How Do DiTs Denoise? An Aggregator–Predictor View of Attention and MLPs

Anonymous authors
Paper under double-blind review

Abstract

How do diffusion transformers (DiTs) represent image content and learn rules for reusing it to generalize? We develop an aggregator–predictor view of DiTs as *input-adaptive local denoisers*, in which context aggregation and patch prediction provide functional roles for interpreting attention and MLPs. We first establish that, for Gaussian Markov random field (GMRF) priors, the global Bayes denoiser can be approximated using only a subset of context patches, with error decaying exponentially in context radius. We further show that patchwise linear denoisers, often invoked to explain diffusion generalization, recover the empirical local denoiser with sufficiently informative aggregation. Within this view, AdaLN-modulated attention naturally parameterizes an input- and noise-dependent aggregator, while the MLP admits a learned key–value representation for tokenwise patch prediction. Our experiments support this view. DiTs trained on real images exhibit noise-dependent denoising contexts resembling those in the analytic GMRF setting, while attention mass exhibits a qualitatively corresponding broadening pattern. In a controlled CelebA64 study, shared-memory patch predictors used in place of DiT MLPs achieve competitive denoising and generative performance. Finally, cross-dataset component transplantation reveals compatibility patterns consistent with the proposed roles of QK in spatial weighting, VO in feature transport, and MLPs in patch prediction. Together, these results provide a component-level account of DiT denoising and motivate architecture-specific fine-tuning and controlled intervention.

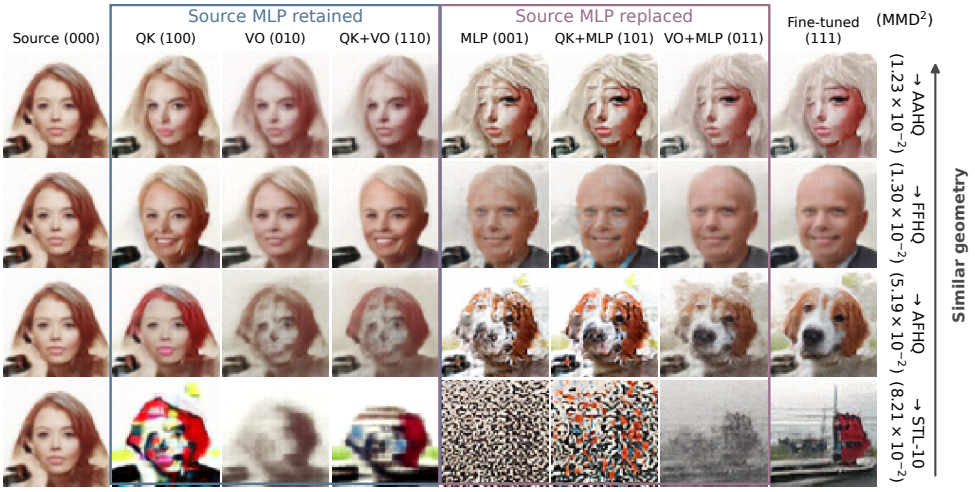

Figure 1: In each row, a CelebA source checkpoint is fine-tuned on the AAHQ, FFHQ, AFHQ, or STL-10 dataset. The source and fine-tuned generated images are shown in columns (000) and (111). In the middle columns, attention QK/VO and MLP weights of the source checkpoint are replaced with the corresponding fine-tuned weights; e.g., (001) means only MLP is replaced, while QK/VO are retained. **For face-to-face fine-tuning, replacing only QK generates source-like faces, while replacing MLP brings visual changes: cartoon faces, bald heads, or dogs.** Among the tested CelebA→STL-10 hybrids, coherent target-like structure appears primarily in the joint endpoint (111); reverting VO (101) leaves fragmented texture. See Sec. 5.3 and Appendix E.4.

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.

1 INTRODUCTION

Diffusion and flow-matching models are widely used for image generation (Song et al., 2021; Lipman et al., 2023), with Diffusion Transformers (DiTs) becoming a standard backbone (Peebles & Xie, 2023; Esser et al., 2024; Xie et al., 2025; Li & He, 2026). Growing evidence connects their generalization to *representations of images* and *rules for reusing them*. Diffusion features encode semantic and geometric information (Baranchuk et al., 2022; Tang et al., 2023), and aligning them with pretrained representations improves training (Yu et al., 2025). Models also generate combinations absent from the training data, as shown by analyses of synthetic hierarchical data (Okawa et al., 2023; Favero et al., 2025). How components of DiTs realize these capabilities remains unclear.

Existing explanations connect diffusion-model generalization to geometry-adaptive harmonic or balanced representations (Kadkhodaie et al., 2024; Zhang et al., 2026), and inductive biases toward spatial locality and translation equivariance (Kamb & Ganguli, 2025; Niedoba et al., 2025). These accounts leave open how DiT components divide the denoising computation. Rosu et al. (2025) show that attention can implement exact score-based denoising, but they consider whole-image tokens, leaving interactions among spatial patches unresolved. An et al. (2025) associate DiT generalization with attention mimicking CNN-like spatial locality, but attention can also select input-dependent, spatially non-contiguous context (Sec. 5.1). We ask: *How do attention, MLPs, and adaptive layer normalization (AdaLN) divide the work of representing image content and learning context-dependent rules for generalization?*

Understanding these component roles matters for intervention: domain adaptation, concept editing, and controlled generation require deciding which computations to modify and which to preserve. In language models, interpreting MLPs as key–value memories has motivated methods for editing stored associations (Geva et al., 2021; Meng et al., 2022; 2023), with related analyses in multimodal transformers (Pan et al., 2024a; Basu et al., 2024a). A functional account of DiTs could similarly guide component-wise adaptation across domains and interventions that change image content while preserving desired spatial relationships. Related work is reviewed in Appendix A.

We address this question by proposing an aggregator–predictor view of DiTs as *input-adaptive local denoisers*, supported by a statistical foundation and a set of architectural and empirical observations:

(1) A statistical foundation for local denoising. We justify local denoiser, in which each patch is denoised using only a subset of context patches, by identifying its statistical target and establishing when it approximates the global Bayes denoiser. Specifically, for Gaussian Markov random field (GMRF) image priors, the global denoiser can be approximated by a radius-r local denoiser with error decaying exponentially in r (Thm. 1). We further connect existing patchwise retrieval-based explanations of generalization to this target by showing that, with sufficiently informative aggregation, a patchwise linear denoiser exactly recovers the empirical local denoiser (Thm. 3).

(2) An aggregator–predictor interpretation of DiT components. We decompose the local denoising into *context aggregation* followed by *patch prediction* (Def. 2). Under this view, the algebraic structure of AdaLN-modulated attention implements an input- and noise-dependent aggregation rule: QK parameterizes spatial compatibility weights, while VO transforms the corresponding features (Sec. 4.1). In turn, MLP admits a key–value memory interpretation and operates on the resulting contextualized token (Sec. 4.2). These algebraic correspondences are exact for the isolated branches considered in Sec. 4; in a multilayer residual stack, the proposed roles interact and need not remain functionally separable (Appendix D.5). This view extends existing fixed spatial-locality accounts by allowing the relevant context to depend on the image and noise level and to include spatially non-contiguous patches. It further suggests the possibility of replacing DiT MLPs with more interpretable memory-based components, as supported by our experiments (Sec. 5.2).

(3) Empirical evidence consistent with the proposed component roles. Across real-image datasets, DiT sensitivity fields exhibit noise-dependent context broadening resembling the GMRF setting, while attention mass shows a related qualitative trend (Sec. 5.1). We further construct explicit shared-memory patch predictors that retain competitive denoising and generation performance in a controlled CelebA64 study (Sec. 5.2). Finally, cross-domain component transplantation reveals systematic component-compatibility patterns associated with source–target spatial similarity. These patterns are consistent with, but do not uniquely identify, the proposed roles in context aggregation and patch prediction (Sec. 5.3).

2 Background and Problem Setup

Let $X_0 = (X_{0,1}, \dots, X_{0,M}) \in \mathbb{R}^{Md_p}$ be a clean image with M patches, where $X_{0,m} \in \mathbb{R}^{d_p}$ is patch m. At time t, consider $X_t = a_t X_0 + b_t \varepsilon$, with $a_t, b_t > 0$ and $\varepsilon \sim \mathcal{N}(0, I_{Md_p})$ independent of X_0. This formulation includes variance-exploding (VE) and variance-preserving (VP) diffusion (Song et al., 2021), as well as linear Gaussian interpolation paths used in flow matching (Lipman et al., 2023). We define the normalized observation $Y_t := X_t/a_t = X_0 + \sigma_t \varepsilon$ and the effective noise level $\sigma_t := b_t/a_t$. Under standard squared-loss objectives, population-optimal clean-image, noise, and velocity predictions are related by known affine transformations to the denoiser (Li & He, 2026)

$$D^*_{m,t}(y) := \mathbb{E}[X_{0,m} \mid Y_t = y, t], \qquad m \in \{1, \dots, M\}. \tag{1}$$

Throughout the paper, unless otherwise specified, expectations involving Y_t are over $X_0 \sim p_0$ and the independent Gaussian corruption. We call (1) a population *global denoiser* because the prediction conditions on all noisy patches.

For training images $\{x^{(i)}\}_{i=1}^n$ and the empirical prior $\hat{p}_0 = n^{-1} \sum_{i=1}^n \delta_{x^{(i)}}$, the empirical version of the global denoiser (1) is a weighted sum of whole images:

$$\hat{D}_{m,t}(y) = \sum_{i=1}^n \mathrm{sm}_i \left\{ -\frac{1}{2\sigma_t^2} \left\| y - x^{(j)} \right\|^2 \right\}_{j \in [n]} x_m^{(i)}, \tag{2}$$

where the softmax sm is taken over the n training images. This creates a tension between training-loss minimization and generalization in diffusion models: under exact reverse dynamics and the corresponding terminal initialization, the empirical denoiser reproduces training images at the clean endpoint (Biroli et al., 2024; Kamb & Ganguli, 2025). Even when these conditions are not met, the shared mixture weights across patches keep the full-image prediction in the convex hull of whole training images, restricting its ability to recombine their local content independently (Kamb & Ganguli, 2025; Niedoba et al., 2025).

