TEXTURE INTERPOLATION AS TEXTON MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Texture interpolation asks what should lie between two textures. Standard methods typically blend pixels or features at corresponding spatial locations, implicitly assuming that co-located elements should be mixed. For unrelated textures, however, this correspondence is arbitrary: a crack at one location has no particular relationship to a fiber at the same location in another image.

We study whether this hidden pairing choice is itself a major determinant of interpolation quality. We represent textures as collections of local appearance vectors, or textons, and replace positional pairing with appearance-aware matching while keeping the encoder, representation, and decoder fixed. This isolates the effect of correspondence from changes in model capacity or training.

Across texture pairs, appearance-aware matching consistently improves the distributional quality of interpolated midpoints over positional feature blending. Soft matching produces the strongest midpoint statistics, whereas one-to-one assignments yield more balanced transitions and closer agreement with the target endpoint. The same intervention also improves a second pretrained texture interpolator without retraining.

These results identify correspondence as a first-class design choice in texture interpolation: meaningful transitions depend not only on how strongly two textures are mixed, but also on which local appearances are mixed with one another.

1 INTRODUCTION

What should lie between cracked earth and lined fur? A useful texture transition forms a plausible material whose local appearance changes with the interpolation parameter. An RGB cross-fade often leaves two transparent patterns overlaid. Blending learned features can produce the same failure (Figure 1d). Both operations pair elements by spatial index, even though a crack at one location has no special relationship to the fiber at that location in an unrelated image. Interpolation chooses a representation and an association between its elements. Spatial pairing is useful for registered images. For unregistered textures, appearance gives a more informative association. We test this distinction by changing the partner map inside a fixed encoder and decoder (Figure 1a–c).

We demonstrate this idea on Compositional Neural Textures (CNT) (Tu et al., 2024). CNT maps a texture image to a collection of textons, where each texton is represented with a Gaussian for its spatial location and an appearance vector. A texture image is rendered from the localized Gaussian primitives with a learned decoder (Tu et al., 2024). We align the dense appearance vectors produced by splatting while retaining source conditioning: soft texton matching (entropic OT) solves one balanced entropic transport problem per pair, and each interpolation coefficient then costs one decoder pass.

Gaussian alignment explains most of the improvement over positional blending. Random permutation covers the target samples but ignores appearance, producing typical-looking images with poor set-level statistics. These controls measure the effect of matching without retraining a generator.

Similarly, Texture Mixer breaks a texture image into local features that capture texton information. It then uses interpolation-specific training (Yu et al., 2019) for texture interpolation. This gives better interpolation statistics than texton matching. Yet, we show that replacing the positional latent blend of Texture Mixer with texton-matching lowers its KID without retraining, while a one-to-one assignment raises it.

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.

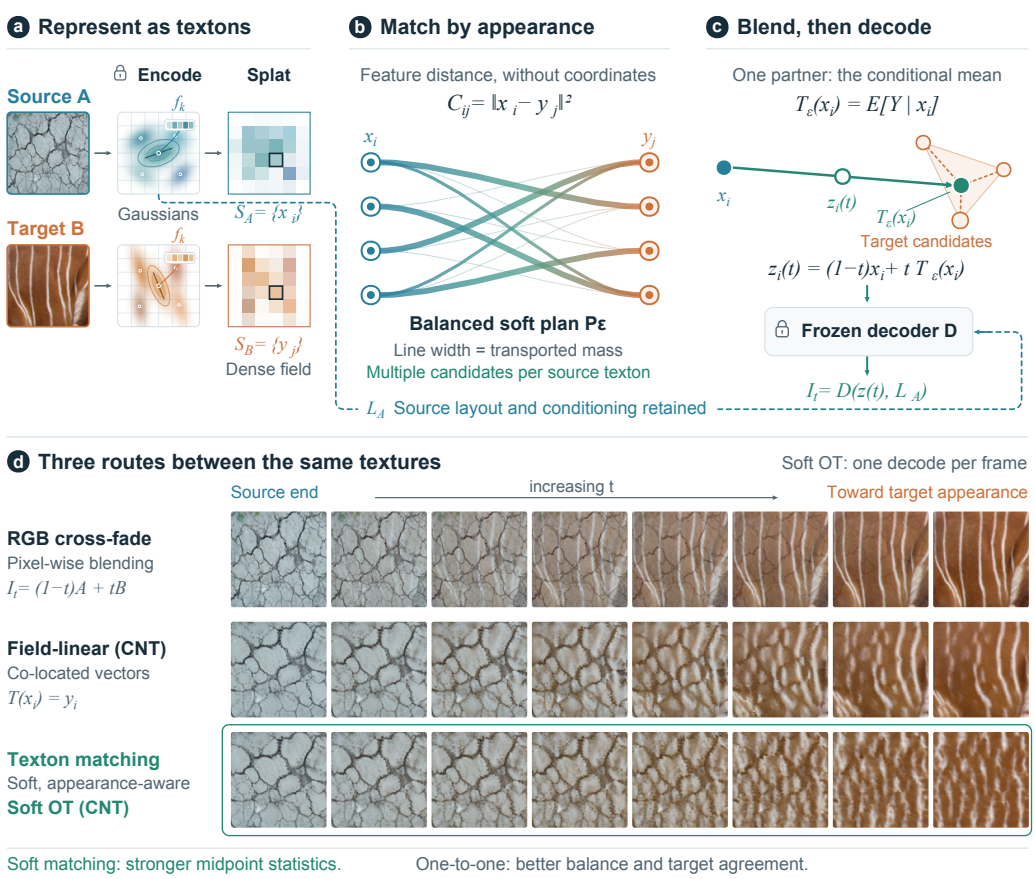

Figure 1: **Texture interpolation is a choice of partners.** Blending the same locations in unrelated textures can leave two patterns overlaid. We instead associate local appearances, keeping the learned representation and decoder fixed. (a) CNT splats localized Gaussian primitives into dense fields of local appearance vectors (textons). (b) Balanced entropic transport softly matches their vectors by appearance. (c) Each source vector moves toward the weighted mean of its matches. One partner map is reused across t, with source conditioning retained and one decoder pass per frame. (d) Cracked earth to lined fur through RGB cross-fading, positional CNT blending, and soft texton matching. The two CNT rows differ only in pairing. Across experiments, soft matching improves midpoint statistics without retraining, also in Texture Mixer, while one-to-one matching improves midpoint balance and target agreement. Our prompted comparator scores better at midpoints, but has no shared explicit path. The soft texton matching path is directed and need not reconstruct the target. Panels (a–c) are schematic.

As an alternative, we evaluate GPT-Image-1.5 that synthesizes an intermediate image from the images and a description of the midway image. This gives better results than texton matching but is not parametric, does not form a controllable path, and comes at a higher computational price.

We make three contributions:

- A texton matching formulation and controlled comparison of texture interpolation in a frozen CNT decoder, identifying appearance-aware distribution alignment as the source of most of the improvement over positional pairing, with an explicit account of barycentric contraction, target-end behavior, and the cost of reusing a coupling across frames.

- Evidence that the intervention transfers to a second pretrained decoder (Texture Mixer), with the quantitative outcome reported in Section 4.3.

- An evaluation of matching rules and learned generators that separates midpoint statistics, endpoint balance, and path behavior, and isolates the regime in which positional correspondence remains useful.

2 Related Work

Textons. Julesz proposed that texture perception rests on local elements, textons, and their statistics (Julesz, 1981). Parametric models made this operational by matching joint statistics of wavelet coefficients (Portilla & Simoncelli, 2000), and deep variants match the statistics of convolutional features (Gatys et al., 2015). CNT makes the elements explicit as localized Gaussian primitives with appearance vectors (Tu et al., 2024). Every such representation ends with a set of local descriptors, which we call textons, so interpolation has to decide which textons combine.

Texture transitions. Patch correspondence (Ruiters et al., 2010), image melding (Darabi et al., 2012), stationary Gaussian models (Xia et al., 2014), and optimized neural statistics (Vacher et al., 2020) provide different routes to texture mixing. Texture Mixer learns local and global latent spaces with interpolation-specific objectives (Yu et al., 2019). Mix-Max uses content-derived priority maps to select among overlaid texels for spatial material transitions (Fournier & Sauvage, 2024). Our study varies appearance association inside an existing decoder, including rules that preserve marginals and rules that do not.

Distribution matching. Statistical texture synthesis ranges from wavelet models (Portilla & Simoncelli, 2000) to deep feature statistics (Gatys et al., 2015). Wasserstein texture barycenters (Rabin et al., 2012), sliced Wasserstein losses (Heitz et al., 2021), and GOTEX (Houdard et al., 2023) already use transport for synthesis or mixing. Risser (2020) moves autoencoder features toward their optimal-transport targets and interpolates between them; we build on this operator and isolate the pairing rule as the only variable inside a fixed encoder and decoder. Gaussian feature transforms (Li et al., 2017), to which our Gaussian row is related through covariance alignment (see also Huang & Belongie, 2017), and transport on generative latents (Agustsson et al., 2019) also motivate our distributional controls. We compare matching rules and their endpoint effects under a shared representation and renderer, then evaluate the decoded results. Full-reference texture metrics rest on the premise this paper builds on: DISTS tolerates texture resampling by comparing spatial statistics of deep features rather than co-located values (Ding et al., 2022), and the information–estimation metric derives a distance from an image prior that reduces to a second-moment distance for Gaussian priors (Ohayon et al., 2026). Both find that low-order statistics of a good representation carry most of a texture's perceptual content, which our Gaussian and target-mean rows show from the operator's side; our object is an interpolation operator, not a distance. We systematically vary the pairing rule within a shared frozen encoder–decoder to separate the effects of appearance information, target coverage, and matching sharpness; that isolation, not the transport machinery, is the contribution.

CNT and correspondence. CNT matches latent Gaussians using centers and weights, then interpolates their geometry and appearance; it explicitly proposes softer transport as a future extension (Tu et al., 2024). We study dense splat-field vectors, weighted by their spatial coverage, rather than moving the sparse primitives. Exact OT (one-to-one) in our experiments acts on these same vectors. This separates the object being matched from the hardness of its matching rule. Related transport methods preserve content through self-similarity constraints in style transfer (Kolkin et al., 2019).

