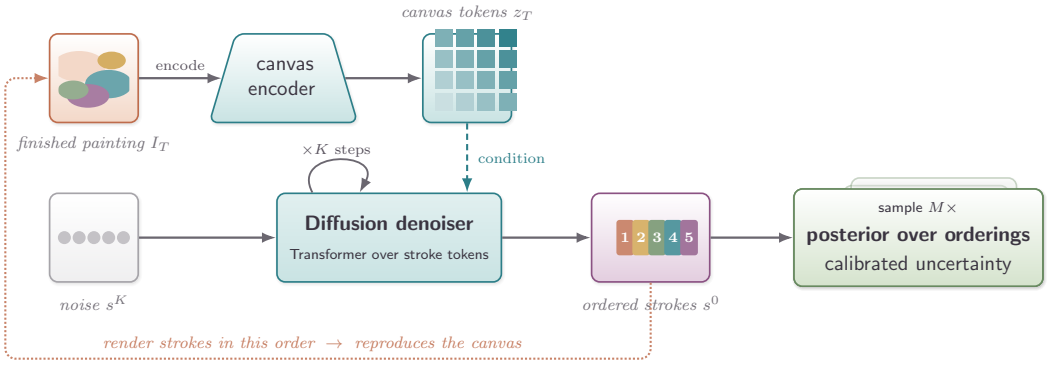



Figure 2: Our workflow. An encoder maps the painting $I_T$ to canvas tokens $\mathbf{z}_T$; conditioned on these, the denoiser refines noise $\mathbf{s}^K$ into an *ordered* stroke sequence $\mathbf{s}^0$ over $K$ steps. $M$ such samples form the posterior; repainting them approximately reproduces $I_T$ (dotted).

A second line generates plausible painting *videos* from a finished image, namely TimeCraft (Zhao et al., 2020), ProcessPainter (Song et al., 2024), Inverse Painting (Chen et al., 2024), and Loomis Painter (Pobitzer et al., 2025), but all operate at the frame level, emitting pixel sequences scored frame-by-frame; none recover stroke-parametric trajectories. Loomis Painter's "reverse-painting" trick reorders pixel frames, unrelated to our inference-time *unpainting*. Our model works one level deeper, emitting ordered stroke *actions* scored against true stroke order; it is an amortized conditional diffusion solver that, unlike measurement-consistency approaches (Li & Pereira, 2024; Chung et al., 2025; Wang et al., 2024), never evaluates the renderer at inference.

## 3 Method

In this paper, we propose `Pentimento`, a learned *denoiser* (Fig. 2) that, starting from random stroke parameters and looking at the finished canvas, repeatedly refines them into a plausible *ordered* list of strokes that, painted in that order, approximately reproduce that canvas. Because each run starts from different random values, sampling many times yields not one answer but a *distribution* of construction orders, whose spread measures how confidently order can be recovered.

### 3.1 Problem formulation

Let $I_T \in \mathbb{R}^{H\times W\times 3}$ denote a finished painting; we call $I_T$ the *canvas* throughout, meaning the finished surface in its role as the model's only observation. Let $\mathbf{s} = (s_1, s_2, \ldots, s_N)$ denote an ordered sequence of $N$ parametric strokes, where each $s_i \in \mathbb{R}^d$ encodes the geometric and chromatic parameters of a single stroke. Here $N$ is that painting's own stroke count and the subscript indexes position in the sequence, so $s_N$ is the last stroke laid down. Let $R : \mathbb{R}^{N\times d} \to \mathbb{R}^{H\times W\times 3}$ be a renderer that composites a stroke sequence in order, producing a final image $\hat{I}_T = R(\mathbf{s})$.

We are interested in recovering the artist's stroke trajectory $\mathbf{s}^\star$ that gave rise to an observed canvas $I_T$: concretely, in sampling from the posterior $p(\mathbf{s} \mid I_T)$ over orderings. Its Bayesian form, a stroke prior against a renderer likelihood, is given in Appendix A.

The problem is fundamentally ill-posed (Fig. 1): many distinct orderings of the same stroke *set* yield indistinguishable canvases whenever a later stroke fully occludes an earlier one, and many stroke sets reconstruct the canvas to within renderer tolerance.

We therefore target the full posterior rather than a single estimate. We do not evaluate the renderer likelihood at inference; we learn an *amortized* conditional model $p_\theta(\mathbf{s} \mid I_T)$ from paired (canvas, trajectory) data by supervised conditional diffusion (Section 3.3), the renderer entering only as the forward map that generates training canvases (Appendix A).

### 3.2 STROKE REPRESENTATION AND RENDERING

Each stroke carries **14 fields**, following (Liu et al., 2021; Zou et al., 2021): three quadratic-Bézier control points (6 values, normalized to the canvas), start and end widths, RGB color and opacity, a discrete brush-type id, and a discrete layer id.

The diffusion model operates on the $d{=}12$ continuous geometry-and-appearance dimensions; brush-type and layer ids are discrete metadata that the model does not predict (the layer id is the auxiliary-supervision target for runs D and E). The parameterization is deliberately compatible with prior stroke-based rendering; our contribution is its treatment as the latent variable of an inverse problem.

A deterministic **renderer** $R$ turns a stroke sequence into an image: spines are rasterized as variable-width polylines and composited in order with source-over alpha blending. We need $R$ only to synthesize the training canvases (Section 4); since the denoiser learns from stroke parameters rather than pixels, $R$ does not need to be differentiable or photorealistic. What matters is *order-sensitivity*: rendering $\mathbf{s}$ in a different order generically changes the image (later strokes occlude earlier ones), which is what makes order recoverable in principle.

### 3.3 CONDITIONAL TRAJECTORY DIFFUSION

We adapt action-sequence diffusion (Chi et al., 2025; Janner et al., 2022) to stroke trajectories. A clean stroke sequence $\mathbf{s}^0 \in \mathbb{R}^{N\times d}$ is corrupted under a variance-preserving DDPM schedule across $K$ steps; token order is preserved by the forward process and is what the denoiser must learn to respect. A conditioning embedding $\mathbf{z}_T = \phi(I_T) \in \mathbb{R}^{L\times d_z}$ is extracted from the finished canvas by a learned convolutional encoder over a $16{\times}16$ patch grid (sizes in Appendix B).

The denoiser is a Transformer that processes $\mathbf{s}^k$ as $N$ tokens with sinusoidal stroke-index positional encodings, cross-attending to $\mathbf{z}_T$ at every layer and conditioned on step $k$ via adaptive layer norm. The primary loss is standard $\boldsymbol{\epsilon}$-prediction over real stroke positions; the `+aux` runs (D, E) add an auxiliary per-patch layer-prediction loss ($\lambda{=}0.5$).