This raises the question of why trained models often generalize beyond memorizing training images. Several works attribute this to compositional generation, in which retrieval weights vary across patch locations (Niedoba et al., 2025; Kamb & Ganguli, 2025; Lukoianov et al., 2025). We formalize the estimator underlying the compositional accounts as a *patchwise linear denoiser (PLD)*. PLD depends on y only through the context patches $y_{\mathcal{B}_{m,t}}$ and allows different output patches to combine content from different training images. It is linear in how the context patches are aggregated.

Definition 1 (Patchwise linear denoiser (PLD)). *Fix a context set $\mathcal{B}_{m,t} \subseteq \{1, \dots, M\}$ and a matrix $S_{m,t} \in \mathbb{R}^{d_h \times Md_p}$ with full row rank whose lth block of columns, $[S_{m,t}]_l \in \mathbb{R}^{d_h \times d_p}$, is zero for $l \notin \mathcal{B}_{m,t}$. Given a symmetric positive definite $\Gamma_{m,t} \in \mathbb{R}^{d_h \times d_h}$, define the patchwise linear denoiser (PLD) by*

$$\hat{D}^{\mathrm{pld}}_{m,t}(y) = \sum_{i=1}^n \mathrm{sm}_i \left\{ -\frac{1}{2} \left\| S_{m,t}(y - x^{(j)}) \right\|^2_{\Gamma_{m,t}} \right\}_{j \in [n]} x_m^{(i)}, \tag{3}$$

where $\|u\|^2_{\Gamma_{m,t}} := u^\top \Gamma_{m,t} u$ and $x_m^{(i)}$ is patch m of training image i.

However, it is not clear when PLD achieves low population denoising risk or how a neural architecture can implement it. In the following sections, we address three central questions: what statistical target does the PLD approximate, when is that target close to the population denoiser, and how can a DiT implement such an estimator?

3 Local denoiser can approximate global denoiser

We first show when global denoising can be accurately approximated using only a local subset of patches. Suppose that the clean patches jointly follow a Markov random field (MRF) with respect to a graph $G_0 = ([M], E_0)$, and let $B_r(m) \subseteq [M]$ be a *graph-radius context set*, containing all patches within graph distance r of patch m. This subset need not be spatially contiguous (Fig. 2(a)). Define

$$D^*_{m,t}(y) := \mathbb{E}[X_{0,m} \mid Y_t = y, t]\,, \qquad D^{*,r}_{m,t}(y) := \mathbb{E}\big[X_{0,m} \mid Y_{t,B_r(m)} = y_{B_r(m)}, t\big]$$

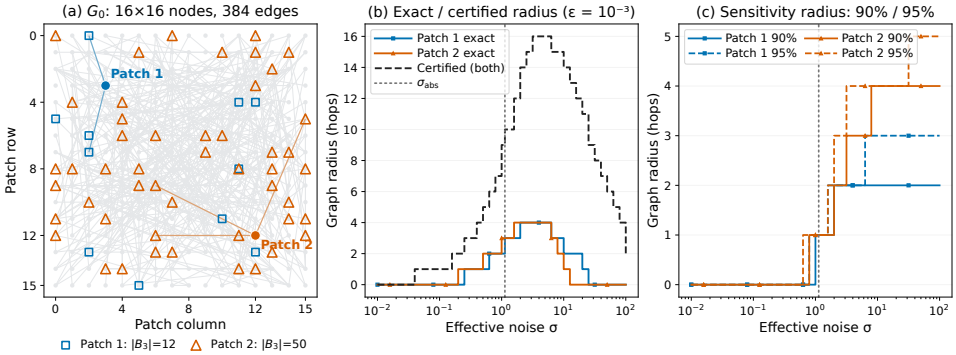

Figure 2: **Exact context set in a Gaussian MRF prior.** The known graph and precision matrix allow exact analysis of context set in this synthetic setting. (a) Radius-3 contexts $B_3(m)$ for two selected patches. Defined by graph distance, these sets need not be spatially contiguous. (b) Exact and certified radii at $\epsilon = 10^{-3}$. Corollary 2 guarantees that the certified radius is nondecreasing up to σ_{abs} (dotted lines); the exact radii can be substantially smaller. (c) 90%/95% sensitivity radii (Sec. 5.1), showing that relative influence spreads over a broader context as noise increases. Details in Appendix E.1.

as the *global denoiser* and $B_r(m)$*-local denoiser*, respectively. We define their mean-squared discrepancy $\mathbb{E}\left\|D^*_{m,t}(Y_t) - D^{*,r}_{m,t}(Y_t)\right\|_2^2$ as the *locality error* and say that p_0 is *exponentially local* with respect to G_0 if the following holds.

Assumption 1 (Exponential locality)**.** *For every t, there exists a constant $c_t > 0$ such that, for every patch m and integer $r \geq 0$, $\mathbb{E}\left\|D^*_{m,t}(Y_t) - D^{*,r}_{m,t}(Y_t)\right\|_2^2 \lesssim_t \exp(-c_t r)$.*

An explicit example is a Gaussian MRF image prior $X_0 \sim \mathcal{N}(0, Q^{-1})$ with positive definite block precision $Q \in \mathbb{R}^{Md_p \times Md_p}$ supported on G_0. Here, the global denoiser has the closed form $D^*_{m,t}(y) = [(I + \sigma_t^2 Q)^{-1} y]_m$. Although this denoiser generally depends on all noisy patches, analyticity of the inverse function and the graph sparsity of Q allow approximation using only radius-r context, with error decaying exponentially in r (Benzi & Razouk, 2007).

Theorem 1 (Exponential locality of GMRF, informal)**.** *A Gaussian MRF is exponentially local in the sense of Assump. 1. The full bound and conditions are in Appendix C.1.*

Intuitively, noisier patches provide weaker evidence individually, making broader context useful for denoising. The following guarantee formalizes this intuition for the bound; the optimal local denoiser may achieve the same tolerance with less context (Fig. 2(b)).

Corollary 2 (Bounded-range monotonicity of the locality radius, informal)**.** *For any error tolerance $\epsilon > 0$, the smallest radius $r_\epsilon(\sigma_t)$ certified by the explicit exponential bound is nondecreasing in noise on $0 \leq \sigma_t \leq \sigma_{\text{abs}}$, where $\sigma_{\text{abs}} > 0$ depends on Q. The full statement, radius formula, and noise threshold are in Appendix C.2.*

This guarantee concerns a bounded noise range: at extreme noise, the posterior mean approaches the zero prior mean and the absolute bound vanishes. Fig. 2 illustrates the context $B_r(m)$ defined by graph distance and the radius guarantee in Cor. 2 using a synthetic GMRF prior. Similar locality patterns appear in DiTs trained on real images (Sec. 5.1).

3.1 Aggregator–Predictor Decomposition of Local Denoiser

We next interpret PLD and DiTs as estimators of the same local denoiser. Without assuming that the image prior is an MRF, fix a patch m, a time t, and a *context set* $\mathcal{B}_{m,t}$. We define the following population target as the $\mathcal{B}_{m,t}$*-local denoiser*

$$D^{*,\mathcal{B}}_{m,t}(y) := \mathbb{E}[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}} = y_{\mathcal{B}_{m,t}}, t]. \tag{4}$$

We decompose its estimation in two stages: *local aggregation of the relevant context*, followed by *patch prediction from the aggregated statistic*.

Specifically, a measurable map $\mathcal{S}_{m,t} : \mathbb{R}^{Md_p} \to \mathbb{R}^{d_h}$ is a $\mathcal{B}_{m,t}$-*local aggregator* if $y_{\mathcal{B}_{m,t}} = y'_{\mathcal{B}_{m,t}}$ implies $\mathcal{S}_{m,t}(y) = \mathcal{S}_{m,t}(y')$ for all y, y'. It depends only on the context patches and maps them, possibly nonlinearly, to a statistic in $\mathbb{R}^{d_h}$. A measurable map $g_{m,t} : \mathbb{R}^{d_h} \to \mathbb{R}^{d_p}$, called a *patch predictor*, then predicts the clean patch from this statistic. For a given local aggregator, a predictor recovering the local conditional mean exists if and only if $\mathrm{sf}(D^{*,\mathcal{B}}_{m,t}(Y_t)) \subseteq \mathrm{sf}(\mathcal{S}_{m,t}(Y_t))$; that is, local aggregation preserves all information needed to recover the local conditional mean. The aggregator may discard details that are unnecessary for determining this mean. In that case, the tower property shows that such predictor can be chosen as $g^*_{m,t}(s) = \mathbb{E}[X_{0,m} \mid \mathcal{S}_{m,t}(Y_t) = s, t]$.