3 Interpolation with an Explicit Partner Map

Two routes lead from a pair of textures to a midpoint: a parametric operator moves along a representation under an explicit coefficient t, and a generative model describes the midpoint and samples it. This paper is about the first; Section 4.4 compares it with the second.

3.1 From Gaussian primitives to a dense appearance field

For an image A, CNT predicts about 101 localized Gaussian primitives, of which about 30 carry most of the weight, with appearance features f_k and spatial parameters. Splatting and alpha-compositing

these primitives gives a conditioning field $S_A(p) = \sum_k s_k(p) f_k$, where $s_k(p)$ is the contribution of primitive k at position p. The pretrained generator uses such fields at several resolutions (Tu et al., 2024). We operate on the finest field; each cell of the field is a local appearance vector, a texton:

$$S_A = (x_i)_{i=1}^N, \qquad S_B = (y_i)_{i=1}^N, \qquad N = 256^2, \quad x_i, y_i \in \mathbb{R}^{384}. \tag{1}$$

CNT's own interpolation matches and moves the primitives; we couple the per-cell vectors beneath them and retain the source layout. A field cell can mix several primitive features. Treating the cells as uniformly weighted samples therefore weights appearance by spatial coverage; it does not assign equal mass to each Gaussian. Coordinates are omitted from the transport cost, but the splatted fields still contain spatial correlations. Let L_A denote the retained source layout and remaining decoder conditioning.

3.2 One source-indexed path, several choices of partner

Let T assign a target appearance vector to each source texton. We define

$$z_i(t) = (1-t)x_i + tT(x_i), \qquad I_t = D\big((z_i(t))_{i=1}^N, L_A\big), \quad t \in [0,1]. \tag{2}$$

The encoder and decoder D are frozen (Figure 1a–c). Positional blending uses $T(x_i) = y_i$; an appearance-based map instead associates nearby textons or aligns the two distributions. The path keeps the partner map separate from the representation and renderer, and admits Gaussian affine maps whose outputs are not convex combinations of observed target samples.

3.3 Balanced entropic transport

For M sampled vectors per field with cost $C_{ij} = \|x_i - y_j\|^2$ and uniform sampled measures, the balanced entropic objective is

$$P_\varepsilon = \arg\min_{P \geq 0} \langle P, C\rangle - \varepsilon H(P), \quad P\mathbf{1} = P^\top \mathbf{1} = M^{-1}\mathbf{1}, \quad H(P) = -\sum_{ij} P_{ij} \log P_{ij}. \tag{3}$$

Sinkhorn iterations solve it (Cuturi, 2013). The barycentric partner is

$$T_\varepsilon(x_i) = \frac{\sum_j P_{\varepsilon,ij} y_j}{\sum_j P_{\varepsilon,ij}} = M \sum_j P_{\varepsilon,ij} y_j. \tag{4}$$

The normalized expression applies to the numerical plan; the last equality holds for its prescribed marginals. Each unsampled cell uses the partner of its nearest sampled source vector. We reuse this map for every t; we call the resulting path soft texton matching; the unregularized assignment is exact OT.

3.4 What the matching controls

We use matching for any rule that pairs textons, including soft rules that spread each texton over several counterparts; a coupling is the transport plan that defines one. Field-linear pairs each cell with its co-located texton, $T(x_i) = y_i$. Random permutation pairs through a random bijection, $T(x_i) = y_{\sigma(i)}$, an appearance-blind negative control with exact sample coverage. The target mean replaces every partner by $\bar{y}$. Nearest-neighbor matching pairs each source texton with its nearest target texton, $T(x_i) = y_{j^*(i)}$, and may reuse targets. The Gaussian W_2 rule is the quadratic-cost optimal affine map between fitted Gaussians, ridge $10^{-4}I$ (Section B.3). Entropic transport uses the plan of Equation (3) and the conditional mean of Equation (4) on 4,096 samples. Exact OT (one-to-one) is a Hungarian assignment under squared feature cost on 1,024 samples, $T(x_i) = y_{\sigma^*(i)}$, with the same nearest-neighbor extension.

The field-level controls share encoded inputs, source conditioning, and decoder. The primitive-linear baseline instead interpolates CNT elements by index before rendering, a representation-level reference distinct from the native CNT application that first matches Gaussian primitives; RGB cross-fading is an image-space reference.

3.5 PROJECTION, ENDPOINTS, AND DIRECTION

For a sampled pair $(X, Y) \sim P_\varepsilon$, the map satisfies $T_\varepsilon(X) = \mathbb{E}[Y \mid X]$. Hence

$$\mathrm{Cov}(Y) = \mathrm{Cov}(T_\varepsilon(X)) + \mathbb{E}[\mathrm{Cov}(Y \mid X)]. \tag{5}$$

The second term is positive semidefinite. A diffuse plan therefore contracts the projected target cloud; the measured default retains 38% of target variance. The same relation holds along the path. If $Z_t = (1-t)X + tY$ keeps the joint pairs and $\bar{Z}_t = (1-t)X + t\mathbb{E}[Y \mid X]$ uses one partner per source, then

$$\mathrm{Cov}(Z_t) - \mathrm{Cov}(\bar{Z}_t) = t^2\mathbb{E}[\mathrm{Cov}(Y \mid X)]. \tag{6}$$

Thus the distributional effect of projection grows toward the target end. Section B derives these identities and distinguishes the joint displacement measure from the source-indexed field used by the decoder.

At $t = 0$, Equation (2) returns CNT's reconstruction of A. At $t = 1$, it decodes transported appearance with L_A. Exact OT (one-to-one) removes conditional variance on its sampled cloud, but extension and source conditioning still affect the image endpoint; the task is a directed transition, and the two directions can differ.

With an exact quadratic-cost plan the joint displacement path is a W_2 geodesic (McCann, 1997); the projected, decoded path is deterministic in t and affine in the conditioning field once its samples and plan are fixed, and its perceptual linearity remains empirical.

4 EXPERIMENTS

4.1 PROTOCOL AND PRIMARY COMPARISON

Unless otherwise stated, the default configuration uses 4,096 sampled vectors per field, 100 Sinkhorn iterations, $\varepsilon = 0.05$ in raw squared-feature units, and float32 arithmetic. The equal-sample and Texture Mixer ablations iterate Sinkhorn to a marginal error below 10^{-3} and set ε as stated in Sections D.4 and D.5. The primary comparison uses 1,000 stratified cross-category DTD pairs (Cimpoi et al., 2014). Endpoint images are stored at 128×128 (Texture Mixer's native resolution) and bicubically upsampled to 256×256 before CNT encoding; all outputs are resized to 128 px before feature extraction. FID/KID compare midpoint sets with 2819 DTD references (Heusel et al., 2017; Bińkowski et al., 2018). FID is the community standard but biased upward at small n; KID is unbiased and admits a paired interval, so comparisons rest on KID. The first 200 pairs form the matching ablation; a separate 120-pair subset includes GPT-Image-1.5. Set-level scores are compared within each protocol.

We measure individual-sample manifold realism in Inception and DINOv2 features (Kynkäänniemi et al., 2019; Oquab et al., 2024). Midpoint balance is

$$b(o; A, B) = \left| \frac{d(o, A)}{d(o, A) + d(o, B)} - \frac{1}{2} \right|, \tag{7}$$

using DINOv2 distance; balance measures relative proximity, FID/KID agreement with the reference set. Path behavior is assessed separately.

Within the CNT decoder, soft texton matching improves FID/KID from 108.0/0.030 for field-linear to 94.7/0.021, Gaussian alignment is close at 97.9/0.024, and every soft texton-matching rule beats positional blending on FID/KID within the same decoder, and exact OT trades set-level statistics for balance and endpoint proximity. The gain is distributional: positional blending has the lower measured balance error ($b = 0.048$ against 0.081), and balance alone does not distinguish a coherent intermediate from a superposition. Texture Mixer, a separately trained interpolator, is compared in Section 4.3.

4.2 WHICH PROPERTIES OF THE PARTNER MAP MATTER?

The 200-pair study varies appearance information, marginal constraints, and assignment sharpness within the CNT encodings and renderer (Table 7, Section D.3); it and the 1,000-pair benchmark use the original rendering protocol, whereas every row of the matched-rendering experiment (Table 2) shares the field blend and pyramid construction.

Table 1: *Does replacing positional blending by texton matching improve interpolation in a fixed decoder?* Primary comparison: within each block, encoder, representation and decoder are fixed and the pairing of textons changes; the rendering-controlled comparison, with paired intervals, is Table 2 (CNT, 200 pairs) and Section D.5 (Texture Mixer, 1,000 pairs); exact OT trades set-level statistics for balance. –: not defined for the reference row.

Method	KID ↓	FID ↓	Inc. real. ↑	b ↓	ΔKID vs. positional
CNT, 1,000 pairs					
Positional blending (field-linear)	0.030	108.0	0.925	0.048	–
Texton matching: Gaussian moments	0.024	97.9	0.926	0.105	20%
Texton matching: Soft OT (ours)	**0.021**	**94.7**	0.942	0.081	**30%**
Texton matching: Exact OT	0.033	112.8	**0.951**	**0.044**	−10%
Texture Mixer, 1,000 pairs					
Positional blending (native)	0.0121	82.4	0.948	**0.056**	–
Texton matching: Target mean	**0.0090**	79.6	**0.970**	0.077	**26%**
Texton matching: Soft OT	0.0093	**79.1**	0.955	0.078	23%

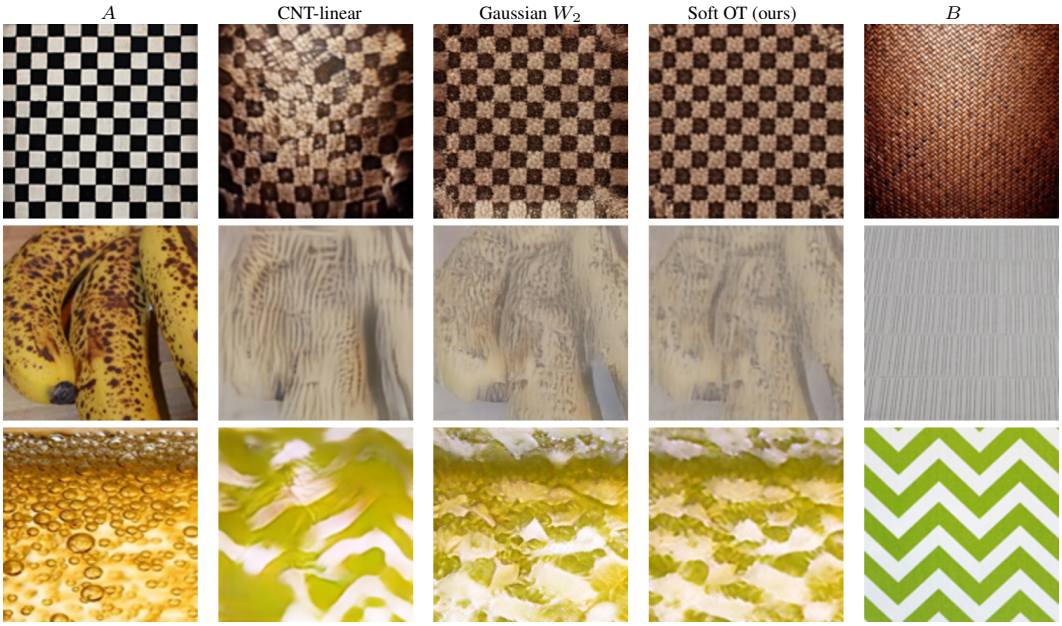

Figure 2: Texton matching within the CNT decoder: rows 2, 4, and 5 of the six-pair grid in Section E, Texture Mixer column omitted. Columns: A, CNT-linear, Gaussian W_2, Soft OT (ours), B. Distribution alignment blends appearance where the linear baseline retains endpoint structures; Gaussian and empirical transport often look similar.