Variable-length sequences use a fixed $N_{\max}$ with an end-of-painting token; padded positions are excluded from the loss *and* masked out of attention, at training and at sampling alike. Sampling a painting at more than its own $N$ without that mask lets the denoiser read pad slots as strokes, which measurably depresses recovered order. At inference we sample $\mathbf{s}^K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and run the standard DDIM reverse loop (Ho et al., 2020; Song et al., 2021) for $K'{=}25$ steps with $I_T$ as conditioning, recovering $\mathbf{s}^0$.

Because each run starts from fresh noise, drawing $M$ independent samples for a canvas $I_T$ yields a posterior over orderings whose dispersion we read two ways. *Order entropy*: for each pair of distinguishable strokes $(i, j)$, the empirical probability that $i$ precedes $j$ across samples defines a Bernoulli, and we report the mean binary entropy over pairs ($\approx 0$ when the samples agree on a single order and approaches $\log 2$ when they are split evenly). *Set agreement*: which strokes the samples agree are present, independent of order, via Hungarian-matched cosine distance. Together these flag *how much* of a recovered process to trust, and where that trust breaks down.

### 3.4 IMPROVING RECOVERY: REGION ATTENTION AND A LENGTH CURRICULUM

Two low-cost changes aim to improve recovery without more data or compute. The first is a spatial bias on conditioning: a stroke's order is most informative about the canvas right where it was painted, yet standard (global) cross-attention lets every stroke look at every canvas patch with no spatial prior, re-learning locality at every layer, so we bias it toward the stroke's **own neighborhood**.

Concretely we add to each cross-attention logit a learned, per-head, step-conditioned penalty on the distance between a stroke's centroid and each canvas patch (Appendix C), at $\sim$5k parameters and $+6\%$ wall time.

The second is a length curriculum over the stroke count $N$: training starts on the shortest paintings and the cap expands at fixed milestones — caps $(120, 200, 300)$ at steps $(1500, 3000)$ — so only the length distribution changes.



$\mathcal{P}_0$: widths only $\mathcal{P}_1$: $\mathcal{P}_0$+ color $\mathcal{P}_2$: $\mathcal{P}_1$+ regions

Figure 3: The three prior levels: finished canvases, fixed seed, 50 strokes. $\mathcal{P}_0$ and $\mathcal{P}_1$ share geometry and differ only in color; $\mathcal{P}_2$ also moves strokes.

## 4 SYNBRUSH: SYNTHETIC STROKE PROCESSES

A finished real painting preserves no *complete* record of the order its strokes were laid in, so order recovery cannot be *measured* against truth on real data. We therefore introduce SYNBRUSH, a procedurally generated corpus in which every painting carries its exact construction order by design.

Steps (i)–(iii) are common to every prior level: the generator (i) samples 2–4 logical layers (background, mid-ground, detail, highlights); (ii) draws stroke widths conditioned on layer, decreasing across layers; and (iii) samples stroke positions with local coherence within a layer (a Gaussian-neighborhood walk with probability $p_{\text{local}}$=0.6, uniform otherwise).

Two further steps are optional, and switching them on is what defines the levels: (iv) modulating color per layer in HSV (hue, saturation, value), so later layers read brighter and more saturated; and (v) partitioning the canvas into 3–6 regions, each with its own sub-palette, painted one region at a time within a layer.

This gives $\mathcal{P}_0$ (steps i–iii alone), $\mathcal{P}_1$ (+ iv) and $\mathcal{P}_2$ (+ v): three nested levels of increasing structure (Fig. 3). Each added step writes a different kind of order evidence into the finished canvas. In particular, step (iv) makes depth legible from brightness, and step (v) from spatial color clustering. In Section 5, we report how much of it the model recovers. Each sample stores a flat RGB canvas and the complete ordered $(N, 14)$ stroke array, with $N$ between 50 and 300. We use 100 paintings per level (90/10 train/val) for the main experiments and 1,000 for the 10×-data run (H), with a disjoint 100-painting set held out for the final check (Section 5). Appendix D shows the generator over a wider range of configurations than the corpus uses, un-painted stroke by stroke.

## 5 EXPERIMENTS

We evaluate `Pentimento` along four axes: how much of the order is recoverable at all, against non-learned references (Section 5.1); which design choices have an impact on the results (Section 5.2); whether the recovery uses the canvas (Section 5.3); and what the posterior's dispersion does and does not track (Section 5.4). All results are on SYNBRUSH, with mean ± s.d. over 3 seeds for the comparisons the argument rests on. No prior method is directly comparable on $\tau$: stroke-based renderers (Liu et al., 2021; Huang et al., 2019) order strokes by a chosen execution schedule rather than against the artist's, Sketch & Paint (Prudviraj & Jamwal, 2025) reports no order metrics, and frame-level methods (Chen et al., 2024; Song et al., 2024; Pobitzer et al., 2025; Zhang, 2024) emit no parametric strokes. We therefore calibrate against non-learned references instead.

### 5.1 HOW MUCH ORDER IS RECOVERABLE

We report Kendall's $\tau$ and Spearman's $\rho$ on stroke orderings (after Hungarian matching predicted to ground-truth strokes, so order metrics stay well-defined when the predicted set differs), and Dynamic Time Warping (DTW) on action sequences. Kendall's $\tau$ measures pairwise order agreement ($\tau$=1 exact, 0 chance, −1 reversed); DTW rewards local order coherence while tolerating global shifts (lower is closer). We evaluate on a 10% *validation* split of SYNBRUSH at the training prior; a disjoint 100-painting *held-out test* set is reserved for the final check.

On these metrics, the $\mathcal{P}_2$ baseline (run C) recovers mean $\tau = 0.270$ (best $\tau^\star = 0.313$) over 3 seeds, the operating point against which we test the levers below; the $\mathcal{P}_1$ variant (run B) is weaker ($\tau =$

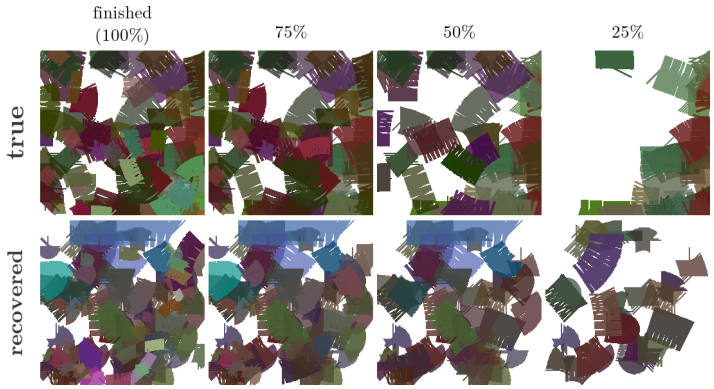



Figure 4: The artist's order (top) and one posterior sample (bottom) at matched stroke fractions. The *set* and the coarse-to-fine progression come back; the order *within* each stage does not.