Definition 2 (Aggregator–predictor decomposition)**.** *A pair of measurable maps $(\mathcal{S}^*_{m,t}, g^*_{m,t})$ is an* aggregator–predictor decomposition *of $D^{*,\mathcal{B}}_{m,t}$ if $\mathcal{S}^*_{m,t}$ is $\mathcal{B}_{m,t}$-local and, for P_{Y_t}-almost every y, $D^{*,\mathcal{B}}_{m,t}(y) = (g^*_{m,t} \circ \mathcal{S}^*_{m,t})(y)$.*

This decomposition is generally nonunique but provides a framework for separately estimating the aggregator and predictor of the local denoiser (4). The PLD is one instance; for a patch m and time t with $\sigma_t > 0$, setting $\Gamma_{m,t} = (\sigma_t^2 S_{m,t} S_{m,t}^\top)^{-1}$ in Def. 1, the PLD is decomposed into $\mathcal{S}^{\mathrm{lin}}_{m,t}$, the *linear aggregator*, and $g^{\mathrm{rbf}}_{m,t}$, the *radial basis function (RBF) predictor*:

$$\hat{D}^{\mathrm{pld}}_{m,t}(y) = \left(g^{\mathrm{rbf}}_{m,t} \circ \mathcal{S}^{\mathrm{lin}}_{m,t}\right)(y),$$
$$\mathcal{S}^{\mathrm{lin}}_{m,t}(y) = S_{m,t} y, \qquad g^{\mathrm{rbf}}_{m,t}(s) = \sum_{i=1}^{n} \mathrm{sm}_i \left\{ -\frac{1}{2} \left\| s - \mathcal{S}^{\mathrm{lin}}_{m,t}(x^{(j)}) \right\|^2_{(\sigma_t^2 S_{m,t} S_{m,t}^\top)^{-1}} \right\}_{j \in [n]} x^{(i)}_m. \tag{5}$$

Bayes' rule identifies the predictor $g^{\mathrm{rbf}}_{m,t}(s)$ with $\mathbb{E}_{\hat{p}_0}[X_{0,m} \mid \mathcal{S}^{\mathrm{lin}}_{m,t}(Y_t) = s, t]$, the conditional mean given the aggregated statistic s under the empirical prior; see Appendix C.3. This correspondence has been noted in prior work (Niedoba et al., 2025; Kamb & Ganguli, 2025; Lukoianov et al., 2025). In our framework, these approaches correspond to using cropping matrices, prescribed neighborhood masks, or thresholded Wiener-filter sensitivity masks for $S_{m,t}$, respectively. We go a step further and show that, with a sufficiently expressive aggregator, $\hat{D}^{\mathrm{pld}}_{m,t}$ exactly recovers the local denoiser under the empirical observation law.

Theorem 3 (PLD recovers the empirical local denoiser)**.** *Under the setup above, let $\sigma_t > 0$ and take $\Gamma_{m,t} = (\sigma_t^2 S_{m,t} S_{m,t}^\top)^{-1}$ in Def. 1. If $\mathcal{S}^{\mathrm{lin}}_{m,t}$ preserves the empirical local conditional mean, i.e., $\mathrm{sf}(\mathbb{E}_{\hat{p}_0}[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}}, t]) \subseteq \mathrm{sf}(\mathcal{S}^{\mathrm{lin}}_{m,t}(Y_t))$, then, almost surely under the empirical observation law, $\hat{D}^{\mathrm{pld}}_{m,t}(y) = \mathbb{E}_{\hat{p}_0}[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}} = y_{\mathcal{B}_{m,t}}, t]$.*

The condition holds whenever the full local observation can be recovered from the aggregate, as with a patch-selection matrix. A compressive linear aggregator can also suffice: when the training contexts restricted to $\mathcal{B}_{m,t}$ lie in a lower-dimensional affine subspace, a fixed linear aggregator whose row space contains their difference subspace is sufficient; see (17).

Theorem 3 therefore connects the PLD, used to explain generalization by composition, to the population local denoiser (4): with sufficient aggregation, retrieval recovers its empirical counterpart exactly. However, requiring all training-context differences to lie in a single low-dimensional subspace is restrictive. This motivates the nonlinear DiT aggregator considered next.

4 DiTs as Adaptive Local Denoisers

We next relate DiT attention and MLPs to the aggregator and predictor roles of the local-denoising framework. We consider a single layer without residual connections $\hat{D}^{\mathrm{dit}}_{m,t}(y) = (g^{\mathrm{mlp}}_t \circ \mathcal{S}^{\mathrm{attn}}_{m,t})(y)$, where the attention aggregator $\mathcal{S}^{\mathrm{attn}}_{m,t}$ and MLP predictor g^{mlp}_t are defined in the following sections. Appendix D gives the derivation and residual-stack extension; Fig. 6 summarizes the component mapping. Note that in a residual stack, VO and MLP updates can alter later QK routing, and attention can affect later MLP updates, so the branch-level algebra does not necessarily imply a functionally separable decomposition of the full network.

4.1 ATTENTION AS AN INPUT-ADAPTIVE AGGREGATOR

PLD's linear aggregator (5) uses a fixed context rule across inputs. A more flexible denoiser can adapt its context to the input, drawing on different patches as their relevance changes with the image and noise level. We express this through an *input-adaptive aggregator*

$$\mathcal{S}_{m,t}(Y_t) = \sum_{l=1}^{M} S_{ml}(Y_t; t)\Phi(Y_{t,l}; t, l). \tag{6}$$

Here $\Phi(Y_{t,l}; t, l) \in \mathbb{R}^{d_h}$ represents patch l together with its position and noise level, while $S_{ml}(Y_t; t) \in \mathbb{R}^{d_h \times d_h}$ determines how it contributes to the context for denoising patch m.

DiT attention, incorporating patch, position, and time embeddings together with AdaLN conditioning, provides a factorized parameterization of such an input-adaptive aggregator. Let $E \in \mathbb{R}^{d_h \times d_p}$ be the patch projection, $e_l \in \mathbb{R}^{d_h}$ the positional embedding, and $\phi_{t,l}(Y_t) := \Phi^{\mathrm{attn}}(EY_{t,l}; e_l, t)$ the feature formed by patch embedding, position embedding, and AdaLN. For each of $\tilde{H}$ equal-width heads, $\tilde{W}_h^Q$, $\tilde{W}_h^K$, and $\tilde{W}_h^V$ are the query (Q), key (K), and value (V) projections, and $\tilde{W}_h^O$ is the output (O) projection. Writing the attention AdaLN residual gate as $\tilde{\alpha}_t \in \mathbb{R}^{d_h}$, the attention branch constitutes an *attention aggregator* defined as

$$\begin{aligned} \mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t) &= \sum_{l=1}^{M} S^{\mathrm{attn}}_{ml}(Y_t; t)\phi_{t,l}(Y_t), \\ S^{\mathrm{attn}}_{ml}(Y_t; t) &= \sum_{h=1}^{\tilde{H}} \mathrm{sm}_l \left\{ \frac{\langle \tilde{W}_h^Q \phi_{t,m}(Y_t), \tilde{W}_h^K \phi_{t,j}(Y_t) \rangle}{(d_h/\tilde{H})^{1/2}} \right\}_{j \in [M]} \mathrm{diag}(\tilde{\alpha}_t) \tilde{W}_h^O \tilde{W}_h^V . \end{aligned} \tag{7}$$

A single attention branch separates QK spatial weighting from VO value transformation and mixing (7). The QK projections determine the compatibility scores of patches through $(\tilde{W}_h^Q)^\top \tilde{W}_h^K$, and softmax converts these scores into spatial weights. By smoothly suppressing patches with low relative scores, attention performs soft thresholding and forms an approximately local aggregator when its weights concentrate on a suitable context. In this branch-level sense, QK parameterizes which patch positions receive greater or smaller weight in the constructed context, depending on image content, position, and noise level. Appendix D.3 details the local approximation.

VO transforms and combines the selected features through the shared, time-dependent channel operators $\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}_h^O\tilde{W}_h^V$. Attention thus combines $\tilde{H}$ shared channel operators with input-dependent spatial weights, rather than learning an independent operator for every patch pair. This decomposition motivates an aggregator interpretation: for fixed input features, QK directly controls relative spatial weights, while VO applies patch-shared channel operators to the weighted features. The downstream impact of an attention weight also depends on the value features, channel operators, and residual interactions. These roles interact through learned features and across layers, but the distinction motivates examining QK and VO separately in Sec. 5.3.

4.2 MLP AND SHARED-MEMORY PATCH PREDICTORS

Within the single-layer model, the DiT MLP g_t^{mlp} acts on the attention-aggregated context $s = \mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t)$ in the patch-predictor role; in a residual stack, its output is a tokenwise feature update. The framework also allows alternative patch predictors, such as the RBF predictor $g^{\mathrm{rbf}}_{m,t}$ in (5), which explicitly retrieves from training patches. In a multilayer residual stack (Appendix D.5), these predictors can share memory across layers, just as $g^{\mathrm{rbf}}_{m,t}$ would consult the same training patches at each layer. This motivates a sequence of *shared-memory patch predictors* connecting explicit patch retrieval to an ordinary MLP. We present simplified forms below; Appendix D.4 gives the full parameterizations, including biases, output gates, compatibility projections, and multiple heads.

We first note that g_t^{mlp} and $g^{\mathrm{rbf}}_{m,t}$ admit similar key–value memory interpretations (Geva et al., 2021). If coupled with the attention aggregator, $g^{\mathrm{rbf}}_{m,t}$ retrieves from patch-specific values $\{x_m^{(i)}\}_{1:n}$ using contextualized training images $\mathcal{S}^{\mathrm{attn}}_{m,t}(x^{(i)})$ as keys. A transformer MLP of hidden width R likewise consists of learnable key–value pairs $\{k^{(r)}, v^{(r)} \in \mathbb{R}^{d_h}\}_{r=1}^{R}$. In DiTs, AdaLN modulates the

memory query to produce time-dependent retrieval weights:

$$g_t^{\mathrm{mlp}}(s) = \sum_{r=1}^{R} \mathrm{gelu}\Big(\mathrm{AdaLN}_t(s)^\top k^{(r)}\Big)\, v^{(r)}. \tag{8}$$

Compared with $g_{m,t}^{\mathrm{rbf}}$ in (5), there are three differences: (i) the memory is learned rather than drawn from training patches, (ii) the same bank of memory is used across patches, and (iii) the retrieval weights are unnormalized and may have either sign. These differences motivate three patch-predictor variants that start from $g_{m,t}^{\mathrm{rbf}}$ and progressively resemble g_t^{mlp}: we first make the memory learnable, then share it across patches, and finally replace normalized retrieval weights with unnormalized, potentially signed coefficients.