Matched-rendering result. Under one rendering path and one scoring session (Table 2), the operator lowers KID from 0.0322 for positional pairing to 0.0198 (-0.0124, CI [-0.0172,-0.0076]; 38% lower KID, 8% lower FID). A one-to-one assignment does not show a statistically resolved improvement over positional pairing (-0.0022, CI [-0.0066,+0.0022]) and loses +0.0085 to the soft plan (CI [+0.0049,+0.0120]); a sharp entropic plan is marginally better than the default (-0.0018, CI [-0.0035,-0.0001]). Every soft appearance-aware rule beats positional pairing on KID, and entropic transport outperforms exact OT (one-to-one; cf. Ferradans et al., 2014), so the form of the matching matters, not appearance matching alone. The one-to-one assignment gives better balance and a smaller target gap than the soft variants (0.043 against 0.072, $p = 1.7 \times 10^{-14}$; 0.81 against 0.87, $p = 9.5 \times 10^{-13}$, both from the 200-pair matched-rendering study; field-linear's endpoint is the target field by construction; gap values in Section D.4). The regularization sweep is in Figure 7.

Table 2: *Is the gain the matching itself, and which form of matching?* One rendering path and one scoring session; the exact and 1,024-sample soft rows share the same sampled textons. Soft OT lowers KID by 38% against positional blending; exact OT does not show a statistically resolved improvement over it. Intervals and target-end gaps in Section D.4.

Rule	KID ↓	FID ↓	b ↓	Inc. real. ↑
Field-linear (same pipeline)	0.0322	189.3	0.046	0.923
Exact OT (1,024)	0.0300	183.4	**0.043**	**0.952**
Soft OT 1,024 $\varepsilon = 0.005$	**0.0197**	**174.5**	0.067	0.933
Soft OT 1,024 $\varepsilon = 0.05$	0.0216	177.6	0.072	0.931
Soft OT 4,096 (ours)	0.0198	174.9	0.076	0.942

Sample coverage alone is insufficient. Random permutation has about four times the KID of either entropic plan despite covering the same target samples, yet the highest Inception realism, 1.035 (DINO 0.926): individually typical outputs whose ensemble poorly matches the reference.

Moment alignment accounts for most of the improvement. The Gaussian map reaches 0.022 KID against 0.020 for default entropic transport, with the same ordering on 1,000 pairs: the larger change is from positional association to distribution alignment. A target-mean map also attains low KID, 0.025, with per-image realism -0.016 (CI [-0.029,-0.003]) relative to soft texton matching.

When does spatial correspondence help? Across overlapping crops, controlled warps, and unrelated textures (Figure 9), positional interpolation blurs or overlays the endpoint patterns as alignment weakens. To separate alignment from content we take 40 seamless images and let B be a cyclic shift of A, so the texture distribution is identical at every shift. Positional blending's deviation from the source reconstruction grows monotonically with the shift (0 at 0 px, 15.2 at 16, 23.9 at 64, DINOv2 distance), while the matching rules are far less sensitive to displacement, staying at 18.2–22.4, an offset to which projection, sampling, extension and rendering all contribute. On this family the regimes cross between 16 and 64 px (Figure 10): a reliable positional correspondence should be used when it exists, and texton matching when it does not.

4.3 Does the effect extend beyond CNT?

The intervention transfers. Texture Mixer's representation and decoder are not ours. It blends its 32×32 local latent positionally; replacing that pairing by texton matching of the two latent clouds, rendered by its own generator with its encodings and global-latent blend untouched, improves its set-level statistics without retraining (Table 1). The target-mean matching lowers KID by -0.0031 (CI [-0.0044,-0.0018], 26%) and the soft matching by -0.0028 (CI [-0.0039,-0.0017]) on 1000 pairs, with higher per-image realism ($p = 0.005$); the two do not differ (+0.0003, CI [-0.0006,+0.0012]). One-to-one matching raises KID (+0.0087, CI [+0.0042,+0.0132], $n = 120$) and gives the best balance. In this latent the diffuse limit is the best set-level setting; the element-level plan is what buys balance.

Appearance-aware matching also retains more source-feature spread than positional blending in DINO and SAM (Kirillov et al., 2023) feature spaces (Table 8).

4.4 How does a prompted midpoint compare?

We evaluate GPT-Image-1.5 on the same 120 DTD pairs through its image-editing interface with both endpoints supplied. A directed prompt requests the source arrangement with appearance halfway toward the target; a symmetric prompt requests a natural midpoint. Each output is one stochastic 1024^2 image resized to 128^2; intermediate coordinates are separate calls.

The directed prompt achieves KID 0.0051, compared with 0.0119 for Texture Mixer and 0.0190 for soft texton matching (Table 3); on these 120 pairs soft texton matching lowers KID against field-linear by -0.0114 (CI [-0.0174,-0.0054]). Its DINO realism is close to Texture Mixer's; the symmetric prompt is higher. Per midpoint, texton matching costs 0.21 s on one A100 against 52.4 s for optimization-based interpolation and about 38 s of API time for prompted generation; each further t is

Table 3: *How does texton matching compare with learned and prompted generators, and at what cost?* Prompted generation gives the best midpoint statistics at about 38 s per output; texton matching needs no training and pays its cost once per pair (Section D.10).

Method	FID ↓	KID ↓	Inc. real. ↑	DINO real. ↑	b ↓	s / midpoint	training
Field-linear	218.9	0.0305	0.913	0.856	**0.041**	0.18	none
Soft OT (ours)	208.2	0.0190	0.924	0.851	0.081	0.21	none
Exact OT	211.8	0.0302	0.943	0.871	0.044	0.66	none
Texture Mixer	204.0	0.0119	0.948	0.907	0.061	0.012	GAN, domain-specific
Texture Mixer + soft OT	204.0	0.0099	0.942	0.922	0.072	0.012+0.03	GAN, domain-specific
GPT directed	**193.7**	**0.0051**	**0.960**	0.906	0.054	38 (API)	proprietary
GPT symmetric	201.1	0.0087	0.953	**0.969**	0.046	38 (API)	proprietary

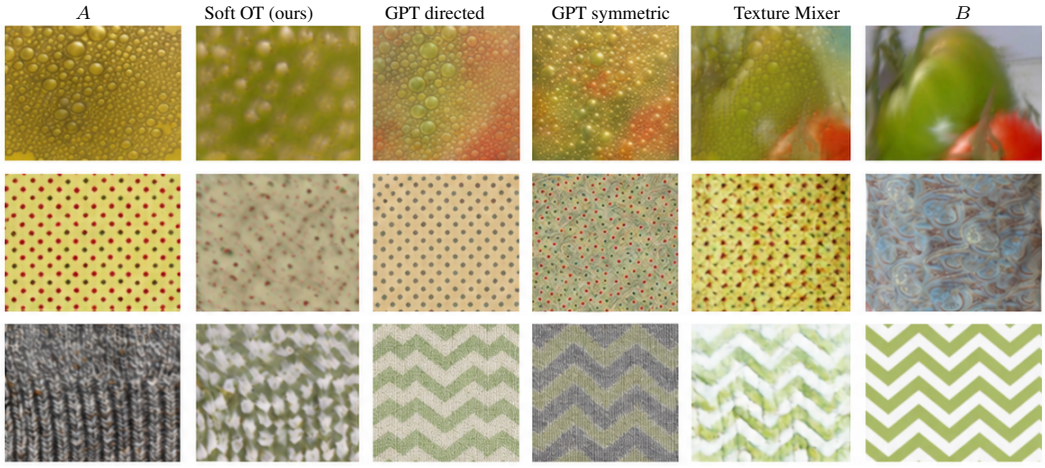

Figure 3: The first three pairs of the 120-pair comparison, in pair-ID order. Columns: A, Soft OT (ours), GPT directed, GPT symmetric, Texture Mixer, B. The prompted outputs treat the source arrangement differently from pair to pair; all eight pairs are in Figure 6.

one decoder pass (0.08 s). Prompted generation is a strong option for a single plausible intermediate, and the prompts treat source arrangement differently from pair to pair (Figure 3; all eight pairs in Figure 6).

Soft texton matching reuses one fixed source-indexed map, so its path is affine in the conditioning field; prompted generation prompts each frame independently, without a shared explicit map.

4.5 What is paid once, and what is paid per frame?

Encoding and matching are paid once per pair and each coefficient costs one renderer pass (Table 11, Section D.10). Texture Mixer's generator is faster than CNT's decoder at every t (0.003 s against 0.02 s at 128 px); texton matching requires no interpolation-specific training, training domain, or fixed output size.

4.6 Applications

Anchor-based texture mixtures. Transport maps from one anchor texture to several others give fields on the anchor's source grid, which can be blended with nonnegative weights summing to one and decoded under the anchor's conditioning (Figure 4). The construction depends on the chosen anchor and is not an exact multi-distribution barycenter.