0.234). Fig. 4 shows what that number looks like: the stroke set and the coarse-to-fine progression come back, the order within each stage does not.

Those are development-validation figures (10 paintings, selected over rounds); we verify once on a disjoint 100-painting test set with the frozen best-validation checkpoint. The best-validation checkpoint (run K, curriculum + region attention; Appendix B) scores $\tau = 0.304$ there against the baseline's 0.278; the gain (+0.026) generalizes. Both fall below their best-validation figures, the expected optimism of a 10-painting validation set; the held-out $\tau \approx 0.30$ is the honest estimate of recovery from the finished painting alone [3], well above the $\approx 0$ floor.

To calibrate these numbers, we order the *oracle* stroke set by simple rules (Table 1): random and proximity orderings score $\tau \approx 0$, a width-descending sort reaches 0.55, and the generator's layer key recovers order exactly. Substantial order signal is thus present *given the set*; the model's $\tau \approx 0.30$ from the painting alone reflects the combined difficulty of recovering set *and* order once strokes occlude, and no cheap lever closes the gap. Section 5.3 shows the shortfall lies in what the model extracts, not in what the canvas retains.

Table 1: Reference points, not competitors. The upper block is *handed the correct stroke set* and need only order it; `Pentimento` sees the finished canvas alone and must recover the set *and* its order. Scored against the artist's order on $\mathcal{P}_2$; bars are proportional to $\tau$.

| Ordering rule | Kendall $\tau$ | Spearman $\rho$ |
|---|---|---|
| *Given the true stroke set, order unknown* | | |
| Random permutation | 0.00 | 0.00 |
| Spatial-proximity tour (greedy nearest-neighbor) | 0.00 | 0.00 |
| Dark→bright (luminance) | 0.33 | 0.48 |
| Width-descending (coarse-to-fine) | 0.55 | 0.75 |
| Occlusion-graph topological sort | **0.58** | n/a |
| Layer key (generator oracle) | 1.00 | 1.00 |
| *From the finished canvas alone* | | |
| `Pentimento` (held-out test) | 0.30 | 0.45 |

## 5.2 What moves recovery, and what does not

We sweep five design levers from the $\mathcal{P}_2$ baseline (Table 2; the full twelve-run sweep, with DTW and the cheap variants, is in Appendix B). *Prior strength* helps monotonically ($\mathcal{P}_0 \to \mathcal{P}_1 \to \mathcal{P}_2$: mean $\tau$ 0.142 → 0.234 → 0.270): each added layer of generator structure makes the construction order more visible, and recovery keeps rising rather than saturating.

[3] All numbers use the padding-masked, batch-invariant protocol of Section 3.3.

Table 2: The runs the argument leans on. Mean ± s.d. over 3 seeds except A; bars are proportional to $\tau$, on the same scale as Table 1. The full twelve-run sweep is in Appendix B.

| Run | Configuration | Kendall $\tau$ | $\tau^\star$ |
|---|---|---|---|
| *Prior strength, at fixed compute* | | | |
| A | $\mathcal{P}_0$ (widths only) | 0.142 | 0.191 |
| B | $\mathcal{P}_1$ (+ color) | 0.234±.014 | 0.277±.020 |
| C | $\mathcal{P}_2$ (+ scene regions) | 0.270±.006 | 0.313±.012 |
| *Training length and architecture, at $\mathcal{P}_2$* | | | |
| F | 4× training (20k) | **0.343**±.018 | **0.395**±.016 |
| G | 1D U-Net denoiser | 0.051±.015 | 0.127±.033 |

*Training* 4× *longer* is the only lever that moves $\tau$ substantially, to 0.343. Auxiliary supervision is neutral, 10× data at fixed compute hurts, and the two cheap variants of Section 3.4 (*i.e.,* a length curriculum and region attention) sit barely above the baseline on best-of-seed $\tau^\star$ (0.323, 0.326, 0.328 against 0.313). *A 1D U-Net* is the informative exception: at matched compute it reaches *lower* training loss but collapses to $\tau \approx 0.05$, because its locality bias is wrong for stroke trajectories, since temporally adjacent strokes need not be spatially adjacent.

No cheap lever closes the gap to the evidence-only sort of Table 1, and Section 5.3 explains why: none of them changes how little of the order the model takes from the canvas.

### 5.3 The recovery is predominantly prior, not evidence

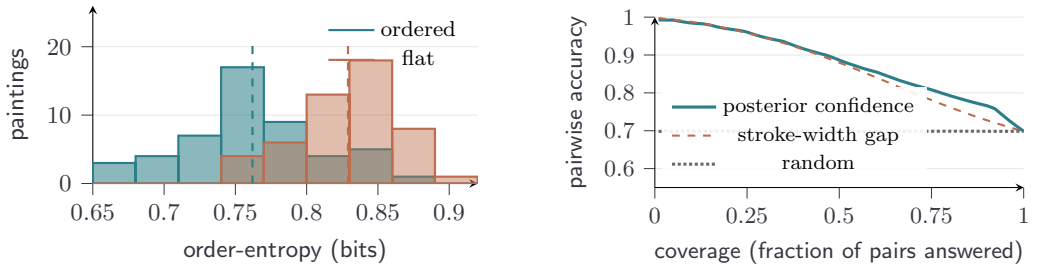
If the recovered order came from the painting, removing the painting should destroy it. On a frozen checkpoint we sample the same 100 held-out paintings under four conditioning arms: (1) the *correct* canvas, (2) a *stranger's* canvas (a global derangement), (3) the *dataset-mean* canvas tokens, and (4) *zeroed* tokens. In these settings, we hold the initial noise fixed per painting so that differences isolate conditioning, and we report the paired contrast.