All predictors act on the same contextualized patch $s = \mathcal{S}_{m,t}^{\mathrm{attn}}(Y_t)$. Let W^Q, W^K, W^V be layerwise projections, with softmax taken over the R memory entries.

$$g_{m,t}^{\mathrm{lpld}}(s) = \sum_{r=1}^{R} \mathrm{sm}_r \left\{\mathrm{AdaLN}_t(s)^\top (W^Q)^\top W^K \mathcal{S}_{m,t}^{\mathrm{attn}}(z^{(j)})\right\}_{j\in[R]} W^V z_m^{(r)}, \tag{9}$$

$$g_t^{\mathrm{sca}}(s) = \sum_{r=1}^{R} \mathrm{sm}_r \left\{\mathrm{AdaLN}_t(s)^\top (W^Q)^\top W^K k^{(j)}\right\}_{j\in[R]} W^V v^{(r)}, \tag{10}$$

$$g_t^{\mathrm{smlp}}(s) = \sum_{r=1}^{R} \mathrm{gelu}\Big(\mathrm{AdaLN}_t(s)^\top (W^Q)^\top W^K k^{(r)}\Big)\, W^V v^{(r)}. \tag{11}$$

Compared to PLD, the first variant (9) replaces the training images with a learned memory bank $\{z^{(r)} \in \mathbb{R}^{Md'_h}\}_{r=1}^R$ shared across layers, and RBF retrieval with softmax. Just as a training image $x^{(i)}$ consists of M patches, each memory entry $z^{(r)}$ consists of M learnable patches $z_m^{(r)} \in \mathbb{R}^{d'_h}$. The memory width d'_h may equal the context width d_h or differ from it; a learned compatibility projection shared across layers accommodates differing widths. Together, these choices define the **learned PLD (LPLD)** predictor $g_{m,t}^{\mathrm{lpld}}$. LPLD uses the attention-aggregated memory $z^{(r)}$ to form keys, whereas the value is taken directly from its mth patch.

Storing a separate set of memory patches at each patch position can be redundant, especially given that patchification is an architectural choice. In (10) and (11), the patch-specific memory bank in $g_{m,t}^{\mathrm{lpld}}$ is replaced with an untied key–value bank $\{k^{(r)}, v^{(r)} \in \mathbb{R}^{d'_h}\}_{r=1}^R$ shared across patches and layers. Coupled with the softmax retrieval, this gives a **shared cross-attention (SCA)** predictor g_t^{sca}. Finally, we replace normalized softmax weights with unnormalized GELU coefficients for a more flexible retrieval rule. This gives a **shared MLP (SMLP)** predictor g_t^{smlp}. Absorbing the projection matrices into the keys and values allows SMLP to recover the MLP form (8). For all three variants, sharing the memory bank across layers can reduce the parameter count, allowing us to allocate more capacity to predictor heads at matched parameter budgets.

5 Experiments

Our experiments examine whether trained DiT denoisers exhibit locality patterns resembling those of the oracle GMRF denoiser (Sec. 5.1), and whether, as suggested by our aggregator–predictor framework, layerwise MLPs can be replaced with more interpretable, shared-memory patch predictors (Sec. 5.2). Finally, we use component transplantation to examine whether compatibility patterns among jointly adapted QK, VO, and MLP parameters are consistent with this framework (Sec. 5.3). Experimental details and additional results are in Appendix E.

5.1 Real-image denoising echoes GMRF locality

An exact analysis as in Fig. 2 is unavailable for real images. Instead, we identify context sets using *sensitivity weights*, which quantify how the prediction of patch m responds to perturbations of input patch l, based on a trained DiT's Jacobian, similar to Niedoba et al. (2025). The population denoiser's Jacobian measures posterior dependence between clean patches given the noisy patches

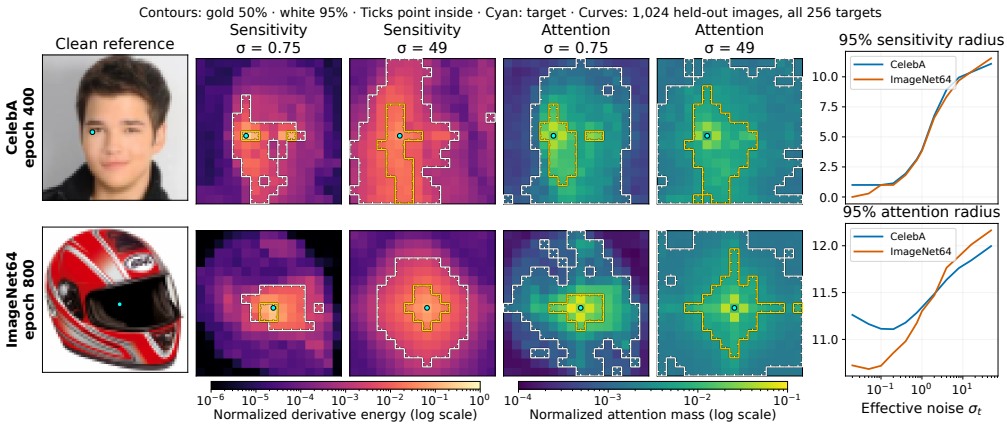

Figure 3: **Sensitivity-defined contexts broaden with noise, with attention mass showing a related trend.** Gold/white contours enclose 50%/95% of normalized squared sensitivity or attention mass; ticks point inward and cyan dots mark target patches. Right: 95% radii averaged over target patches; Details in Appendix E.2.

(Dytso et al., 2021). Based on these weights, we define the *q-sensitivity radius* as the smallest radius of a square centered on patch m containing at least a fraction q of the total weight. The *sensitivity context set* contains the fewest patches carrying at least that fraction.

Denoising sensitivity exhibits GMRF-like broadening; attention shows a related trend. For the synthetic GMRF where the true Jacobian is available, Fig. 2(c) shows that sensitivity spreads over broader contexts as noise increases. Denoising real images echoes this pattern: Fig. 3 shows concentrated sensitivity at low noise and broader, sometimes disconnected contexts at higher noise, together with increasing sensitivity radii. Notably, replacing sensitivity weights with attention mass yields analogous attention contexts and radii, which also generally broaden with noise. Together, these observations support the GMRF setting as a tractable reference model for denoising locality and the interpretation of attention as an input-adaptive context aggregator. Sensitivity and attention mass measure different quantities, so the comparison concerns a shared qualitative trend rather than identical context sets. Additional results appear in Appendix E.2.

5.2 Shared-memory predictors can replace MLPs in a controlled setting

We replace DiT MLPs with LPLD, SCA, and SMLP on CelebA64. We compare similar parameter budgets using a common backbone architecture, with modest models and one training seed per configuration for unconditional 64×64 face generation. Figure 4 reports held-out velocity loss, denoising error, and FID at epoch 400. Three patterns appear; first, the patch-shared SCA and SMLP banks outperform the position-specific LPLD bank at comparable parameter budgets, showing that competitive prediction need not assign separate memory for every patch position. Second, SMLP improves on SCA, supporting unnormalized GELU retrieval as an effective alternative to softmax. Third, SMLP with $H = 8$ achieves competitive denoising and lower FID than the ordinary MLP, despite sharing its base memory across layers. Multiple predictor heads generally improve performance, suggesting that multiple projections of the shared bank can be useful. Together, these results demonstrate that an explicit, layer-shared memory bank can support competitive patch prediction

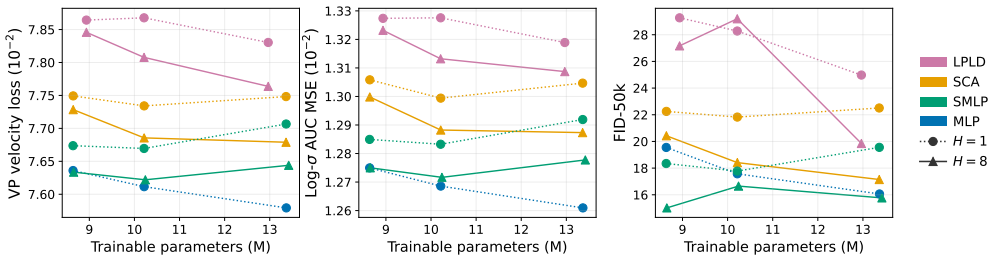

Figure 4: **Shared-memory patch predictors at comparable parameter budgets.** Models are trained on CelebA64 for 400 epochs; see Appendix E.3 for details.

in this controlled setting. We note that these results by themselves do not establish that layerwise MLP parameters generally encode a single shared memory or that this design transfers unchanged to larger conditional DiTs. Training details and limitations are in Appendix E.3.

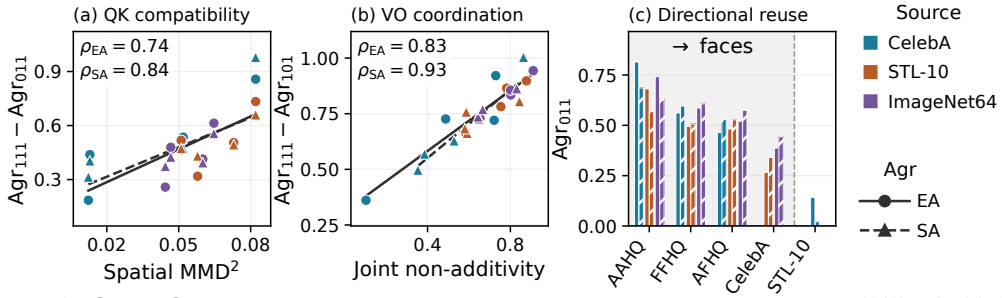

Figure 5: **QK, VO, and MLP component transplants.** (a) Source-QK compatibility is higher across spatially similar datasets. (b) VO reversion costs more as joint non-additivity increases. (c) Natural-image source QK retains partial agreement on face targets. See Appendix E.4.2.

5.3 Component transplants support the aggregator–predictor view

Transplanting jointly adapted components probes compatibility and coadaptation. To examine compatibility patterns in relation to the proposed component roles, we fine-tune a pretrained source model on a target dataset. We then transplant combinations of the adapted components into the source checkpoint. When the datasets have similar spatial structure, our framework suggests that source QK may remain compatible with adapted VO and MLP parameters. Larger spatial shifts may be associated with stronger coadaptation. We study twelve transfer directions between face and natural-image datasets. For clarity, we jointly fine-tune only QK, VO, and MLP, leaving AdaLN and all other parameters fixed, assuming that the learned time conditioning transfers across domains. Each of the eight resulting hybrids is labeled by QK/VO/MLP bits: 0 for source weights and 1 for jointly adapted weights. These transplants probe the compatibility and coadaptation of parameters learned jointly; they do not measure the performance obtainable by training QK-, VO-, or MLP-only models. Separately, under the same target-domain update budget, QK-only and QK+VO-only fine-tuning generally fall further behind joint adaptation on larger domain shifts (Appendix E.4.5).