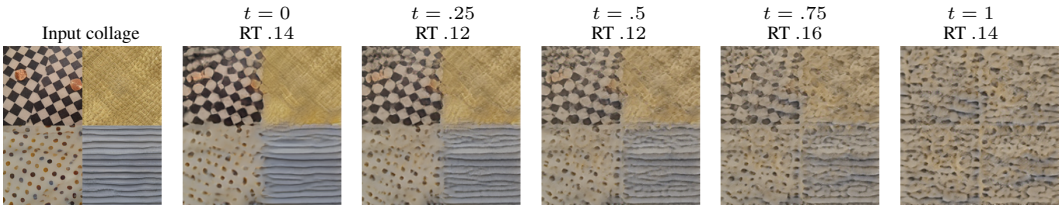

Figure 4: Four-texture mixing with a common anchor. The input collage identifies the textures; mixing strength increases from left to right; RT is the re-encoding round-trip error. Formulation in Section F.1.

Source-conditioned appearance transfer. Setting $t = 1$ in Equation (2) decodes the transported target appearance under the source conditioning L_A; reversing the two inputs exchanges those roles, so each texture can be rendered with the other's appearance (Figure 5).

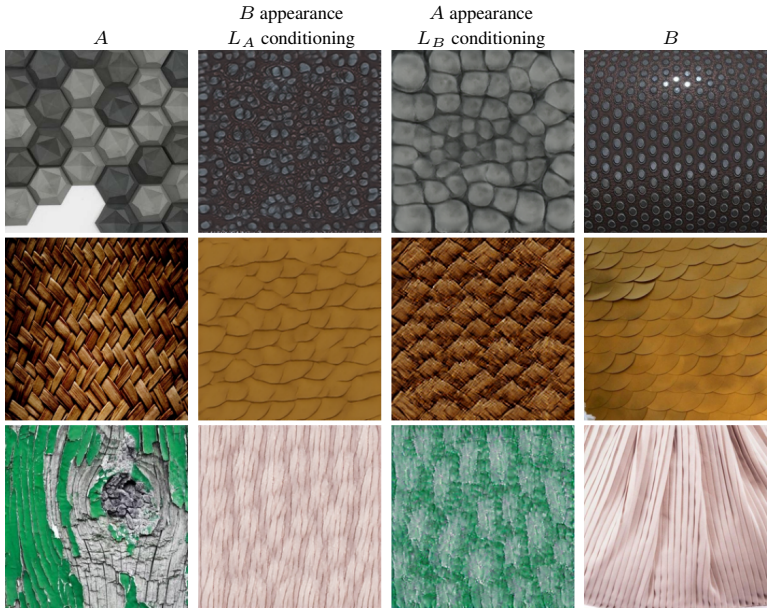

Figure 5: Source-conditioned appearance transfer in both directions, three pairs. Columns: A, B's appearance under A's conditioning, A's appearance under B's conditioning, B. Each output keeps the conditioning texture's arrangement; the operation does not impose pixelwise structure preservation.

5 Discussion and Limitations

The task is a directed transition under retained source conditioning: a diffuse barycentric map does not reproduce the target distribution exactly, retaining L_A does not guarantee pixelwise preservation of source structure, and a texture interpolated with itself is not returned unchanged. Periodic textures remain difficult because local marginals omit phase and long-range organization, and CNT reconstruction error bounds the outputs (Section D.6). No single metric establishes that both endpoints are perceptually represented, and endpoint balance rewards superposition as readily as realism rewards typicality; the generative comparator uses single outputs from fixed prompts; no human preference study is included.

6 Conclusion

Texture interpolation is a matching problem before it is a blending problem. Deciding which textons combine, inside a fixed encoder and decoder, improves midpoints in CNT and in a trained interpolator; soft matching favours ensemble statistics, one-to-one matching favours balance.

AI USE STATEMENT

Generative AI tools assisted with manuscript organization and drafting, literature lookup, analysis of experimental results, mathematical exposition, figure layout, and LaTeX build and validation scripts, and assisted with research ideation and experiment execution, including evaluation code. GPT-Image-1.5 outputs are evaluated explicitly as a generative baseline. The experimental results and image assets are our own; editorial assistance did not generate replacement measurements. Every number in the main-text tables was recomputed from the saved outputs with the released scoring code. The authors are responsible for the submitted content.

REPRODUCIBILITY STATEMENT

The appendix specifies the transport objective, numerical settings, Gaussian map, projection identities, and evaluation protocols. The accompanying package includes per-pair scores, seeds, solver diagnostics, and pair manifests for the Texture Mixer matching ablation, the equal-sample ablation, the alignment study, the timing study, and the transport-cost comparison, together with the reference image list. The older evaluations retain aggregate values only. The prompts and the call ledger are in the repository. Code, the cached features needed to recompute the main-text tables and the appendix results listed in the repository's check report, installation scripts, and two Colab notebooks that reproduce the pipeline end to end are available at https://anonymous.4open.science/r/TextureMatcher.

ETHICS STATEMENT

The study concerns texture interpolation and reports no human-subject experiment. The generated-image baseline and its role in the evaluation are identified throughout. Texture editing can alter material appearance in ways that viewers may mistake for captured imagery; edited outputs should be identified as synthetic when that distinction matters.

REFERENCES

Eirikur Agustsson, Alexander Sage, Radu Timofte, and Luc Van Gool. Optimal transport maps for distribution preserving operations on latent spaces of generative models. In *ICLR*, 2019.

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *ICLR*, 2018.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, 2013.

Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B. Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on Graphics*, 31 (4):82:1–82:10, 2012. doi: 10.1145/2185520.2185578.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE TPAMI*, 44(5):2567–2581, 2022.

Sira Ferradans, Nicolas Papadakis, Gabriel Peyré, and Jean-François Aujol. Regularized discrete optimal transport. *SIAM Journal on Imaging Sciences*, 7(3):1853–1882, 2014. doi: 10.1137/130929886.

Romain Fournier and Basile Sauvage. Mix-max: A content-aware operator for real-time texture transitions. *Computer Graphics Forum*, 43(6):e15193, 2024. doi: 10.1111/cgf.15193.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *NeurIPS*, 2015.

Eric Heitz, Kenneth Vanhoey, Thomas Chambon, and Laurent Belcour. A sliced wasserstein loss for neural texture synthesis. In *CVPR*, pp. 9412–9420, 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 *NeurIPS*, 2017.

Antoine Houdard, Arthur Leclaire, Nicolas Papadakis, and Julien Rabin. A generative model for texture synthesis based on optimal transport between feature distributions. *Journal of Mathematical Imaging and Vision*, 65(1):4–28, 2023. doi: 10.1007/s10851-022-01108-9.

Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pp. 1510–1519, 2017. doi: 10.1109/ICCV.2017.167.

Béla Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290:91–97, 1981.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *ICCV*, pp. 4015–4026, 2023.

Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *CVPR*, 2019.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019.

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *NeurIPS*, 2017.

Robert J. McCann. A convexity principle for interacting gases. *Advances in Mathematics*, 128(1): 153–179, 1997.

Guy Ohayon, Pierre-Étienne H. Fiquet, Florentin Guth, Johannes Ballé, and Eero P. Simoncelli. Learning a distance measure from the information-estimation geometry of data. In *ICLR*, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaa El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024.

Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. BEiT v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022.

Javier Portilla and Eero P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *IJCV*, 40(1):49–70, 2000.

Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision*, pp. 435–446, 2012. doi: 10.1007/978-3-642-24785-9_37.

Eric Risser. Optimal textures: Fast and robust texture synthesis and style transfer through optimal transport. *arXiv preprint arXiv:2010.14702*, 2020.

Roland Ruiters, Ralf Schnabel, and Reinhard Klein. Patch-based texture interpolation. *Computer Graphics Forum*, 29(4):1421–1429, 2010. doi: 10.1111/j.1467-8659.2010.01739.x.

Peihan Tu, Li-Yi Wei, and Matthias Zwicker. Compositional neural textures. In *SIGGRAPH Asia Conference Papers*, 2024. doi: 10.1145/3680528.3687561.

Jonathan Vacher, Aida Davila, Adam Kohn, and Ruben Coen-Cagli. Texture interpolation for probing visual perception. In *NeurIPS*, 2020.

Gui-Song Xia, Sira Ferradans, Gabriel Peyré, and Jean-François Aujol. Synthesizing and mixing stationary gaussian texture models. *SIAM Journal on Imaging Sciences*, 7(1):476–508, 2014. doi: 10.1137/130918010.

Ning Yu, Connelly Barnes, Eli Shechtman, Sohrab Amirghodsi, and Michal Lukáč. Texture mixer: A network for controllable synthesis and interpolation of texture. In *CVPR*, 2019.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

A Protocol and Baseline Definitions

The primary benchmark uses 1,000 stratified cross-category pairs from DTD, which contains 5,640 images in 47 categories (Cimpoi et al., 2014). Endpoint images are stored at 128×128 (Texture Mixer's native resolution) and bicubically upsampled to 256×256 before CNT encoding; all outputs are resized to 128 px before feature extraction. This holds for the primary, 200-pair matching, and 120-pair generative comparisons. The reference pool contains 2819 DTD images. FID and KID compare midpoint sets with that pool (Heusel et al., 2017; Bińkowski et al., 2018); their absolute values change with the sample count and reference protocol. The older evaluations below retain their own sample counts and reference definitions.

Two positional baselines. Field-linear blends co-located dense CNT feature vectors; our experiment code names it "pixel-linear." Primitive-linear blends the indexed CNT elements and is named "texton-linear" there. The latter is distinct from CNT's published interpolation application, which first matches Gaussians by the sum of differences between centers and weights, then interpolates the matched parameters (Tu et al., 2024), Appendix B.5. An RGB cross-fade blends image pixels directly. It appears as a separate reference in the teaser and parametric-texture experiment. These are three distinct operations.

Transport configuration. The finest conditioning field has 256^2 cells of dimension 384. The default uses $M = 4096$ uniform samples per field, raw feature vectors, uniform masses, squared Euclidean cost, $\varepsilon = 0.05$, and 100 Sinkhorn iterations in float32 on the GPU. The sampled barycentric map is extended through nearest sampled source vectors. The hard implementation uses 1,024 samples. These different sample counts are part of the measured implementations. Gaussian transport uses covariance ridge $10^{-4}I$. Our runs draw the samples at random, and we report the resulting repeat variation below; fixing a sampled map makes its subsequent path deterministic.