Table 3: Conditioning ablation, 100 held-out paintings. At inference, a stranger's painting, the dataset mean, or no canvas at all never costs accuracy; on two checkpoints the correct canvas is significantly *worse* than none. Trained without a canvas from the start, the model loses 0.015 $\tau$. $p$: Wilcoxon signed-rank on the paired per-painting difference.

| Checkpoint | real | stranger | mean | zero | real − zero | $p$ |
|---|---|---|---|---|---|---|
| Curriculum + region (K) | 0.306 | 0.316 | 0.312 | **0.326** | −0.020 | 3.5e−5 |
| Baseline (C) | 0.278 | 0.283 | 0.287 | 0.273 | +0.004 | 0.24 |
| 4× training (F) | 0.364 | 0.357 | 0.365 | 0.361 | +0.002 | 0.95 |
| Region + curriculum | 0.305 | 0.313 | 0.312 | **0.326** | −0.021 | 5.9e−6 |
| *Trained from scratch with no canvas at all* | | | | | | |
| Unconditional (3 seeds) | | 0.257 | | | vs. conditional C 0.272; paired +0.015 | |

Every checkpoint gives the same answer (Table 3): at inference, the correct canvas does not improve accuracy, and on two of the four the model scores significantly *higher* with no canvas at all. The conditioning is not simply disconnected. The encoder tells paintings apart (cosine similarity 0.19 between two paintings' tokens, nowhere near the 1.0 of a constant encoder), and those tokens move the denoiser's output by about a sixth of its own spread (0.09 against a standard deviation of 0.53). The conditioning reaches the denoiser and perturbs it; that perturbation simply carries almost no order information.

In addition, zeroing a trained model's conditioning is an out-of-distribution edit, so we also trained the denoiser from scratch with the canvas tokens zeroed throughout — architecture, data, optimizer, schedule, steps, seeds and checkpoint rule identical to run C, the image simply absent. Scored on the same disjoint 100-painting test set, three seeds reach $\tau = 0.257$ against the conditional baseline's 0.272 (*i.e.*, 94% of it). Paired painting by painting, the canvas is worth $+0.015$ $\tau$, with a 95% interval of $[-0.007, +0.036]$, resampling paintings and seeds, so we do not claim equivalence. Holding these




(a) order-entropy, ordered vs. flat canvases

(b) risk–coverage over 1.79M held-out pairs

Figure 5: Two readings of the posterior. (a) Entropy separates the two generator regimes, structured $\mathcal{P}_2$ canvases from flat controls (0.762 vs. 0.829 bits, $U$=324, $p<10^{-9}$), but does not track what a given canvas determines (Section 5.4). (b) Abstaining sharpens accuracy, but posterior confidence barely beats a stroke-width gap (AURC 0.122 vs. 0.134).

three trained models fixed and resampling paintings alone brings the interval to $[+0.005, +0.025]$. Either way, these scores are small against the 0.311 that separates the model from the occlusion sort of Section 5.1: conditioning closes about 5% of that gap. What it does inform is the strokes themselves: DTW over stroke *parameters* improves from 230 to 217, on 90 of the 100 paintings. The canvas says substantially more about what the strokes are than about the order they were laid in. In addition, via undertraining, the null is not an artifact of undertraining either: it holds on a checkpoint trained $4\times$ longer (run F), which scores the highest $\tau$ and uses the canvas least. That is what we expect if extra compute sharpens the prior rather than the reading.

Meanwhile, rasterizing each stroke's footprint and recording which is visible inside which yields an occlusion graph that directly evidences 6.1% of pairs, lifted to 20.8% by transitive closure (16.8% of strokes leave no trace at all). A topological sort of that graph scores $\tau = 0.583$ on the true stroke set (Table 1), roughly twice the model's 0.306. The sort is handed the strokes the model must infer, so these experiments are carried out in different settings. However, this comparison, while not being fully fair, does highlight that the occlusion evidence in the canvas carries information that the model is not learning.

Two further checks probe what the recovered order depends on. First, the metric: $\tau$ scores every pair, including pairs that never touch and whose order the canvas arguably cannot evidence, so restricting it to interacting pairs is the natural correction. Marking the pairs whose footprints intersect (a minority, 16%) and scoring them separately does *not* raise the score: $\tau$ is 0.278 on intersecting pairs against 0.311 on disjoint pairs and 0.306 overall.

This raw comparison is, however, confounded. Two strokes can only intersect if they land in the same place, and painters return to a place within a short stretch, so intersecting pairs sit closer in time (median separation 56 against 64), and near-in-time pairs are hard for any coarse-to-fine prior. Re-weighting the disjoint pairs to match that distribution removes most of the gap, and what remains is a null: across three checkpoints the evidence-matched difference spans $-0.011$ to $+0.001$. The conclusion is therefore not that the model is *worse* where the canvas carries evidence, but that it is *no better*. It is not reading the local over-under cue; it rides the global width, layer and color regularities that the references of Table 1 exploit. Nor does the result depend on SYNBRUSH's distinct palettes: on a tonal variant with the same geometry it is unchanged (Appendix E).

### 5.4 WHAT THE POSTERIOR STILL TELLS US

A single trajectory is unreliable, but the spread across trajectories is not. At the same frozen checkpoint (run K) we draw $M$=16 samples for 50 strongly-ordered $\mathcal{P}_2$ paintings and 50 *flat* controls, whose sparse strokes rarely occlude, and take the painting-level order-entropy of Section 3.3.

The two distributions are well separated (Fig. 5a): ordered $\mathcal{P}_2$ canvases carry lower order-entropy than flat ones (0.762 vs. 0.829 bits), and set-agreement moves the same way (0.837 vs. 0.754). It is also calibrated on this held-out SYNBRUSH split (expected calibration error 0.028, Brier 0.161

against $0.25$ for always-$0.5$), so within this distribution the reported pair probabilities can be read as probabilities even though the modal sequence cannot.

But this separation is between two generator regimes, and does not by itself show that entropy tracks what a *given* canvas determines. Within the held-out $\mathcal{P}_2$ set it does not. Across the 50 held-out paintings with both an entropy and an occlusion graph, entropy is uncorrelated with the fraction of pairs that graph fixes ($r = 0.155$, 95% CI $[-0.14, 0.44]$) and *positively* correlated with what an evidence-only ordering achieves ($r = 0.467$, $[0.26, 0.67]$) — the opposite sign a recoverability signal would give, and controlling for stroke count does not help. The entropy measures the model's confidence, not the painting. The same quantity also resolves *spatially*, showing where on a painting the samples agree (Fig. 8 in Appendix F).

Calibration makes selective prediction possible: ranking all $1.79$M held-out pairs by confidence and answering only the most confident, accuracy rises from $0.699$ at full coverage to $0.969$ at a fifth and $0.985$ at a tenth (Fig. 5b): within this distribution, accuracy can be bought with coverage.