Following work on patch self-similarity and spatial normalization (Tumanyan et al., 2022; Singh et al., 2026), we measure spatial distance between datasets using MMD^2 of patch self-similarity descriptors. We compare each hybrid's outputs with the fully adapted model's using matched initial noise. We measure this agreement using DINOv2 global features for endpoint agreement (EA) and patch self-similarity patterns for spatial agreement (SA). Both scores assign 0 to the source and 1 to the fully adapted model. Full protocols and results are in Appendix E.4.

QK compatibility is higher for spatially similar datasets, while VO reversion penalties track component coadaptation. With source QK retained (011), CelebA→AAHQ/FFHQ/AFHQ produces recognizable faces. In contrast, among the tested CelebA→STL-10 hybrids, coherent target-like structure emerges primarily only when all jointly adapted components are present (111; Figs. 1 and 13). Consistent with this pattern, the QK reversion penalty tends to increase with spatial distance (Fig. 5(a)), which also accompanies stronger component coadaptation (Appendix E.4.3). Additionally, restoring source VO (101) yields fragmented textures in the natural-image-to-face transfers (Figs. 14 and 15). Its reversion penalty increases with joint non-additivity—the gain from joint adaptation beyond the sum of isolated component changes (Fig. 5(b)).

Source QK from broader natural-image domains retains partial compatibility on face targets. With source QK retained (011), STL-10 hybrids recover human and animal face structure, including a blurred face for STL-10→CelebA, while joint adaptation produces clearer results (Fig. 14). This contrasts with the poorly formed scene in the reverse CelebA→STL-10 direction (Fig. 1); additional examples in Appendix E.4.4. Figure 5(c) supports this asymmetry: source QK from STL-10 retains greater endpoint agreement on CelebA than in the reverse direction, while QK from natural-image domains retains partial agreement on face targets at levels comparable to CelebA QK.

AI USE STATEMENT

Generative AI tools were used as research assistants for literature review, refining the presentation of the work, writing and cleaning up code, assisting with experiment execution, cleaning up proofs, providing feedback on the manuscript, and polishing the text. All AI-assisted results were independently reviewed and verified by the authors. The authors made the final scientific and methodological decisions and take full responsibility for the contents of the paper.

REPRODUCIBILITY STATEMENT

Assumptions and proofs are in Appendix B. Appendix E documents the datasets, model configurations, training and evaluation protocols, and metric definitions for all three experiments. We provide anonymous code and reproduction instructions for dataset preparation, training, evaluation, and regeneration of the figures and tables, together with the small numerical summaries used to render the reported plots.

REFERENCES

Jie An, De Wang, Pengsheng Guo, Jiebo Luo, and Alex Schwing. On inductive biases that enable generalization in diffusion transformers. In *Advances in Neural Information Processing Systems*, volume 38, pp. 31738–31757. Curran Associates, Inc., 2025. doi: 10.52202/085713-1066. URL https://papers.neurips.cc/paper_files/paper/2025/hash/2dc52e27a26afabb0bf2123bceaf9208-Abstract-Conference.html.

Harry C Andrews and Bobby R Hunt. *Digital image restoration*. Prentice-Hall, 1977.

Haim Avron and Sivan Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM*, 58(2):8:1–8:17, 2011. doi: 10.1145/1944345.1944349. URL https://www.cs.tau.ac.il/~stoledo/Bib/Pubs/randomized-trace.pdf.

Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *International Conference on Learning Representations*, 2022.

Samyadeep Basu, Martin Grayson, Cecily Morrison, Besmira Nushi, Soheil Feizi, and Daniela Massiceti. Understanding information storage and transfer in multi-modal large language models. *Advances in Neural Information Processing Systems*, 37, 2024a.

Samyadeep Basu, Keivan Rezaei, Priyatham Kattakinda, Vlad I. Morariu, Nanxuan Zhao, Ryan A. Rossi, Varun Manjunatha, and Soheil Feizi. On mechanistic knowledge localization in text-to-image generative models. In *Proceedings of the 41st International Conference on Machine Learning*, 2024b.

Samyadeep Basu, Nanxuan Zhao, Vlad I. Morariu, Soheil Feizi, and Varun Manjunatha. Localizing and editing knowledge in text-to-image generative models. In *International Conference on Learning Representations*, 2024c.

Michele Benzi and Nader Razouk. Decay bounds and $O(n)$ algorithms for approximating functions of sparse matrices. *Electronic Transactions on Numerical Analysis*, 28:16–39, 2007.

Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(2):192–236, 1974.

Giulio Biroli, Tony Bonnaire, Valentin de Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15:9957, 2024. doi: 10.1038/s41467-024-54281-3. URL https://www.nature.com/articles/s41467-024-54281-3.

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mézard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2505.17638.

Rama Chellappa and Shankar Chatterjee. Classification of textures using Gaussian Markov random fields. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(4):959–963, 1985. doi: 10.1109/TASSP.1985.1164641.

Xinlei Chen, Zhuang Liu, Saining Xie, and Kaiming He. Deconstructing denoising diffusion models for self-supervised learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8b52c29d1a77e3fcc6c8f90fa0ac0e8e-Abstract-Conference.html.

Yunjin Chen, René Ranftl, and Thomas Pock. Insights into analysis operator learning: From patch-based sparse models to higher order MRFs. *IEEE Transactions on Image Processing*, 23(3): 1060–1072, 2014. doi: 10.1109/TIP.2014.2299065.

Caroline Choi, Yoonho Lee, Annie Chen, Allan Zhou, Aditi Raghunathan, and Chelsea Finn. AutoFT: Learning an objective for robust fine-tuning, 2024. URL https://arxiv.org/abs/2401.10220.

Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. StarGAN v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. URL https://arxiv.org/abs/1912.01865.

Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of ImageNet as an alternative to the CIFAR datasets. *arXiv preprint arXiv:1707.08819*, 2017. URL https://arxiv.org/abs/1707.08819.

Adam Coates, Honglak Lee, and Andrew Ng. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223, 2011. URL https://proceedings.mlr.press/v15/coates11a.html.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, 2022.

DLMF. NIST Digital Library of Mathematical Functions. Release 1.2.8, 2026. URL https://dlmf.nist.gov/18.12.E8. Released September 15, 2026. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, editors.

Alex Dytso, H. Vincent Poor, and Shlomo Shamai (Shitz). A general derivative identity for the conditional mean estimator in Gaussian noise and some applications. *arXiv preprint arXiv:2104.01883*, 2021. URL https://arxiv.org/abs/2104.01883.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633. PMLR, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Alessandro Favero, Antonio Sclocchi, Francesco Cagnetta, Pascal Frossard, and Matthieu Wyart. How compositional generalization and creativity improve as diffusion models are trained. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 16286–16306. PMLR, 2025. URL https://proceedings.mlr.press/v267/favero25a.html.

Chaofan Gan, Yuanpeng Tu, Xi Chen, Tieyuan Chen, Yuxi Li, Mehrtash Harandi, and Weiyao Lin. Unleashing diffusion transformers for visual correspondence by modulating massive activations. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/a654298de44ec95dec28aa37e8d0af51-Abstract-Conference.html.

Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6 (6):721–741, 1984. doi: 10.1109/TPAMI.1984.4767596.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, 2021.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

Klaus Greff, Rupesh K. Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Skn9Shcxe.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012. URL https://www.jmlr.org/papers/v13/gretton12a.html.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=D3DBqvSDbj.

Andi Han, Wei Huang, Yuan Cao, and Difan Zou. On the feature learning in diffusion models. In *International Conference on Learning Representations*, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/7f7fa581cc8a1970a4332920cdf87395-Abstract-Conference.html.

Yujin Han, Andi Han, Wei Huang, Chaochao Lu, and Difan Zou. Can diffusion models learn hidden inter-feature rules behind images? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21704–21732. PMLR, 2025b. URL https://proceedings.mlr.press/v267/han25b.html.

Dominik Hintersdorf, Lukas Struppek, Kristian Kersting, Adam Dziedzic, and Franziska Boenisch. Finding NeMo: Localizing neurons responsible for memorization in diffusion models. In *Advances in Neural Information Processing Systems*, 2024.

Benjamin Hoover, Hendrik Strobelt, Dmitry Krotov, Judy Hoffman, Zsolt Kira, and Duen Horng Chau. Memory in plain sight: Surveying the uncanny resemblances of associative memories and diffusion models, 2023. URL https://arxiv.org/abs/2309.16750.

Stanislaw Jastrzebski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual connections encourage iterative inference. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SJa9iHgAZ.

Zahra Kadkhodaie, Florentin Guth, Eero P. Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ANvmVS2Yr0.

Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 28795–28831. PMLR, 2025. URL https://proceedings.mlr.press/v267/kamb25a.html.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. URL https://arxiv.org/abs/1812.04948.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2021. URL https://nvlabs.github.io/stylegan3/.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Kumari_Multi-Concept_Customization_of_Text-to-Image_Diffusion_CVPR_2023_paper.html.

Stan Z Li. *Markov random field modeling in image analysis*. Springer, 3rd edition, 2009.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36115–36125, June 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Li_Back_to_Basics_Let_Denoising_Generative_Models_Denoise_CVPR_2026_paper.html.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Mingcong Liu, Qiang Li, Zekui Qin, Guoxin Zhang, Pengfei Wan, and Wen Zheng. BlendGAN: Implicitly GAN blending for arbitrary stylized face generation. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2110.11728.

Zhiheng Liu, Ruili Feng, Kai Zhu, Yifei Zhang, Kecheng Zheng, Yu Liu, Deli Zhao, Jingren Zhou, and Yang Cao. Cones: Concept neurons in diffusion models for customized generation. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015. URL https://liuziwei7.github.io/projects/FaceAttributes.html.

Artem Lukoianov, Chenyang Yuan, Justin Solomon, and Vincent Sitzmann. Locality in image diffusion models emerges from data statistics. In *Advances in Neural Information Processing Systems*, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/88ad9774ffcb7a272457e9396f793a07-Abstract-Conference.html.