Learned and optimization-based references. Texture Mixer uses the released earth-texture checkpoint and its fixed 128×128 generator (Yu et al., 2019). DTD is outside that specific training domain. The Vacher variants optimize Gram, Gaussian, or Wasserstein statistics (Vacher et al., 2020). We run them for 30 iterations; the published configuration uses 150. GPT-Image-1.5 is a proprietary pretrained generator, evaluated with the image-editing interface and two reference inputs. No interpolation-specific training is performed for the CNT operator, which itself relies on pretrained CNT weights. Timing was re-measured on the 120 pairs under one protocol for all methods (encode logged once per pair, one renderer pass per t); earlier drafts used a 20-pair run.

KID estimators. The CNT block of the 1,000-pair table and the matching-rules table report KID as the mean over 20 reference subsets; every other KID, and every ΔKID, uses the full reference with the trace-corrected self-term.

What the metrics condition on. Manifold realism is the per-image score of Kynkäänniemi et al. (2019), evaluated in Inception or DINOv2 features (Oquab et al., 2024). It measures proximity to reference-image neighborhoods without conditioning on the selected texture pair. We report medians over five reference-manifold draws; the empirical realism resolution of the observer experiments is .036, which is not a confidence interval for FID, KID, or fidelity. DINO balance is the absolute displacement of the endpoint-distance ratio from one half. LPIPS path measurements use perceptual image distance (Zhang et al., 2018). Portilla–Simoncelli and BEiTv2 serve as additional source-fidelity observers (Portilla & Simoncelli, 2000; Peng et al., 2022).

Available evidence. The accompanying package includes per-pair scores, seeds, solver diagnostics, and pair manifests for the Texture Mixer matching ablation, the equal-sample ablation, the alignment study, the timing study, and the transport-cost comparison, together with the reference image list. The older evaluations retain aggregate values only, with confidence intervals where we computed them. The prompts and the call ledger are in the repository. The full scoring code is available at the same address.

B Operator Derivations

B.1 Joint transport and deterministic projection

For equal-weight sampled clouds, a balanced coupling P defines random variables (X, Y) with the prescribed empirical marginals. A row-normalized plan defines $T(X) = \mathbb{E}[Y \mid X]$. With $\epsilon_Y = Y - \mathbb{E}[Y \mid X]$, we have $\mathbb{E}[\epsilon_Y \mid X] = 0$. Therefore $T(X)$ and ϵ_Y have zero cross-covariance, giving

$$\mathrm{Cov}(Y) = \mathrm{Cov}(T(X)) + \mathbb{E}[\mathrm{Cov}(Y \mid X)]. \tag{8}$$

The second term is positive semidefinite, so $\mathrm{Cov}(T(X)) \preceq \mathrm{Cov}(Y)$. Equality holds when the conditional variance vanishes in every direction.

The joint displacement variable $Z_t = (1-t)X + tY$ satisfies

$$\mathbb{E}[Z_t \mid X] = (1-t)X + tT(X) = \bar{Z}_t, \qquad \mathrm{Cov}(Z_t \mid X) = t^2\,\mathrm{Cov}(Y \mid X). \tag{9}$$

Applying total covariance again gives

$$\mathrm{Cov}(Z_t) - \mathrm{Cov}(\bar{Z}_t) = t^2\mathbb{E}[\mathrm{Cov}(Y \mid X)]. \tag{10}$$

This identity describes the cost of replacing the conditional distribution by its mean before decoding. It imposes no corresponding equality on decoded image statistics, since the decoder is nonlinear and uses additional conditioning.

The joint displacement measure $((1-t)x + ty)_{\#}P$ reaches both empirical marginals. A diffuse plan can give it up to M^2 support points, whereas the decoder input requires one vector per source grid cell. Barycentric projection supplies that field but changes the target distribution. With an unregularized quadratic-cost optimal plan, joint displacement gives a W_2 geodesic (McCann, 1997). Entropic regularization, projection, nearest-neighbor extension, and nonlinear decoding each limit the direct use of this geometric interpretation for images.

B.2 What quadratic cost rewards

For fixed marginals with means m_A, m_B,

$$\mathbb{E}\|X - Y\|^2 = \|m_A - m_B\|^2 + \mathbb{E}\|X - m_A\|^2 + \mathbb{E}\|Y - m_B\|^2 - 2\mathbb{E}\langle X - m_A, Y - m_B\rangle. \tag{11}$$

Unregularized quadratic transport maximizes the cross term. Entropic transport trades that alignment against plan entropy. A random permutation preserves the marginals but does not optimize their feature alignment. Coverage and appearance information are therefore separate controls. Perceptual quality need not follow feature alignment.

B.3 Gaussian transport

For fitted Gaussian models $\mathcal{N}(m_A, \Sigma_A)$ and $\mathcal{N}(m_B, \Sigma_B)$ with positive-definite covariances, the quadratic-cost optimal affine map is

$$\begin{aligned} T_G(x) &= m_B + A(x - m_A), \\ A &= \Sigma_A^{-1/2}(\Sigma_A^{1/2}\Sigma_B\Sigma_A^{1/2})^{1/2}\Sigma_A^{-1/2}. \end{aligned} \tag{12}$$

The implementation adds $10^{-4}I$ to the covariances and clips negative eigenvalues from floating-point error. CNT field vectors are mixtures of a much smaller number of primitive features, so covariance regularization can affect this map. The affine path is the W_2 geodesic for the regularized Gaussian models. The empirical input cloud need not be Gaussian, and the decoded path retains source conditioning. The Gaussian row is a moment-alignment baseline for the empirical operator.

B.4 Exact OT (one-to-one)

A minimum-cost assignment on 1,024 equally weighted source and target vectors is a permutation. Its conditional variance is zero on that sampled cloud. Nearest-neighbor extension to the dense field can repeat assigned targets, so sample-level exactness does not imply exact full-field target recovery. The exact-OT target gap is 0.80 on the 1,000-pair canonical benchmark; the 200-pair matching-rules study reports 0.81 for exact OT and 0.87 for the default soft plan. A symmetric (OT-sym) variant averaging the two directed paths reduced order dependence but lowered recall and set-level statistics, and is not used.

C GENERATIVE COMPARISON

The 120-pair comparison uses a category-stratified subset and the same 128^2 scoring resolution for every method. GPT-Image-1.5 receives both endpoint images in a fixed order and produces one 1024^2 output per prompt and pair. The directed prompt requests source layout and halfway target appearance; the symmetric prompt requests a natural midpoint with no source preference. Both prompts are reproduced below, byte for byte.

> **GPT symmetric** (prompt P1):
>
> You are given exactly two reference images in order: reference image 1 is texture A and reference image 2 is texture B. Generate one single square image M representing a coherent, plausible perceptual texture midpoint between A and B at t = 0.50.
>
> M must be one internally consistent texture that could plausibly exist on its own. Interpolate texture-level properties such as periodicity, orientation, topology, motif or element shape, spatial scale, density, regularity, material appearance, local contrast, and color distribution. Preserve meaningful evidence of both endpoints and aim for approximately equal perceptual texture distance from A and B. Prefer structural and statistical texture cues over semantic object identity.
>
> Do not make a pixel-space blend, alpha blend, cross-fade, transparent overlay, split image, side-by-side composition, collage, or simple average of colors. Do not merely paste motifs from A and B together. Do not include A, B, labels, text, borders, margins, or a presentation panel. Output only M, edge-to-edge. The definition is symmetric: swapping the order of A and B should not change the intended midpoint.

> **GPT directed** (prompt P2):
>
> You are given exactly two reference images in order: reference image 1 is source texture A and reference image 2 is target-appearance texture B. Generate one single square midpoint texture M at t = 0.50 for a directed source-layout appearance interpolation from A toward B.
>
> Preserve the dominant spatial organization, layout, orientation, and large-scale arrangement of A, while moving the appearance of its local texture elements halfway toward B. Interpolate local motif shape, scale, density, regularity, material, local contrast, and color statistics toward B. The result must be one coherent, plausible texture, not two overlaid patterns. Semantic objects that are incidental to texture should not dominate the interpolation.
>
> Do not make a pixel-space blend, alpha blend, cross-fade, transparent overlay, split image, side-by-side composition, collage, or simple average of colors. Do not include A, B, labels, text, borders, margins, or a presentation panel. Output only M, edge-to-edge. This task is directed: A supplies layout and B supplies the target appearance.

Model gpt-image-1.5 through the image-editing interface, quality high, input fidelity high, two ordered reference images (A then B), one 1024×1024 output per call; 240 scored calls, median latency 38 s.

Table 4: Complete 120-pair results. FID is descriptive at this sample count. Bold marks the best value in each column.

Method	FID ↓	KID ↓	Inc. real. ↑	DINO real. ↑	Balance ↓
Field-linear	218.9	0.0305	0.913	0.856	0.041
Primitive-linear	247.9	0.0504	0.897	0.849	**0.040**
Gaussian W_2	216.2	0.0253	0.919	0.847	0.103
Soft OT (ours)	208.2	0.0190	0.924	0.851	0.081
Exact OT	211.8	0.0302	0.943	0.871	0.044
Texture Mixer	204.0	0.0119	0.948	0.907	0.061
GPT, directed	**193.7**	**0.0051**	**0.960**	0.906	0.054
GPT, symmetric	201.1	0.0087	0.953	**0.969**	0.046

The directed prompt's median distance-ratio placement is 0.456. This is a signed location along the endpoint-distance axis, distinct from the absolute balance error in the table. Paired DINO realism differences from the CNT-based variants range from +0.035 to +0.074 ($p \leq 10^{-5}$); the difference from Texture Mixer is -0.021 and not significant. Paired differences and differences between separately summarized table columns need not coincide. Soft texton matching reuses a fixed

source-indexed map and has explicit affine dependence on t before decoding, while independently prompted frames carry no such structure; its dependence on input order remains. The API latency is approximately 38 s per output, versus 0.21 s for a local soft texton matching midpoint in the A100 timing study.



Figure 6: The first eight pairs, in pair-ID order. Columns are A, Soft OT (ours), GPT directed, GPT symmetric, Texture Mixer, and B; the main text shows the first three rows. The directed prompt handles stochastic appearance, structured motifs, and semantic source content differently; the examples do not establish a pixelwise structure guarantee for either method.