Two controls show that this confidence is the prior's rather than the painting's, consistent with Section 5.3. A non-learned stand-in nearly matches it: ranking the same pairs by the absolute difference in stroke width (*i.e.,* the coarse-to-fine cue and nothing else) gives an area under the risk–coverage curve of $0.134$ against the posterior's $0.122$ (random: $0.302$), the two curves nearly coinciding. And it barely knows what the canvas determines: as a detector of occlusion-determined pairs it reaches AUC $0.597$, near blind, and is *more* accurate on pairs fixed only through closure ($0.844$) than on directly evidenced ones ($0.720$). The abstention is usable; the uncertainty underneath it is not a reading of the canvas.

## 6 Discussion

A structural limit of the method proposed in this paper sits in the conditioning: the canvas is encoded to a $16\times16$ grid, so the over-under detail at a stroke boundary (*i.e.,* the evidence that fixes the order of two overlapping strokes) is largely averaged away before the strokes see it. That fits what Section 5.3 finds: conditioning informs the stroke parameters far more than their order, while an evidence-only sort reaches twice the model's $\tau$, so the evidence is not the binding constraint. Whether a finer or multi-scale grid would let a model extract it is the central question we leave open. The renderer also assumes opaque-to-translucent oil/acrylic compositing (Tan et al., 2015) and a Western-easel parameterization; other traditions would need parameterizations developed with their practitioners.

A plausible-looking trajectory is also easily mistaken for *the* trajectory: many sequences are observationally equivalent, so a single sample is not evidence of how a painting was made, and stroke recovery is one signal among many (MA-XRF, IR reflectography, archives), not an oracle. The bound we cannot supply is the human one: how much of the gap to the oracle-set references is algorithmic rather than intrinsic to visual inspection. The two steps that would settle the most are real-painting evaluation from logged stroke histories, which carry exact order for real decisions, and conditioning that preserves boundary evidence at the resolution where over-under relations live.

## 7 Conclusion

We frame brushstroke recovery as a generative inverse problem (which we refer to as *unpainting*) and ask how much of an artist's stroke order a finished canvas determines. We tackle this problem through our proposed method, `Pentimento`, on the SynBrush benchmark that we introduce, containing procedurally generated digital images with stroke ordering information. A curriculum model reaches $\tau \approx 0.30$, well above a $\approx 0$ floor, but a model trained with no canvas reaches $0.257$ against its $0.272$, while a non-learned occlusion sort, handed the true strokes, reaches $0.58$. Conditioning closes about $5\%$ of that gap, informing the strokes far more than their order. What the canvas determines and what the model takes from it are not the same quantity, and that gap is the result. The research question is therefore not *what is in a painting* but *how much of how it came to be can still tell us*, and how much of any given answer is determined by the prior.

## AI USE STATEMENT

We used generative AI tools to assist in implementing the code for our methods in `Pentimento` and SYNBRUSH (training, sampling, generation, evaluation, and analysis code). The algorithms, design of experiments, and validation approaches were all developed by the authors, and generative AI tools were then used to give feedback on experiments, refine hypotheses, and to interpret results. Additionally, we used these tools to edit parts of the manuscript, verify and format references, and search for related work.

We did not use generative AI tools for any other task for which ICLR requires disclosure. All AI-assisted code and analysis were reviewed, re-run and tested by the authors, and every reported number was regenerated under the final evaluation protocol. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

## ETHICS STATEMENT

This study develops methods for characterizing brushstroke order and involves no human subjects or personal data. Assessment of the results was completely done computationally with no human evaluators. SYNBRUSH is generated procedurally and contains no real artwork, no artist-identifying information, and no material scraped from collections or living practitioners; every painting in it, and every stroke order we evaluate against, is synthetic.

Stroke-order recovery touches attribution, dating and authentication, where conclusions carry financial and reputational weight. Our results argue *against* such use: the recovered order comes predominantly from a learned process prior rather than from the painting, many stroke sequences are observationally equivalent, and a single sample is therefore not evidence of how a work was made (Section 6).

## REPRODUCIBILITY STATEMENT

The stroke representation, renderer, diffusion model and posterior diagnostics are specified in Section 3; the generator and its three prior levels in Section 4; and the complete design-lever sweep, with per-run configurations and seeds, in Appendix B.

Three details matter most for exact reproduction. (i) Evaluation protocol. All reported numbers sample with the padding mask applied, which makes $\tau$ invariant to batch composition; sampling padded positions unmasked silently corrupts trajectories for any painting shorter than its batchmate and roughly halves the score. (ii) Splits. The $100$-painting held-out test set is disjoint from both the training pool and the validation split used for checkpoint selection; the development split is $10$ paintings drawn from the training pool and is never used for a reported test number. (iii) Uncertainty. Confidence intervals over pair-level quantities use a cluster bootstrap over *paintings*, not over pairs: the $1.79$M held-out pairs come from $100$ paintings and are strongly dependent, so a pair-level bootstrap would be badly anti-conservative.

Source code for the generator, training, sampling and every analysis script will be released upon acceptance.

## REFERENCES


Bowei Chen, Yifan Wang, Brian Curless, Ira Kemelmacher-Shlizerman, and Steven M. Seitz. Inverse painting: Reconstructing the painting process. In *SIGGRAPH Asia 2024 Conference Papers*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711312. doi: 10.1145/3680528.3687574. URL https://doi.org/10.1145/3680528.3687574.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025. doi: 10.1177/02783649241273668. URL https://doi.org/10.1177/02783649241273668.

Hyungjin Chung, Jeongsol Kim, and Jong Chul Ye. Diffusion models for inverse problems. *arXiv preprint arXiv:2508.01975*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Teng Hu, Ran Yi, Haokun Zhu, Liang Liu, Jinlong Peng, Yabiao Wang, Chengjie Wang, and Lizhuang Ma. Stroke-based neural painting and stylization with dynamically predicted painting region. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM '23, pp. 7470–7480, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701085. doi: 10.1145/3581783.3611766. URL https://doi.org/10.1145/3581783.3611766.

Zhewei Huang, Wen Heng, and Shuchang Zhou. Learning to paint with model-based deep reinforcement learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *Int. Conf. Mach. Learn.*, 2022. arXiv:2205.09991.

Henry Li and Marcus Pereira. Solving inverse problems via diffusion optimal control. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 73549–73571. Curran Associates, Inc., 2024. doi: 10.52202/079017-2339. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/86655bc516148e311bcfcf88f1744de7-Paper-Conference.pdf.

Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Ruifeng Deng, Xin Li, Errui Ding, and Hao Wang. Paint transformer: Feed forward neural painting with stroke prediction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6578–6587, 2021. doi: 10.1109/ICCV48922.2021.00653.