Song Mei. U-Nets as belief propagation: Efficient classification, denoising, and diffusion in generative hierarchical models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/cb3658b9983f677670a246c46ece553d-Abstract-Conference.html.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35:17359–17372, 2022.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023.

Yuki Monma, Kan Aro, and Muneki Yasuda. Hierarchical Gaussian Markov random field for image denoising. *IEICE Transactions on Information and Systems*, E105-D(3):689–699, 2022. doi: 10.1587/transinf.2021EDP7172.

Matthew Niedoba, Berend Zwartsenberg, Kevin Patrick Murphy, and Frank Wood. Towards a mechanistic explanation of diffusion model generalization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 46389–46411. PMLR, 2025. URL https://proceedings.mlr.press/v267/niedoba25a.html.

Maya Okawa, Ekdeep Singh Lubana, Robert P. Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. In *Advances in Neural Information Processing Systems*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Haowen Pan, Yixin Cao, Xiaozhi Wang, Xun Yang, and Meng Wang. Finding and editing multimodal neurons in pre-trained transformers. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1012–1037, 2024a.

Kaihang Pan, Zhaoyu Fan, Juncheng Li, Qifan Yu, Hao Fei, Siliang Tang, Richang Hong, Hanwang Zhang, and Qianru Sun. Towards unified multimodal editing with enhanced knowledge collaboration. In *Advances in Neural Information Processing Systems*, volume 37, 2024b.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Peebles_Scalable_Diffusion_Models_with_Transformers_ICCV_2023_paper.html.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

Jakiw Pidstrigach. Score-based generative models detect manifolds. In *Advances in Neural Information Processing Systems*, 2022.

Emile Pierret and Bruno Galerne. Diffusion models for Gaussian distributions: Exact solutions and Wasserstein errors. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 49355–49381. PMLR, 2025. URL https://proceedings.mlr.press/v267/pierret25a.html.

Paul Rosu, Lawrence Carin, and Xiang Cheng. From softmax to score: Transformers can effectively implement in-context denoising steps. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/d27af2bebeab8a3e6f3848fc71736235-Abstract-Conference.html.

Stefan Roth and Michael J. Black. Fields of experts: A framework for learning image priors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 860–867, 2005. doi: 10.1109/CVPR.2005.160. URL https://download.visinf.tu-darmstadt.de/papers/2005-cvpr-roth-fields_of_experts-preprint.pdf.

Håvard Rue and Leonhard Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC, 2005.

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 2862–2867, 2023.

Joonghyuk Shin, Alchan Hwang, Yujin Kim, Daneul Kim, and Jaesik Park. Exploring multimodal diffusion transformers for enhanced prompt-based image editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Shin_Exploring_Multimodal_Diffusion_Transformers_for_Enhanced_Prompt-based_Image_Editing_ICCV_2025_paper.html.

Hayaru Shouno. Bayesian image restoration for Poisson corrupted image using a latent variational method with Gaussian MRF. *IPSJ Online Transactions*, 8:15–24, 2015. doi: 10.2197/ipsjtrans.8.15.

Jaskirat Singh, Xingjian Leng, Zongze Wu, Liang Zheng, Richard Zhang, Eli Shechtman, and Saining Xie. What matters for representation alignment: Global information or spatial structure? In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/3929a7785bd56f57edcff0152ab41289-Abstract-Conference.html.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Hervé Jégou, and Armand Joulin. Augmenting self-attention with persistent memory. *arXiv preprint arXiv:1907.01470*, 2019.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/0503f5dce343a1d06d16ba103dd52db1-Abstract-Conference.html.

Marshall F. Tappen, Ce Liu, Edward H. Adelson, and William T. Freeman. Learning Gaussian Conditional Random Fields for low-level vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007. doi: 10.1109/CVPR.2007.382979. URL https://people.csail.mit.edu/billf/www/papers/cvpr07-gcrf.pdf.

Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing ViT features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Tumanyan_Splicing_ViT_Features_for_Semantic_Appearance_Transfer_CVPR_2022_paper.html.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1921–1930, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Tumanyan_Plug-and-Play_Diffusion_Features_for_Text-Driven_Image-to-Image_Translation_CVPR_2023_paper.html.

Evelyn Turri, Davide Bucciarelli, Sara Sarto, Lorenzo Baraldi, and Marcella Cornia. Few channels draw the whole picture: Revealing massive activations in diffusion transformers, 2026. URL https://arxiv.org/abs/2605.13974.

Binxu Wang and John J. Vastola. The unreasonable effectiveness of Gaussian score approximation for diffusion models and its applications. *Transactions on Machine Learning Research*, 2024.

Yair Weiss and William T Freeman. What makes a good model of natural images? In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series: With engineering applications*. Technology Press of the Massachusetts Institute of Technology and John Wiley & Sons, 1949.

Wikipedia contributors. Chebyshev polynomials. Wikipedia, The Free Encyclopedia, 2026. URL https://en.wikipedia.org/wiki/Chebyshev_polynomials. Accessed September 2, 2026.

Yu-Han Wu, Pierre Marion, Gérard Biau, and Claire Boyer. Taking a big step: Large learning rates in denoising score matching prevent memorization. In *Proceedings of the Thirty Eighth Conference on Learning Theory*, volume 291 of *Proceedings of Machine Learning Research*, pp. 5718–5756. PMLR, 2025. URL https://proceedings.mlr.press/v291/wu25a.html.

Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: Efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/34345e243156da67605d4b63d71c8d98-Abstract-Conference.html.

Muneki Yasuda, Junpei Watanabe, Shun Kataoka, and Kazuyuki Tanaka. Linear-time algorithm in Bayesian image denoising based on Gaussian Markov random field. *IEICE Transactions on Information and Systems*, E101-D(6):1629–1639, 2018. doi: 10.1587/transinf.2017EDP7346.

TaeHo Yoon, Joo Young Choi, Sehyun Kwon, and Ernest K. Ryu. Diffusion probabilistic models generalize when they fail to memorize. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/d9e42b4d7163931f3689d6d6fbaa11d0-Abstract-Conference.html.

Arman Zarei, Samyadeep Basu, Keivan Rezaei, Zihao Lin, Sayan Nag, and Soheil Feizi. Localizing knowledge in diffusion transformers. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/157a6107a95883f6ccc0e38e80a24295-Abstract-Conference.html.

Zekai Zhang, Xiao Li, Xiang Li, Lianghe Shi, Meng Wu, Molei Tao, and Qing Qu. Generalization of diffusion models arises with a balanced representation space. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a5e3cf29c269b041ccd644b6beaf5c42-Abstract-Conference.html.

Yi Zheng, Yifan Zhao, Mengyuan Ren, He Yan, Xiangju Lu, Junhui Liu, and Jia Li. Cartoon face recognition: A benchmark dataset. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2264–2272, 2020. doi: 10.1145/3394171.3413726. URL https://doi.org/10.1145/3394171.3413726.

APPENDIX CONTENTS

A Related Literature **17**

B Proofs **19**

B.1 Notation 19

B.2 Background facts 19

C Main proofs **20**

C.1 Full statement and proof of Theorem 1 20

C.2 Full statement and proof of Corollary 2 21

C.3 Proof of Theorem 3 22

D From the aggregator–predictor construction to implementation **23**

D.1 Patch, position, and time conditioning 24

D.2 Spatial weights and shared channel operators 24

D.3 Soft locality of Attention Aggregator 25

D.4 General predictor parameterizations 27

D.5 From a single layer to a residual stack 29

E Experimental details **29**

E.1 Experiment 1-1: Exact denoising context under synthetic GMRF priors 30

E.2 Experiment 1-2: Denoising context in DiTs trained on real-image datasets 32

E.3 Experiment 2: Replacing MLPs with shared-memory predictors 35

E.4 Experiment 3: Component compatibility in cross-dataset adaptation 38

A Related Literature

Representations and compositional reuse. Diffusion models learn internal features useful for semantic segmentation and discriminative probing (Baranchuk et al., 2022; Xiang et al., 2023), and semantic and geometric correspondence can emerge without correspondence supervision (Tang et al., 2023). Chen et al. (2025) connect this ability to classical denoising autoencoders by simplifying diffusion models and studying which components support self-supervised representation learning. Han et al. (2025a) provide theoretical and empirical evidence that denoising encourages representations covering multiple data features, including features that classifiers may neglect. REPA connects representation quality to generative training by aligning hidden states with pretrained visual features in diffusion and flow-based transformers (Yu et al., 2025). Singh et al. (2026) further identify the spatial structure of the alignment target as a strong predictor of generation performance.

Compositional generation provides evidence of reuse: Okawa et al. (2023) demonstrate unseen combinations in a synthetic task, and Favero et al. (2025) analyze the learning of composition rules in a hierarchical grammar, where longer-range dependencies require more data. The extent of this ability depends on the task; Han et al. (2025b) find that diffusion models can learn coarse relationships while failing finer dependencies, including reflection and lighting consistency. These findings motivate the introduction's representation-and-reuse framing. In our denoising account, a reuse rule has a specific meaning: it determines which context patches supply evidence for predicting a clean patch from a new noisy image.

Memorization and generalization. Under a finite empirical prior, the squared-error denoising objective is minimized by a weighted average of training images, with weights determined by whole-image comparisons as in (2). With the corresponding terminal initialization, exact reverse dynamics recover the empirical distribution (Pidstrigach, 2022; Yoon et al., 2023; Gu et al., 2025; Biroli et al., 2024; Kamb & Ganguli, 2025). Several explanations address why learned denoisers nevertheless generalize. Training dynamics can favor generalizing solutions before memorization, with larger datasets delaying memorization (Bonnaire et al., 2025); large learning rates can also steer training away from the memorizing solution (Wu et al., 2025). Other accounts characterize the learned estimator: Kadkhodaie et al. (2024) connect generalization to geometry-adaptive harmonic representations, while Zhang et al. (2026) prove a distinction between sample storage and balanced representations of local statistics for a two-layer ReLU denoising autoencoder, and investigate analogous structure in larger models. Our framework concerns how DiT components implement denoising computations compatible with such statistical structure.