D Additional Quantitative Evaluations

D.1 Optimization-based comparison

Table 5: The 50-pair comparison with optimization-based interpolation, scored at 128 px. Timing is on one A100, with 30 iterations for Vacher variants. FID at this sample count is descriptive. –: not defined, no interpolation is performed.

Method	FID ↓	Midpoint (s)	11-frame path (s)
Soft OT (ours)	268.3	0.21	1.02
Field-linear	284.6	0.18	0.99
Vacher, Gram	318.4	13.5	148.5
Vacher, Gaussian	314.3	15.0	165.3
Vacher, Wasserstein	313.6	52.4	576.6
CNT reconstruction	273.9	–	–

We find no significant paired difference in per-sample realism and midpoint balance (all Wilcoxon $p > .25$). This small-sample result does not establish statistical equivalence. The runtime difference follows from reusing the transport map across frames, while the optimization-based method solves for each output independently. The Wasserstein implementation takes 52.1, 52.4, and 56.5 seconds at 128, 256, and 512 pixels, respectively.

D.2 Matching on CNT's Gaussian primitives

CNT predicts about 101 Gaussian primitives per texture, of which about 30 carry most of the weight. On 120 pairs we pair the two primitive sets with a Hungarian assignment under a position cost, an appearance cost, or their mixture, and either freeze the source layout or let the matched primitives move; the encoder, primitives, interpolation, and decoder are otherwise those of the main protocol (Table 6). With a frozen layout the cost is a null result (KID 0.030 / 0.030 / 0.027 for position / appearance / mixed; appearance against position +0.0001, CI [-0.0041,+0.0043]), and reassigning every pairing changes per-image realism by 0.001. With a moving layout, appearance matching is the worst choice (0.080 against 0.036, ΔKID +0.0433, CI [+0.0333,+0.0534]), because look-alike partners are far apart and the moved primitives superpose. Intervals are paired leave-one-pair-out jackknife, 95% CI. A soft matching of the same primitives reaches 0.016 KID, against 0.019 for the dense operator and 0.030 for every one-to-one rule. The matching's softness and the anchoring to the source layout, not the cue used to pair, separate a coherent intermediate from a superposition.

Table 6: KID for one-to-one matchings of CNT's Gaussian primitives, 120 pairs, canonical scoring. Rows fix or move the source layout; columns give the pairing cost.

Layout	Position cost	Appearance cost	Mixed cost
Frozen (source)	0.030	0.030	0.027
Moving (matched)	0.036	0.080	0.046

D.3 Matching rules under the original rendering protocol

The 200-pair matching-rule study of the main text, scored under the original rendering protocol, is tabulated here.

D.4 Equal-sample comparison of hard and soft matchings

The main-text hard and entropic rows use 1,024 and 4,096 samples. The matched-rendering ablation (Table 2) repeats the 200-pair study under one rendering path. Exact OT (one-to-one) and the two 1,024-sample entropic rows share the same sampled clouds; field-linear uses positional correspondence on the full co-located fields; the 4,096-sample operator uses its own sample; all rows share the field-blend and pyramid construction and were scored in one session. Sinkhorn is

Table 7: Matching rules on the same 200 pairs. Random permutation preserves the sampled marginals but ignores appearance. Mean and nearest-neighbor maps test averaging and target reuse. Lower KID/FID is better; gaps are normalized target-end distances (the positional rows reach the target field by construction, so their gap is decoder error only). Bold marks the best value per column; this table uses the original rendering protocol, unlike the matched-rendering table in the main text. Bottom line: every soft appearance-aware rule beats positional pairing on KID, and entropic transport outperforms exact OT (one-to-one) in the matched-rendering comparison (Table 2), so the form of the matching matters, not appearance information alone.

Rule	KID ↓	FID ↓	b ↓	Target gap ↓	Inc. real. ↑
Random permutation	0.077	224.3	**0.035**	1.00	**1.035**
Target mean	0.025	183.7	0.082	0.91	0.921
Field-linear	0.031	188.7	0.044	0.29	0.923
Primitive-linear	0.043	211.8	**0.035**	**0.00**	0.895
Gaussian W_2	0.022	177.6	0.107	0.81	0.928
Nearest neighbor	0.027	182.4	0.046	0.81	0.951
Exact OT	0.030	184.2	0.043	0.81	0.953
Soft OT (ours), $\varepsilon = .05$	0.020	174.9	0.081	0.87	0.933
Soft OT (ours), $\varepsilon = .005$	**0.019**	**173.7**	0.083	0.84	0.942

iterated until the marginal error is below 10^{-3}. The sharp plan retains 0.88 of the target variance and the default plan 0.38. Against field-linear on the same pipeline (KID 0.0322), the default plan gives -0.0107 (CI [-0.0154,-0.0059]), the 4,096-sample operator -0.0124 (CI [-0.0172,-0.0076]), and exact OT (one-to-one) -0.0022 (CI [-0.0066,+0.0022]). The sharp plan's difference from the default plan is -0.0018 (CI [-0.0035,-0.0001]); the difference of exact OT (one-to-one) from the sharp plan is +0.0103 (CI [+0.0070,+0.0135]) and from the default plan +0.0085 (CI [+0.0049,+0.0120]). All ΔKID intervals are paired leave-one-pair-out jackknife, 95% CI. The balance and target-gap differences between hard and default are significant at $p = 1.7 \times 10^{-14}$ and $p = 9.5 \times 10^{-13}$ (paired Wilcoxon). In this matched-rendering study the normalized target-end gap at $t = 1$ is 0.81 for exact OT, 0.84 for the sharp plan and 0.87 for the default plan; it is not defined for field-linear, whose endpoint is the target field by construction, and was not measured for the 4,096-sample operator. Re-encoding intermediates at $t \in \{.25, .5, .75\}$ gives error 0.17 against a 0.13 reconstruction floor, versus 0.32 for field-linear blending.

Table 8: Feature spread retained at the midpoint, $n = 40$. The common direction across representations supports the matching mechanism. In a separate experiment decoding DINO features to images, appearance-aware matching improves LPIPS midpoint placement on 37 of the $n = 40$ pairs.

Representation	Positional blend	Appearance-aware matching
DINO	60%	66%
SAM	57%	71%
CNT primitives	60%	87%
CNT field	67%	99%

D.5 Matching in Texture Mixer's local latent

Texture Mixer interpolates a 32×32 local latent positionally and blends a global latent. For the matching ablation (Table 1) we keep its encodings, global-latent blend, and generator, and replace only the pairing of the two local-latent clouds. The entropic regularization is set as a multiple of the median pairwise cost, $\varepsilon = 0.03\times$median, and Sinkhorn is run until the marginal error is below 10^{-3}. At that setting the projected cloud retains 0.33 of the target variance; in the diffuse limit the coupling reduces to the target-mean rule; at the reported setting it retains 0.33 of the target variance, and exact OT (one-to-one) retains the full sampled target cloud. Scoring is identical to the main protocol at 128 px against the same reference pool: the native, target-mean and soft rows are scored on 1000 pairs, the one-to-one row on the 120-pair subset. ΔKID against the native positional blend is a paired leave-one-pair-out jackknife, 95% CI, for every row; on 1000 pairs the soft matching differs from the

target mean by +0.0003 (CI [-0.0006,+0.0012]), and its per-image realism exceeds the native blend's ($p = 0.005$).

D.6 Encoding ceiling

This study uses a separate reference of 5,640 native 128-px DTD windows. One CNT encode–decode loop has LPIPS 0.21 on the 120-pair set. The KID of the originals is 0.009, measured against this study's 5,640-window reference, so these values are not comparable to Table 3's; their reconstructions reach 0.014, and a second reconstruction 0.025. Soft texton matching midpoints are at 0.023, below the double-reconstruction level. Per-image realism moves from 1.00 for originals to 0.90 for reconstructions and 0.85 for midpoints. Per-pair reconstruction error does not predict midpoint quality (Spearman 0.08).

D.7 Observer dependence and negative controls

An older 100-pair realism experiment includes an LPIPS-optimized control. It attains a near-perfect LPIPS midpoint ratio (.50), despite producing a degenerate image. Its realism is below that of soft texton matching, though not below every baseline. The random-pairing control in the main paper has high per-image realism despite poor set-level statistics, illustrating the converse problem.

Table 9: Per-image realism, $n = 100$. $R \geq 1$ denotes inclusion in the reference manifold estimate. The paired win column compares soft texton matching to each row; –: not defined for its own row.

Method	Mean R	Fraction $R \geq 1$	Soft OT wins (p)
Primitive-linear	.920	14%	65/100 (2.6×10^{-4})
Field-linear	.952	29%	54/100 (.21)
LPIPS control	.938	25%	62/100 (.017)
Soft OT (ours)	.960	41%	–

In a separate 1,000-pair observer evaluation, Texture Mixer and soft texton matching have Inception realism .963 and .962 (paired $p = .28$), respectively. Their Portilla–Simoncelli fidelity scores are $-.575$ and $-.585$ ($p = .18$), while DINO realism is .948 versus .904 and BEiTv2 fidelity is $-.875$ versus $-.906$. These measurements belong to the earlier observer protocol and remain separate from the canonical rerun.

D.8 Endpoint direction

The operator is directed: at $t = 0$ the feature field is exactly the source field, so a nonzero image gap there measures the decoder's reconstruction error, and swapping the two inputs changes the output because the target's appearance is decoded under the source's conditioning.

D.9 Regularization and repeated application

For the 20-pair entropy sweep, mean row entropy changes from 7.09 to 8.31 nats between $\varepsilon = .005$ and .5, against a uniform ceiling of 8.32. Projected target variance changes from 88% to 1%. The extreme-midpoint LPIPS difference is .21 (IQR .17–.25), compared with an approximately .15 resampling floor. These are midpoint measurements; the path covariance identity explains why projection can have larger effects toward $t = 1$.

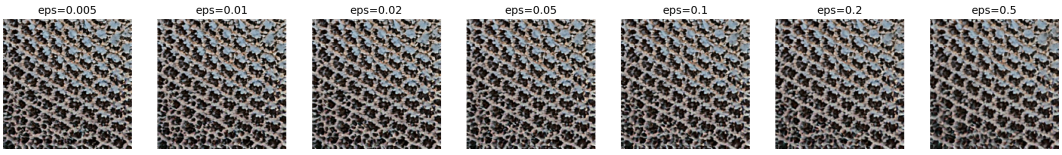

Figure 7: Complete regularization sweep for one pair. The comparatively small visual change at the midpoint accompanies much larger changes in the entropic plan and projected feature variance.