John F.J. Mellor, Eunbyung Park, Yaroslav Ganin, Igor Babuschkin, Tejas Kulkarni, Dan Rosenbaum, Andy Ballard, Theophane Weber, Oriol Vinyals, and S. M. Ali Eslami. Unsupervised doodling and painting with improved SPIRAL. *arXiv preprint arXiv:1910.01007*, 2019.

Markus Pobitzer, Chang Liu, Chenyi Zhuang, Teng Long, Bin Ren, and Nicu Sebe. Loomis painter: Reconstructing the painting process. *arXiv preprint arXiv:2511.17344*, 2025.

Jeripothula Prudviraj and Vikram Jamwal. Sketch & paint: Stroke-by-stroke evolution of visual artworks. In Alessio Del Bue, Cristian Canton, Jordi Pont-Tuset, and Tatiana Tommasi (eds.), *Computer Vision – ECCV 2024 Workshops*, pp. 202–217, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-92808-6.

Z. Sabetsarvestani, B. Sober, C. Higgitt, I. Daubechies, and M. R. D. Rodrigues. Artificial intelligence for art investigation: Meeting the challenge of separating x-ray images of the ¡i¿ghent altarpiece¡/i¿. *Science Advances*, 5(8):eaaw7416, 2019. doi: 10.1126/sciadv.aaw7416. URL https://www.science.org/doi/abs/10.1126/sciadv.aaw7416.

Tomoya Sawada and Marie Katsurai. Mambapainter: Neural stroke-based rendering in a single step. In *SIGGRAPH Asia 2024 Posters*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711381. doi: 10.1145/3681756.3697906. URL https://doi.org/10.1145/3681756.3697906.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Yiren Song, Shijie Huang, Chen Yao, Hai Ci, Xiaojun Ye, Jiaming Liu, Yuxuan Zhang, and Mike Zheng Shou. Processpainter: Learning to draw from sequence data. In *SIGGRAPH Asia 2024 Conference Papers*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711312. doi: 10.1145/3680528.3687596. URL https://doi.org/10.1145/3680528.3687596.

Jianchao Tan, Marek Dvorožňák, Daniel Sýkora, and Yotam Gingold. Decomposing time-lapse paintings into layers. *ACM Trans. Graph.*, 34(4), July 2015. ISSN 0730-0301. doi: 10.1145/2766960. URL https://doi.org/10.1145/2766960.

Yizhe Tang, Yue Wang, Teng Hu, Ran Yi, Xin Tan, Lizhuang Ma, Yu-Kun Lai, and Paul L. Rosin. Attentionpainter: An efficient and adaptive stroke predictor for scene painting. *IEEE Transactions on Visualization and Computer Graphics*, 31(12):10897–10911, 2025. doi: 10.1109/TVCG.2025.3618184.

Zhengyan Tong, Xiaohang Wang, Shengchao Yuan, Xuanhong Chen, Junjie Wang, and Xiangzhong Fang. Im2oil: Stroke-based oil painting rendering with linearly controllable fineness via adaptive sampling. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, pp. 1035–1046, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392037. doi: 10.1145/3503161.3547759. URL https://doi.org/10.1145/3503161.3547759.

Aster Van Vijle. Deep learning for overpaint detection across multi-modal imaging in historical paintings. In *HHAI 2026: Proceedings of the 5th International Conference on Hybrid Human-Artificial Intelligence*, volume 423 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2026.

Jinming Wang, Hai Wang, Hongkai Wen, Geyong Min, and Man Luo. TrajWeaver: Trajectory recovery with state propagation diffusion model. *arXiv preprint arXiv:2409.02124*, 2024.

Lvmin Zhang. Paints-undo: A base model of drawing behaviors in digital paintings. https://github.com/lllyasviel/Paints-UNDO, 2024. Last accessed 19 Aug 2026.

Amy Zhao, Guha Balakrishnan, Kathleen M. Lewis, Frédo Durand, John V. Guttag, and Adrian V. Dalca. Painting many pasts: Synthesizing time lapse videos of paintings. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8432–8442, 2020. doi: 10.1109/CVPR42600.2020.00846.

Zhengxia Zou, Tianyang Shi, Shuang Qiu, Yi Yuan, and Zhenwei Shi. Stylized neural painting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15689–15698, June 2021.

## A THE RECOVERY POSTERIOR IN BAYESIAN FORM

Section 3.1 poses recovery as sampling from $p(\mathbf{s} \mid I_T)$. Written out,

$$p(\mathbf{s} \mid I_T) \;\propto\; p(\mathbf{s})\, p(I_T \mid \mathbf{s}), \tag{1}$$

where $p(\mathbf{s})$ is a prior over plausible stroke trajectories (favoring, for instance, coarse-to-fine progressions and locally coherent ordering) and $p(I_T \mid \mathbf{s})$ is the likelihood of observing canvas $I_T$ given the trajectory, modeled as a Gaussian residual around the renderer output, $p(I_T \mid \mathbf{s}) \propto \exp\!\big(-\|I_T - R(\mathbf{s})\|^2/2\sigma^2\big)$.

We never evaluate either factor. `Pentimento` learns $p_\theta(\mathbf{s} \mid I_T)$ directly from paired (canvas, trajectory) data, so Eq. (1) fixes *what* the model approximates rather than *how* it is computed: the prior is realized by the distribution of training trajectories, the likelihood by the renderer that generated the training canvases. This is the point of difference from measurement-consistency diffusion solvers, which evaluate the forward map at inference time.

## B FULL DESIGN-LEVER SWEEP

Table 4 gives all twelve runs behind Table 2, and Fig. 6 the training curves for the eight baseline runs. Unless otherwise noted, runs use a 4-layer Transformer denoiser with $d_{\text{model}}$=192, 4 heads, and a learned 4-block convolutional canvas encoder producing 16×16 tokens at $d_{\text{ctx}}$=128, trained with AdamW (lr $10^{-4}$, weight decay 0.01, batch 8), a cosine schedule, $K$=100 DDPM training steps, and $K'$=25 DDIM sampling steps; run G swaps the denoiser for a parameter-matched 1D U-Net (Section 3.3), all else unchanged.