Local denoising and its statistical target. Local empirical denoisers explain novel generation through patch recombination. Niedoba et al. (2025) use cropped neighborhoods to approximate trained denoisers; Kamb & Ganguli (2025) derive estimators from locality and translation equivariance; and Lukoianov et al. (2025) choose context from the sensitivity of the optimal linear denoiser, emphasizing the role of data statistics. These constructions motivate the *patchwise linear denoiser (PLD)* in Section 2. In (3), a softmax comparison through $S_{m,t}$ determines how stored training patches $x_m^{(i)}$ contribute to a prediction. Setting $S_{m,t} = I$ and $\Gamma_{m,t} = \sigma_t^{-2} I$ recovers whole-image retrieval; local contexts allow the weights to differ across patches and hence permit recombination.

Our statistical contribution identifies a population target for this construction. Theorem 1 bounds the discrepancy between population local and global denoisers under a GMRF prior. Theorem 3 identifies when sufficient aggregation lets a PLD recover the empirical local conditional mean exactly; this identity alone is not a finite-sample bound on population risk. Together, the results motivate learning both the context aggregator and the patch predictor. The stored patches and context com-

parisons in the PLD provide an explicit example of content and reuse; Section 4 develops their learned counterparts in a DiT.

Architectural accounts of denoising. Mei (2025) show how U-Net encoder–decoder paths and skip connections can implement belief-propagation denoising in hierarchical graphical models, providing a direct precedent for interpreting architecture through a statistical denoising algorithm. For transformers, Rosu et al. (2025) show that attention can implement exact score-based denoising; their image experiments use whole-image tokens, although an anisotropic extension permits patch-wise pixel updates. An et al. (2025) associate DiT generalization with local attention in early layers and show benefits from restricting attention windows with limited training data. Gan et al. (2025) link concentrated large feature activations to AdaLN and use channel modulation to improve DiT features for correspondence. These works explain architectural or conditioning mechanisms relevant to denoising and representation learning. Our account connects attention, MLPs, and AdaLN through context aggregation followed by patch prediction, allowing context to vary with the input and noise level, including the non-contiguous patterns in Section 5.1.

MLPs as learned memory. Language-model MLPs have been interpreted as key–value memories that store and retrieve learned associations (Petroni et al., 2019; Geva et al., 2021; 2022); Sukhbaatar et al. (2019) express them as attention over persistent memory vectors. This interpretation has motivated locating and editing facts through MLP weights (Dai et al., 2022; Meng et al., 2022; 2023), with related analyses and interventions in multimodal models (Schwettmann et al., 2023; Pan et al., 2024a; Basu et al., 2024a; Pan et al., 2024b). At the level of the generative process, Hoover et al. (2023) survey connections between diffusion dynamics and retrieval in energy-based associative memories. Our DiT account assigns a memory role specifically to the MLP's patch prediction from aggregated context (Section 4.2). Replacing layerwise MLPs with explicit memory shared across layers tests this interpretation at the level of the predictor component (Section 5.2). The representation may be distributed across learned memory entries; the interpretation does not require individual entries to correspond to named concepts.

Localization, editing, and component interventions. Several analyses of text-to-image diffusion localize visual attributes in text-encoder self-attention or U-Net cross-attention (Basu et al., 2024c;b), including neurons associated with concepts or memorized training images (Liu et al., 2023; Hintersdorf et al., 2024). For DiTs, Zarei et al. (2025) identify blocks associated with specific concepts and use localized updates for personalization and unlearning. Their component ablations also intervene on self-attention, normalization, and MLP outputs, with cross-attention giving the strongest localization signal in the tested PixArt setting. Thus, component intervention already has direct precedents in DiTs.

Related editing methods illustrate the practical value of these distinctions. Custom Diffusion adapts selected parameters for concept customization (Kumari et al., 2023); Plug-and-Play Diffusion Features injects spatial features and self-attention to preserve layout during translation (Tumanyan et al., 2023); and Shin et al. (2025) analyze text–image attention blocks to design editing interventions for multimodal DiTs. Turri et al. (2026) further show that transferring high-magnitude hidden-state channels between prompt-conditioned trajectories can steer image semantics in DiT and related flow-based models. These activation interventions complement our parameter transplants, which ask which QK, VO, and MLP computations remain compatible across image domains, guided by the aggregator–predictor account (Section 5.3). The transplants probe jointly adapted components with AdaLN fixed, so their outcomes measure reuse and coadaptation under that intervention protocol (Appendix E.4).

Gaussian and Markov image priors. The GMRF prior in Theorem 1 combines joint Gaussian statistics with conditional independence between non-neighboring patches given their graph neighbors. Markov image models developed from spatial statistics (Besag, 1974) into methods for restoration, texture modeling, and segmentation (Geman & Geman, 1984; Li, 2009), with higher-order models extending interactions to larger neighborhoods and patches (Roth & Black, 2005; Weiss & Freeman, 2007; Chen et al., 2014). Gaussian MRFs (Rue & Held, 2005) have been used for texture modeling (Chellappa & Chatterjee, 1985), Bayesian restoration (Shouno, 2015; Yasuda et al., 2018; Monma et al., 2022), and learned conditional denoising (Tappen et al., 2007). The Gaussian optimal denoiser is the Wiener filter (Wiener, 1949; Andrews & Hunt, 1977); its tractability also supports

exact analyses of diffusion sampling (Pierret & Galerne, 2025) and approximations to learned scores at high noise (Wang & Vastola, 2024). We use this tractable prior with patches as variables on an arbitrary graph to establish a locality guarantee, and examine the corresponding qualitative patterns on real images separately.

B PROOFS

B.1 NOTATION

For a positive integer M, let $[M] := \{1, \dots, M\}$. Vectors in $\mathbb{R}^{Md_p}$ are partitioned into M blocks of dimension d_p; x_m denotes block m of a vector x, and A_{ml} denotes the (m, l) block of a conformably partitioned matrix A. The identity matrix is denoted by I, with its dimension determined by context, and $A^\top$ denotes the transpose of A.

For a vector x, $\|x\|_2$ is its Euclidean norm. For a matrix A, $\|A\|_2 := \sup_{\|x\|_2=1} \|Ax\|_2$ is the induced operator norm and $\|A\|_F$ is the Frobenius norm. We write $\mathrm{spec}(A)$ for the spectrum of A, $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ for the smallest and largest eigenvalues of a real symmetric matrix, and $\mathrm{tr}(A)$ for its trace. For real symmetric matrices, $A \preceq B$ means that $B - A$ is positive semidefinite.

For the graph $G_0 = ([M], E_0)$, $d_{G_0}(m, l)$ is the length of a shortest path between vertices m and l, and $B_r(m) := \{l \in [M] : d_{G_0}(m, l) \le r\}$ is the closed graph ball of radius r centered at m. For a block vector x, $x_{B_r(m)}$ collects the blocks indexed by $B_r(m)$. Finally, $\mathbb{E}$ denotes expectation and $\mathcal{N}(\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and covariance Σ.

For the conditional-mean decompositions, work at fixed time t on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ carrying X_0 and the independent Gaussian noise in the corruption model, and assume $\mathbb{E}\|X_0\|_2^2 < \infty$. All Euclidean spaces are equipped with their Borel σ-fields, and measurability of maps between them is understood in this sense. For a random vector Z, write $\mathrm{sf}(Z) \subseteq \mathcal{F}$ for the σ-field it generates, and let P_{Y_t} denote the law of Y_t. Inclusions between σ-fields are understood modulo $\mathbb{P}$-null sets.

B.2 BACKGROUND FACTS

B.2.1 CHEBYSHEV POLYNOMIALS AND THE RECIPROCAL EXPANSION

The Chebyshev polynomials are a family of orthogonal polynomials on $[-1, 1]$ that are useful for constructing near-minimax polynomial approximations. We use the Chebyshev polynomials of the first kind, defined by the recurrence $T_0(s) = 1$, $T_1(s) = s$, and $T_{r+1}(s) = 2sT_r(s) - T_{r-1}(s)$ for $r \ge 1$. Equivalently, $T_r(\cos\theta) = \cos(r\theta)$ for $\theta \in [0, \pi]$. T_r has degree r and $|T_r(s)| \le 1$ for $s \in [-1, 1]$. For general background, see Wikipedia contributors (2026). The ordinary generating function for T_r is given by (DLMF, 2026, Eq. 18.12.8)

$$\sum_{r=0}^{\infty} T_r(s) z^r = \frac{1 - sz}{1 - 2sz + z^2}, \qquad |z| < 1, \quad s \in [-1, 1]. \tag{12}$$

We now introduce a useful lemma on the Chebyshev approximation of the reciprocal function with exponentially decaying error in the degree r.

Lemma 4. *Let $0 < h_- < h_+$ and $r \ge 0$. There exists a polynomial $\psi_r(x)$ of degree at most r such that*

$$\sup_{x \in [h_-, h_+]} \left| \frac{1}{x} - \psi_r(x) \right| = \frac{2}{\sqrt{h_- h_+}(1 - \rho)} \rho^{r+1}, \tag{13}$$

where $\rho = (\sqrt{h_+/h_-} - 1)/(\sqrt{h_+/h_-} + 1)$ is the Chebyshev convergence factor.