Table 10: Eight repeated operations on 24 pairs. Self: use the current image on both sides. Fixed: retain a fixed second endpoint. Positive realism drop denotes deterioration under that observer; negative values denote an increase. –: not defined, no second endpoint.

Method	Realism drop (self)	Realism drop (fixed)	Self drift
Autoencoder only	.105	–	61.9
Primitive-linear	.024	.025	61.0
Field-linear	.024	.029	61.0
Texture Mixer	−.019	.006	59.7
Soft OT (ours)	−.041	−.077	62.8

Autoencoder-only and transport self loops have similar DINO drift. Increasing realism along these trajectories therefore does not imply preservation of the original image; the outputs can move toward smoother, typical textures.

D.10 RUNTIME

Table 11 separates what is paid once per pair from what each interpolation coefficient costs. The optimization-based comparison pays for each frame separately, and the GPT entry is API latency rather than a hardware-normalized runtime.

Table 11: Recorded stage timings in seconds on one A100: 120 pairs for the CNT and Texture Mixer rows; the Vacher et al. timing comes from a 20-pair run on the same hardware with its 30-iteration configuration; GPT is the median API latency over the scored calls. Encoding is separate; each interpolation coefficient is one renderer pass. CNT decoding is at 256 px (128 px in parentheses); Texture Mixer is at 128 px. –: no such stage for this method.

Method	Encode, once per pair	Couple, once per pair	Each requested t
Field-linear	0.09	–	0.08 (0.02) decode
Soft OT (ours)	0.09	0.03	0.08 (0.02) decode
Exact OT	0.09	0.48	0.08 (0.02) decode
Texture Mixer	0.009	–	0.003 generate
Vacher W_2	–	–	52.4 optimize
GPT-Image-1.5	–	–	$\sim$ 38 API call

E Qualitative Comparisons and Applications

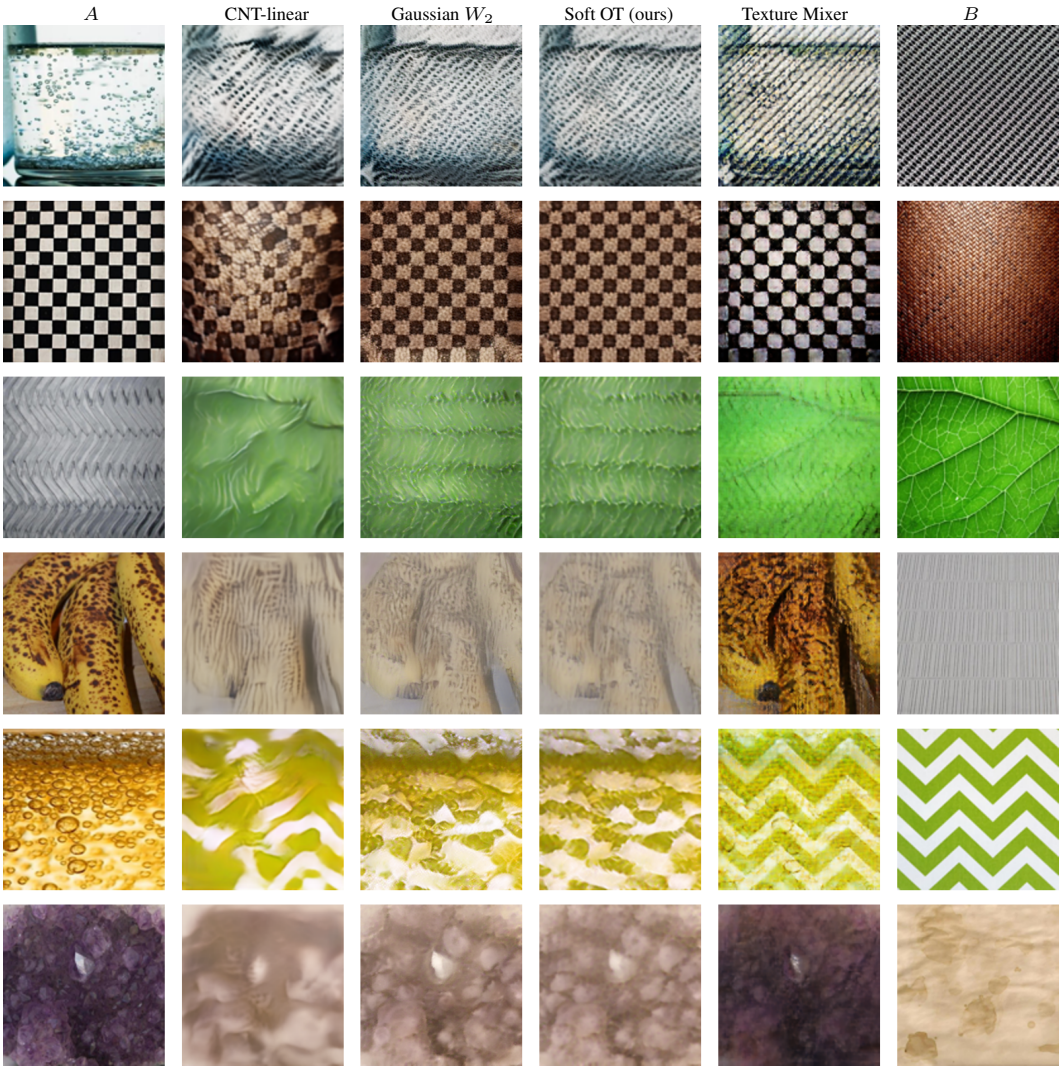

Figure 8: The full six-pair qualitative comparison. The main paper selects rows 2, 4, and 5. The Gaussian and empirical columns often resemble one another, while the relative appearance of all methods depends on the pair.

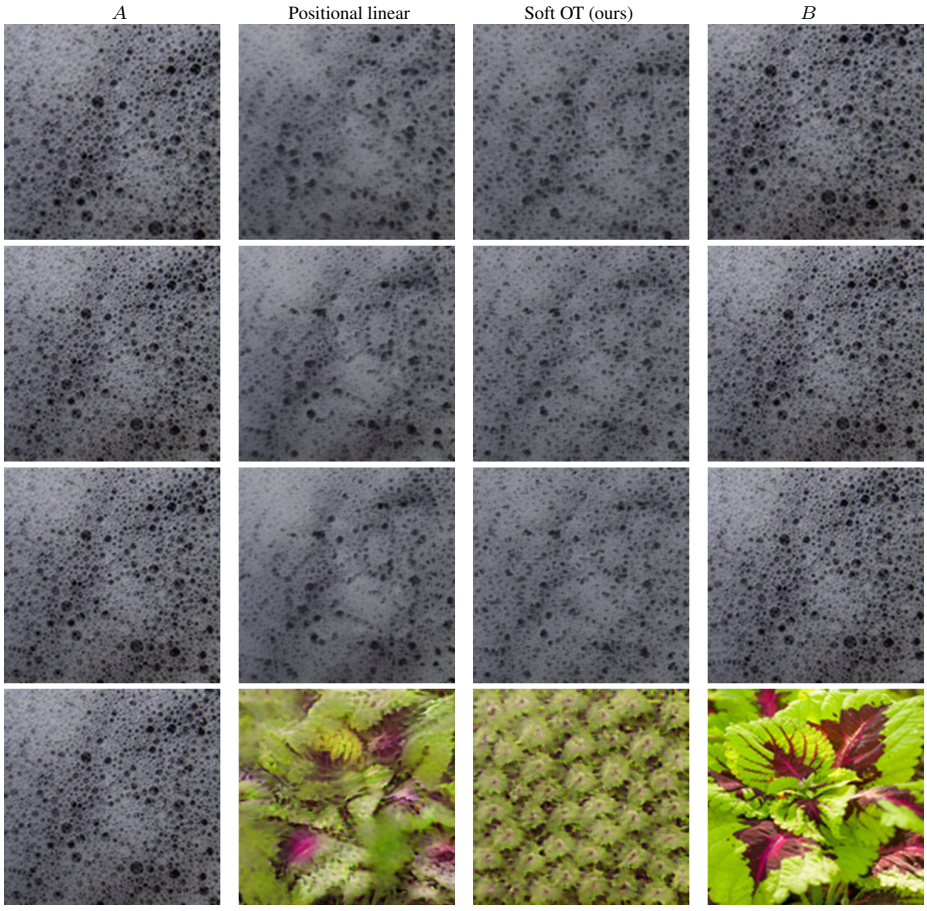

Figure 9: Full correspondence progression. Columns: A, positional linear, Soft OT (ours), B. Rows: overlapping crops, a 2° warp, an 8° warp, and an unrelated pair. Spatial pairing remains useful in the aligned rows; appearance matching gives a coherent intermediate in the unrelated pair.

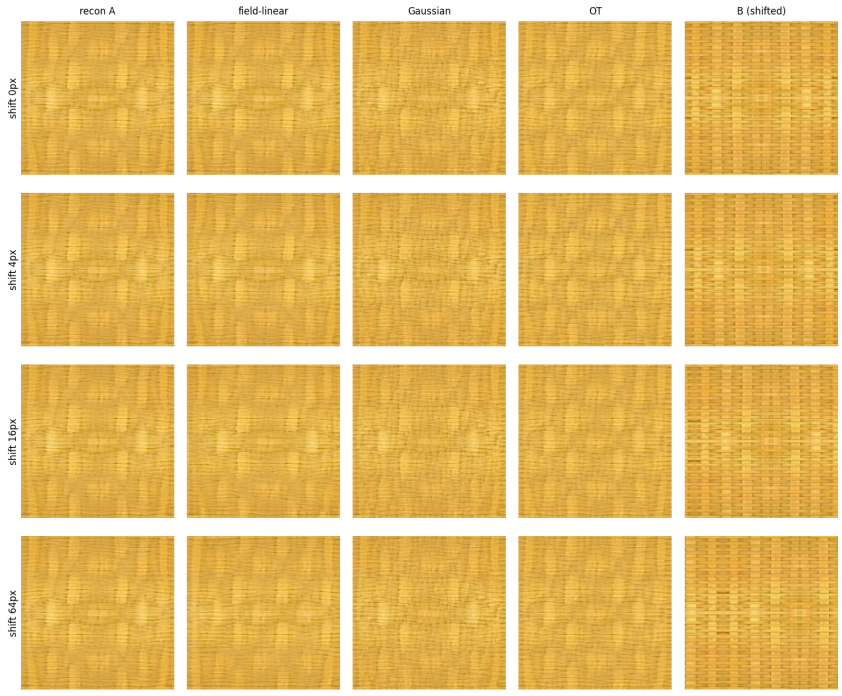

Figure 10: Alignment isolated. Rows: cyclic shift of B by 0, 4, 16, 64 px; columns: source reconstruction, field-linear, Gaussian, Soft OT (ours), B. Positional blending superposes as the shift grows; the appearance-aware rules are far less sensitive to displacement.