Table 4: The full design-lever sweep on SYNBRUSH (10% validation, $d_{\text{model}}$=192). $\tau$ is the mean over 10 validation rounds, $\tau^\star$ the best, DTW the mean (lower is better); mean ± s.d. over 3 seeds except A, H and J. Bars are proportional to $\tau$, on the scale of Table 1.

| Run | Configuration | Arch. / Cond. | Train. | Kendall $\tau$ | $\tau^\star$ | DTW |
|---|---|---|---|---|---|---|
| *Prior strength, supervision, compute, architecture, data scale* | | | | | | |
| A | $\mathcal{P}_0$ (widths only) | Trf. / global | 5k | 0.142 | 0.191 | 192 |
| B | $\mathcal{P}_1$ (+ color) | Trf. / global | 5k | 0.234±.014 | 0.277±.020 | 183 |
| C | $\mathcal{P}_2$ (+ scene regions) | Trf. / global | 5k | 0.270±.006 | 0.313±.012 | 257 |
| D | $\mathcal{P}_2$, +aux loss | Trf. / global | 5k | 0.277±.009 | 0.318±.006 | 256 |
| E | $\mathcal{P}_1$, +aux loss | Trf. / global | 5k | 0.244±.010 | 0.285±.001 | 178 |
| F | $\mathcal{P}_2$, 4× training | Trf. / global | 20k | **0.343**±.018 | **0.395**±.016 | 233 |
| G | $\mathcal{P}_2$, 1D U-Net | U-Net / bottle. | 5k | 0.051±.015 | 0.127±.033 | 259 |
| H | $\mathcal{P}_2$, 10× paintings | Trf. / global | 20k | 0.233 | 0.250 | 220 |
| *Cheap variants (Section 3.4)* | | | | | | |
| CUR | $\mathcal{P}_2$, +curriculum | Trf. / global | 5k | 0.252±.029 | 0.323±.021 | 261 |
| I | $\mathcal{P}_2$, +region attn. | Trf. / region | 5k | 0.286±.017 | 0.326±.016 | 245 |
| K | $\mathcal{P}_2$, +curric. +region | Trf. / region | 5k | 0.260±.020 | 0.328±.019 | 256 |
| J | $\mathcal{P}_2$, +region, 10× | Trf. / region | 20k | 0.249 | 0.271 | **202** |

## C REGION-ATTENTION BIAS

Let $(x_i, y_i) \in [-1, 1]^2$ be the centroid of stroke token $i$, recomputed each diffusion step from its current (noisy) endpoints, and $(u_j, v_j)$ the center of patch $j$. For head $h$ in block $\ell$ we add to the pre-softmax cross-attention logit a bias

$$b_{ij}^{(\ell,h)} \;=\; -\alpha^{(\ell,h)}(k) \cdot \big\|(x_i, y_i) - (u_j, v_j)\big\|_2^2, \tag{2}$$

with $\alpha^{(\ell,h)}(k) = \text{softplus}\Big(W_\alpha^{(\ell,h)}\, \text{cond}(k, I_T)\Big) \geq 0$ a per-head, step-conditioned learned scalar. At $\alpha = 0$ the bias in Eq. (2) vanishes and attention is global; and as $\alpha$ grows the head concentrates near the stroke's current location estimate. Zero-initializing $W_\alpha$ starts every head at $\alpha \approx 0.69$, and step-conditioning relaxes toward global attention at high noise, where the noisy location is uninformative. The cost is ∼5k parameters and +6% wall time.

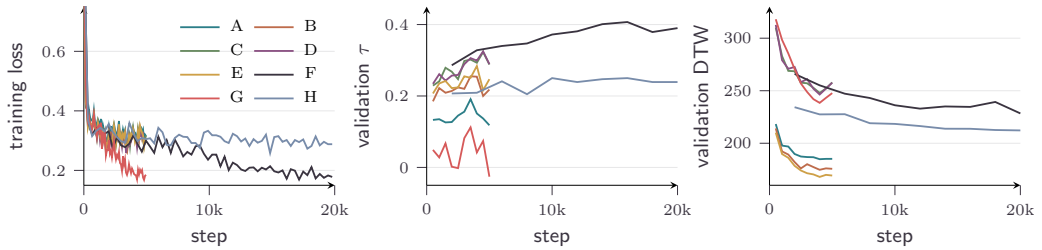



Figure 6: Design levers under global cross-attention (Section 5.2). Training loss (left), validation Kendall $\tau$ (center), and DTW (right) for the eight baseline runs (top block of Table 4). Mean $\tau$ spans $0.14$ to $0.34$ across the baseline runs, and the levers reveal no hard ceiling: recovery improves gradually with prior strength and compute. The 1D U-Net (run G) is the informative exception: it reaches *lower* training loss but collapses to $\tau \approx 0.05$.

## D SynBrush samples

Fig. 7 shows eight SynBrush paintings, one per generator configuration, un-painted stroke by stroke. The rows vary stroke count, stroke scale, ground color, palette structure and opacity, and half of them switch the scene-coherent regions of Section 4 off. ($\mathcal{P}_0$, $\mathcal{P}_1$ and $\mathcal{P}_2$ are `v0`, `v0.1` and `v0.2` in the released generator.) They deliberately span a wider parameter range than the experimental corpus: stroke counts here run from roughly 20 to over 300, against the 50–300 used for training and evaluation.

Two things are worth reading off the figure. The rows are *not* the three prior strengths $\mathcal{P}_0/\mathcal{P}_1/\mathcal{P}_2$ of Section 4, which differ only in color and region structure and are hard to tell apart by eye; they are style variants, chosen to show what the forward process can express. And reading left to right within a row is the ground truth the model is scored against: each column drops the most recently painted quarter of the strokes, so the rightmost panel is the underpainting that went down first, and the leftmost is all the model ever sees.

## E Tonal-palette control

SynBrush palettes sample visually distinct colors, which a tonal painter would not provide. We generate a graduated variant whose palettes span a narrow hue band with smoothly ramped saturation and value, fixing the generator seed so geometry, layering and count are *identical* to a matched control and only color changes (per-channel color spread falls $31\%$, $0.176 \rightarrow 0.122$). Zero-shot on the same checkpoint, $\tau$ is unchanged: $0.309 \pm 0.067$ against $0.303 \pm 0.076$. Recovery therefore does not require distinct color, consistent with Section 5.3: the coarse-to-fine width and layer structure carries it, and that survives a tonal palette.

## F Localizing the posterior's uncertainty

Section 5.4 reads the posterior at the level of a whole painting, as a single entropy number. The same quantity can be read *spatially*. For each stroke we take the spread of its rank across the $M$ posterior samples — large when the samples disagree about when it was painted, small when they agree — and splat that value back onto the region the stroke covers.

Fig. 8 does this for two held-out paintings on one shared color scale: the top row is a painting the model is confident about ($H = 0.67$ bits), the bottom row one it is not ($H = 0.89$). The left column is the finished canvas as the model receives it; the right column is that canvas greyed out with the per-stroke uncertainty laid over it, pale where the samples agree and deep warm where they disagree. The confident painting keeps pale regions where the samples agree closely; the uncertain one reads warm almost throughout. The figure localizes the model's uncertainty, which is worth having; Section 5.4 shows it should not be read as localizing what the painting itself still records.