Source-conditioned material transfer. At $t = 1$, the operator uses target-aligned appearance with source conditioning. Reversing the arguments reverses those roles. The examples in the main-text transfer figure illustrate the resulting control over arrangement; the operation does not impose a pixelwise geometry constraint.

Spatially varying paths. The interpolation coefficient in a frozen map can vary across source cells: $z_i = (1 - t_i)x_i + t_i T(x_i)$. A scalar ramp produces a spatial transition; an externally chosen field gives spatial control. This extension changes the mixing coefficients without requiring another transport solve.

F Exploratory Mixtures, Coding, and Plan Diagnostics

F.1 Anchor-based mixtures

For k textures, choose source X_1 as an anchor and form

$$X_w = \sum_{j=1}^{k} w_j T_{1\to j}(X_1), \qquad w \in \Delta_k, \quad T_{1\to 1} = \mathrm{id}. \tag{13}$$

The decoder uses the anchor's conditioning. This one-step construction is anchor-dependent. On two four-texture sets, changing the anchor changes the output by 46% and 38% of mean source separation, versus same-anchor rerun floors of 14% and 32%. Along a four-texture path, round-trip errors are .12–.16 against a .14 reconstruction floor. These mixtures depend on an anchor and differ from exact multi-distribution Wasserstein barycenters. The four-texture strip is in the main text (Figure 4); its RT labels are the re-encoding round-trip errors quoted above. Figure 11 is a second four-texture example, Figure 12 browses the same construction as a 4×4 palette, and Figure 13 applies it to a natural image split into three regions, where the objects dissolve into a mixture rather than a recognizable scene; none is an exact barycenter.

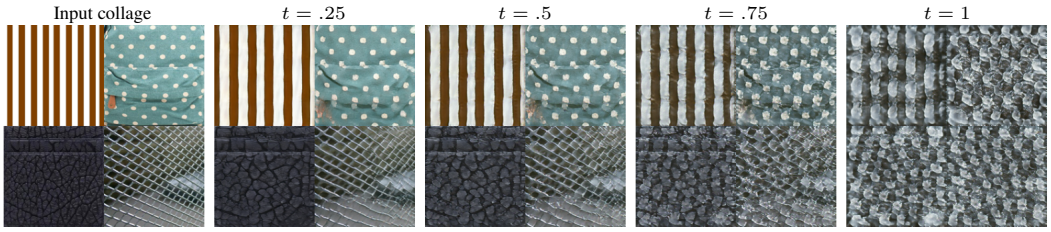

Figure 11: A second four-texture example: a 2×2 collage of four textures moved toward their anchor-based mixture as t increases.

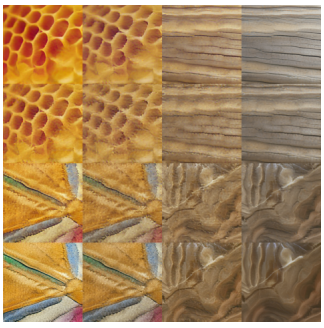

Figure 12: A 4×4 palette spanned by four corner textures under one anchor; every cell is a decoded mixture with the same conditioning.



Figure 13: Exploratory: a natural image clustered into three regions and moved toward their anchor-based mixture. Objects dissolve as distributions; the construction is not an exact barycenter and is outside the texture setting of the paper.

F.2 Sparse appearance coding

For target field Z and dictionary textures A_j, fixed transported atoms $T_{Z\to A_j}(Z)$ share the target's support. The weight problem is

$$\min_{w\in\Delta_K} \left\| \sum_j w_j T_{Z\to A_j}(Z) - Z \right\|_F^2 . \tag{14}$$

It is a convex quadratic problem for fixed atoms; uniqueness requires additional conditions. Sparse subset selection is a separate combinatorial step. Among 83 reconstructable targets and a 40-atom dictionary, the recorded sparse mixture improves on the best single atom in 82% of cases, with a median margin of 22%. Sixteen of 99 targets were excluded because CNT could not reconstruct them. Periodic structure is a recurring failure (Figure 14).

The dictionary study uses a pool of 376 textures. Raw farthest-point sampling can preferentially select corrupted encodings. Screening and a landmark-completed geometry improve that selection procedure; screened FPS with 40 atoms matches the decoded quality of 80 k-medoids atoms ($p = .42$). This result does not establish superiority of a selection rule. We did not retain the subset-search settings and raw coding outputs, so coding remains an exploratory extension.

F.3 Plan alignment

With centered clouds and RMS amplitudes a_A, a_B, define

$$\rho(P) = \frac{1}{a_A a_B} \sum_{ij} P_{ij} \langle x_i - \bar{x}, y_j - \bar{y} \rangle. \tag{15}$$

For nonzero amplitudes this lies in $[-1, 1]$ by Cauchy–Schwarz. A sharp 1,024-point assignment on 1,000 screened pairs yields 929 pairs at 0.977 ± 0.021 and 71 at 0.52 ± 0.08. The threshold was selected on the same data. Sharp entropic and hard-plan scores have correlation .99; the default diffuse plan is less informative. This describes pair geometry in-sample; prospective failure detection has not been validated.

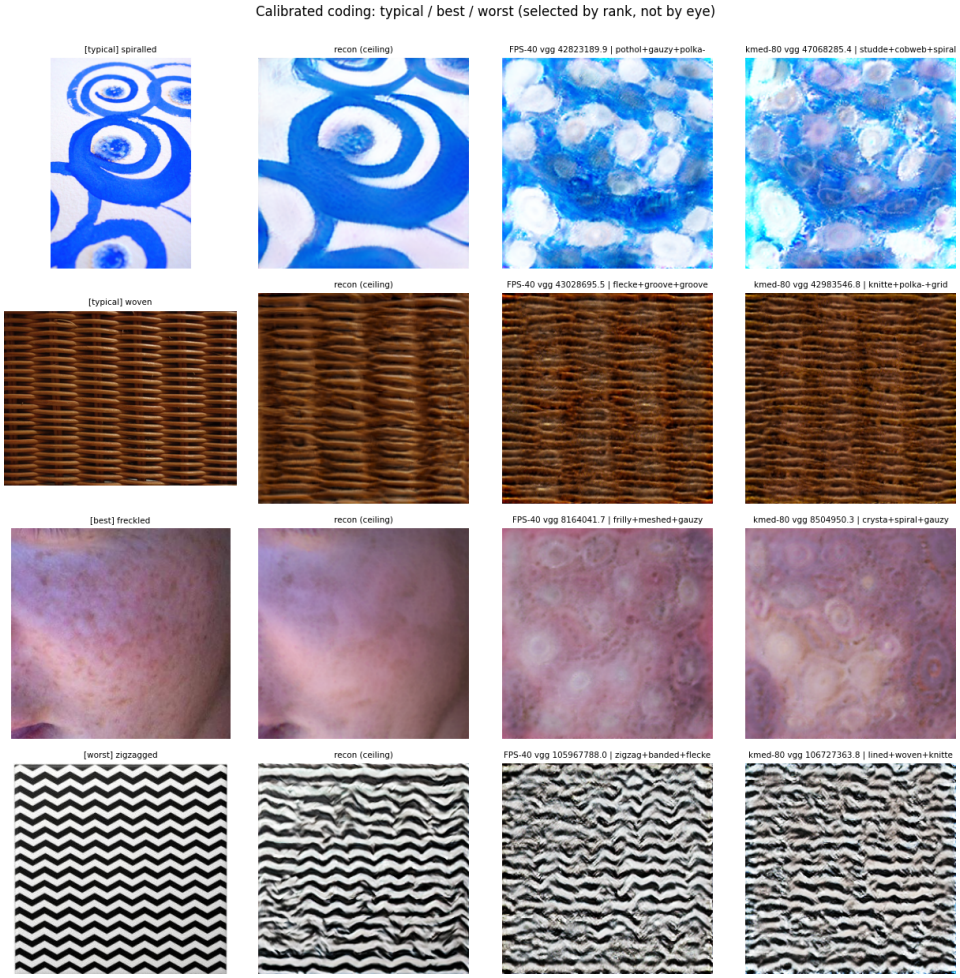

Figure 14: Exploratory dictionary reconstructions. Columns show the target, CNT reconstruction, and decoded constructions from two dictionary choices. The bottom row illustrates the difficulty of preserving regular structure.

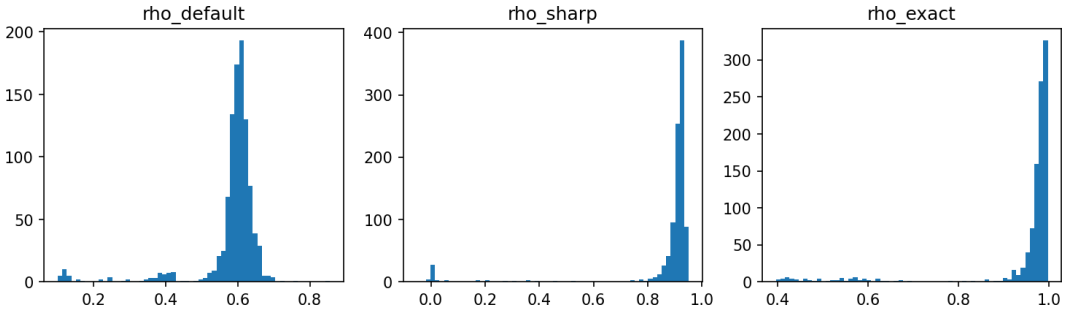

Figure 15: Plan-alignment histograms. The separated low-alignment group appears under sharp and hard plans. The ρ^* notation in the histogram refers to this statistic; only exact unregularized optimization maximizes alignment over the sampled coupling set.