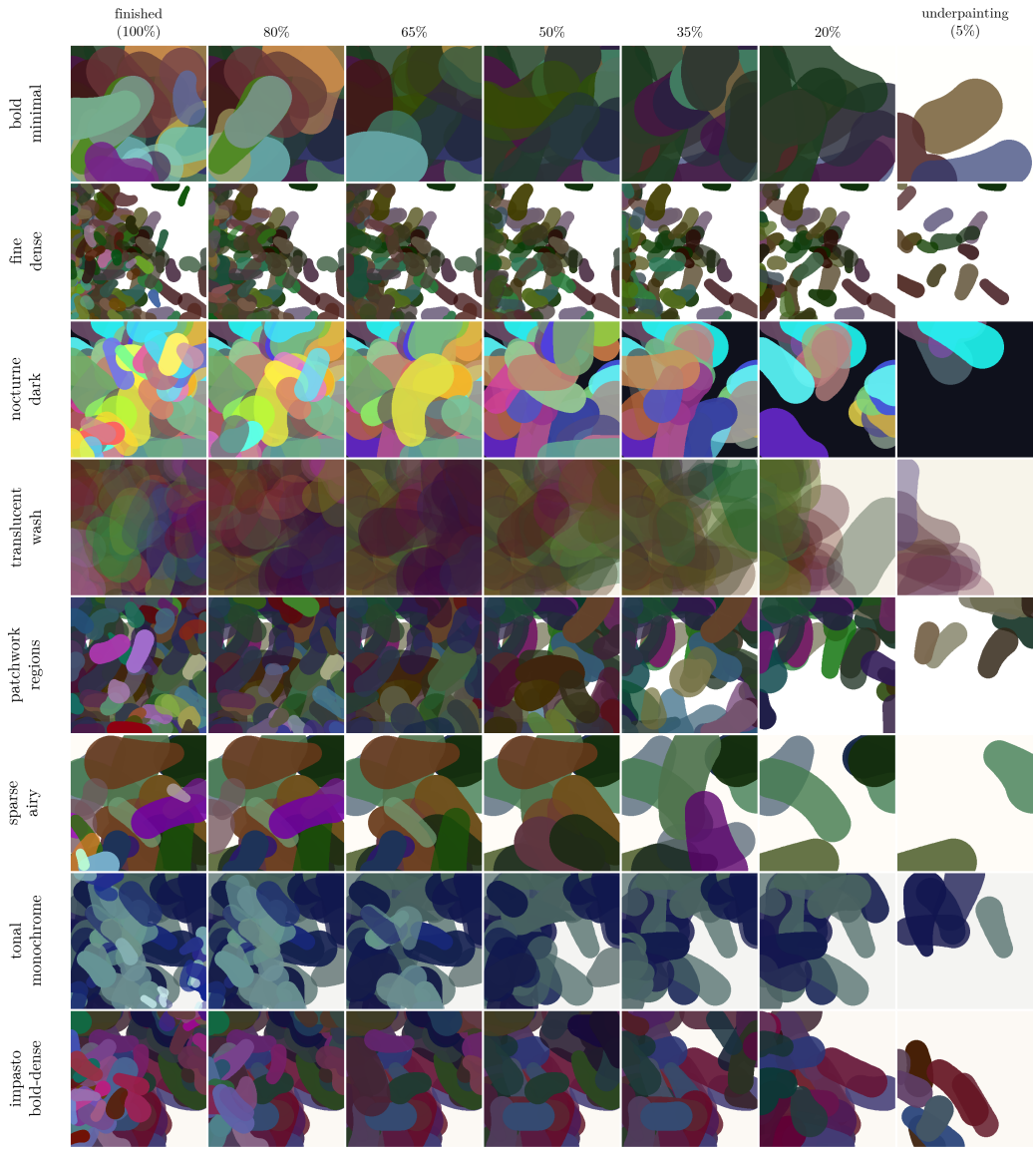



Figure 7: Eight SYNBRUSH configurations (rows), *un-painted* left to right: each column removes the most recently painted $\sim$25% of strokes, ending at the bare underpainting. Recovering that order from the leftmost panel alone is the inverse problem we study.

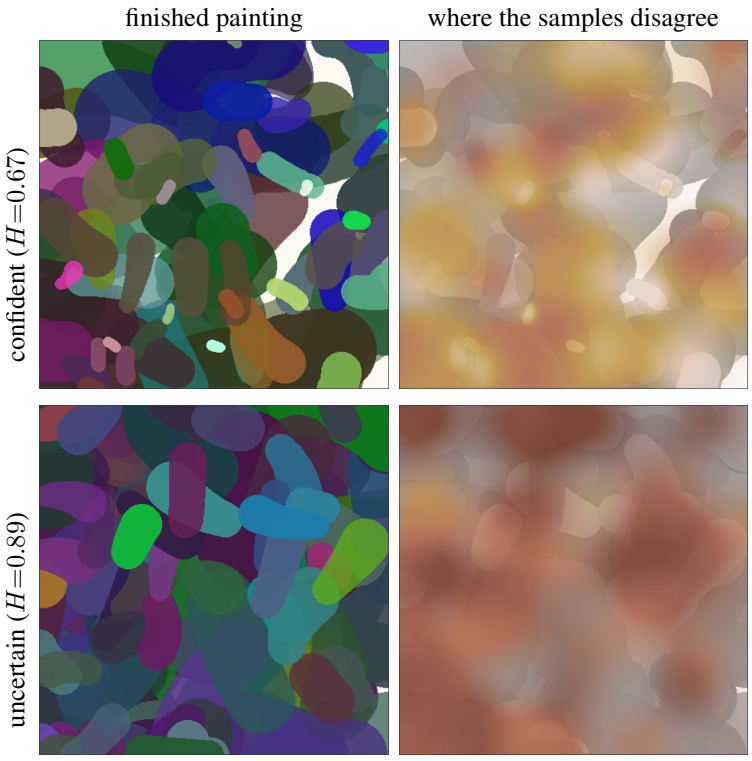



Figure 8: Where the posterior samples disagree, for one painting the model is confident about (top, order-entropy $H$=0.67 bits) and one it is not (bottom, $H$=0.89); higher $H$ means the posterior samples disagree more about the order. Right column: each stroke's rank spread over $M$=24 posterior samples, splatted back onto its own painting on a shared scale — pale where the samples agree on when a stroke was laid down, deep warm where they do not. This is the model's confidence, not a map of what the canvas determines (Section 5.4).