Proof. Let $s(x) = (2x - h_+ - h_-)/(h_+ - h_-)$ be the linear transformation that maps $[h_-, h_+]$ to $[-1, 1]$. Since $T_0(s) = 1$ and the series converges absolutely and uniformly over $s \in [-1, 1]$, from

(12), we have

$$1 + 2\sum_{r=1}^{\infty} z^r T_r(s) = \frac{1 - z^2}{1 - 2sz + z^2}, \qquad |z| < 1, \quad s \in [-1, 1].$$

Substituting $z = -\rho$ and $s = s(x)$ and simplifying gives

$$\frac{1}{x} = \frac{1}{\sqrt{h_- h_+}} \left(1 + 2\sum_{r=1}^{\infty} (-\rho)^r T_r(s(x)) \right), \qquad x \in [h_-, h_+].$$

Now, define the degree-r truncated Chebyshev polynomial as

$$\psi_r(x) := \frac{1}{\sqrt{h_- h_+}} \left(1 + 2\sum_{k=1}^{r} (-\rho)^k T_k(s(x)) \right). \tag{14}$$

Then, since $|T_k(s(x))| \le 1$, we have, for $x \in [h_-, h_+]$,

$$\left| \frac{1}{x} - \psi_r(x) \right| = \frac{2}{\sqrt{h_- h_+}} \left| \sum_{k=r+1}^{\infty} (-\rho)^k T_k(s(x)) \right| \le \frac{2}{\sqrt{h_- h_+}} \sum_{k=r+1}^{\infty} \rho^k = \frac{2}{\sqrt{h_- h_+}(1-\rho)} \rho^{r+1}.$$

At $x = h_-$, we have $s(h_-) = -1$ and $T_k(-1) = (-1)^k$, so $(-\rho)^k T_k(s(h_-)) = \rho^k$ for all $k \ge 1$, and hence the upper bound is tight. □

C Main proofs

C.1 Full statement and proof of Theorem 1

Theorem 1 (Exponential locality of GMRF; full statement)**.** *Suppose $X_0 \sim \mathcal{N}(0, Q^{-1})$, where $Q \in \mathbb{R}^{Md_p \times Md_p}$ is symmetric positive definite and its off-diagonal patch blocks vanish outside the edges of $G_0 = ([M], E_0)$. Let $X_t = a_t X_0 + b_t \varepsilon$ with $a_t, b_t > 0$ and independent $\varepsilon \sim \mathcal{N}(0, I_{Md_p})$, and set $Y_t = X_t/a_t$ and $\sigma_t = b_t/a_t$. For $B_r(m) = \{l : d_{G_0}(m, l) \le r\}$, define*

$$D^*_{m,t}(y) = \mathbb{E}[X_{0,m} \mid Y_t = y, t], \qquad D^{*,r}_{m,t}(y) = \mathbb{E}[X_{0,m} \mid Y_{t,B_r(m)} = y_{B_r(m)}, t].$$

Let $q_- := \lambda_{\min}(Q)$ and $q_+ := \lambda_{\max}(Q)$ satisfy $0 < q_- < q_+ < \infty$. Then for every patch m and every integer radius $r \ge 0$,

$$\mathbb{E} \left\| D^*_{m,t}(Y_t) - D^{*,r}_{m,t}(Y_t) \right\|_2^2 \le \frac{d_p}{q_-} \left[\frac{\left(1 - 1/\sqrt{\chi_t}\right)^2}{1 + \sigma_t^2 q_-} \right] \rho_t^{2r}, \tag{15}$$

where $\chi_t = (1 + \sigma_t^2 q_+)/(1 + \sigma_t^2 q_-)$ is the condition number of $I + \sigma_t^2 Q$ and $\rho_t = (\sqrt{\chi_t} - 1)/(\sqrt{\chi_t} + 1)$ is the Chebyshev convergence factor. In the notation of Assump. 1, the bound is $C_t \exp(-c_t r)$ with

$$C_t = \frac{d_p (1 - 1/\sqrt{\chi_t})^2}{q_-(1 + \sigma_t^2 q_-)}, \qquad c_t = 2\log(1/\rho_t) > 0.$$

If $q_- = q_+$, then $Q = q_- I$ and radius zero is exact.

Proof. Since $Y_t = X_0 + \sigma_t \varepsilon$, Bayes' rule gives posterior precision $Q + \sigma_t^{-2} I = \sigma_t^{-2}(I + \sigma_t^2 Q)$ and posterior mean

$$D^*_{m,t}(y) = \left[(I + \sigma_t^2 Q)^{-1} y \right]_m.$$

Let $H_t := I + \sigma_t^2 Q$. The matrix H_t is real symmetric positive definite, and

$$\operatorname{spec}(H_t) \subseteq [1 + \sigma_t^2 q_-, 1 + \sigma_t^2 q_+].$$

Write its spectral decomposition as $H_t = U \Lambda_t U^\top$, where U is orthogonal and Λ_t is diagonal. We first approximate the scalar inverse function with the degree-r polynomial in (14), and then apply that polynomial to H_t.

Take $h_{-,t} := 1 + \sigma_t^2 q_-$ and $h_{+,t} := 1 + \sigma_t^2 q_+$, so that $\chi_t = h_{+,t}/h_{-,t}$. Then by the Chebyshev approximation in (13), the degree-r Chebyshev polynomial $\psi_{r,t}(x)$ as given in (14) (with h_- and h_+ replaced by $h_{-,t}$ and $h_{+,t}$) satisfies

$$\begin{aligned}\|H_t^{-1} - \psi_{r,t}(H_t)\|_2 &= \|U\left(\Lambda_t^{-1} - \psi_{r,t}(\Lambda_t)\right)U^\top\|_2 \\ &= \max_{\lambda \in \mathrm{spec}(H_t)} \left|\frac{1}{\lambda} - \psi_{r,t}(\lambda)\right| \\ &\le \frac{2\rho_t^{r+1}}{\sqrt{h_{-,t}h_{+,t}}(1-\rho_t)}.\end{aligned}$$

On the other hand, H_t has the same off-diagonal block sparsity pattern as Q, so, for every integer $k \ge 0$, $[H_t^k]_{ml} = 0$ whenever $d_{G_0}(m,l) > k$. Indeed, each nonzero term in the blockwise expansion of $[H_t^k]_{ml}$ corresponds to a length-k walk from m to l, with diagonal blocks allowing the walk to remain at its current vertex. Because $\psi_{r,t}$ has degree at most r, the predictor

$$\widetilde{D}^r_{m,t}(y) := [\psi_{r,t}(H_t)y]_m$$

depends only on $y_{B_r(m)}$ and is therefore measurable with respect to $Y_{t,B_r(m)}$. Graph connectivity determines which patches lie within each radius, while the conditioning of $I + \sigma_t^2 Q$ controls the decay rate: worse conditioning slows the certified decay and can require broader context.

By the tower property, $D^{*,r}_{m,t}(Y_t) = \mathbb{E}[D^*_{m,t}(Y_t) \mid Y_{t,B_r(m)}, t]$. The projection property of conditional expectation therefore implies

$$\mathbb{E}\|D^*_{m,t}(Y_t) - D^{*,r}_{m,t}(Y_t)\|_2^2 \le \mathbb{E}\|D^*_{m,t}(Y_t) - \widetilde{D}^r_{m,t}(Y_t)\|_2^2.$$

Construct $P_m \in \mathbb{R}^{d_p \times Md_p}$ so that it selects patch m of a noisy image $y \in \mathbb{R}^{Md_p}$, i.e., $P_m y = y_m$. Then $D^*_{m,t}(Y_t) - \widetilde{D}^r_{m,t}(Y_t) = P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)Y_t$. Since Y_t is zero-mean Gaussian with covariance $Q^{-1} + \sigma_t^2 I$, we have

$$\begin{aligned}\mathbb{E}\|D^*_{m,t}(Y_t) - \widetilde{D}^r_{m,t}(Y_t)\|_2^2 &= \mathrm{tr}\left(P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)(Q^{-1} + \sigma_t^2 I)\left(H_t^{-1} - \psi_{r,t}(H_t)\right)P_m^\top\right) \\ &= \left\|P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)(Q^{-1} + \sigma_t^2 I)^{1/2}\right\|_F^2 \\ &\le \|Q^{-1} + \sigma_t^2 I\|_2 \left\|P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)\right\|_F^2.\end{aligned}$$

Since $P_m(H_t^{-1} - \psi_{r,t}(H_t)) \in \mathbb{R}^{d_p \times Md_p}$ has d_p rows, the Frobenius–operator norm inequality and the Chebyshev approximation bound give

$$\begin{aligned}\left\|P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)\right\|_F^2 &\le d_p \left\|P_m\left(H_t^{-1} - \psi_{r,t}(H_t)\right)\right\|_2^2 \\ &\le d_p \left\|H_t^{-1} - \psi_{r,t}(H_t)\right\|_2^2 \\ &\le d_p \frac{4\rho_t^{2r+2}}{h_{-,t}h_{+,t}(1-\rho_t)^2}.\end{aligned}$$

Also, since $Q^{-1} + \sigma_t^2 I$ is real symmetric positive definite,

$$\|Q^{-1} + \sigma_t^2 I\|_2 = \lambda_{\max}(Q^{-1} + \sigma_t^2 I) = q_-^{-1} + \sigma_t^2.$$

Combining the above bounds, we have

$$\mathbb{E}\|D^*_{m,t}(Y_t) - \widetilde{D}^r_{m,t}(Y_t)\|_2^2 \le d_p(q_-^{-1} + \sigma_t^2)\frac{4\rho_t^{2r+2}}{h_{-,t}h_{+,t}(1-\rho_t)^2}.$$

Finally, the identities

$$q_-^{-1} + \sigma_t^2 = \frac{h_{-,t}}{q_-}, \qquad \frac{2\rho_t}{\sqrt{h_{-,t}h_{+,t}}(1-\rho_t)} = \frac{1 - 1/\sqrt{\chi_t}}{h_{-,t}}$$

reduce the preceding display to (15). □

C.2 Full statement and proof of Corollary 2

Corollary 2 (Bounded-range monotonicity of the locality radius; full statement)**.** *Under the hypotheses of the full Thm. 1 above, with $0 < q_- < q_+$, define $\sigma_{\mathrm{abs}}^2 := (q_-^{1/3} + q_+^{1/3})/(q_-^{2/3} q_+^{2/3})$. For $\epsilon > 0$ and $0 < \sigma_t \le \sigma_{\mathrm{abs}}$, the smallest radius $r_\epsilon(\sigma_t)$ such that the upper bound in* (15) *is at most ϵ is the least nonnegative integer satisfying*

$$r_\epsilon(\sigma_t) \ge \frac{1}{2} \log \left(\frac{d_p \left(1 - 1/\sqrt{\chi_t}\right)^2}{\epsilon q_- (1 + \sigma_t^2 q_-)} \right) \Big/ \log \left(\frac{1}{\rho_t} \right) \tag{16}$$

The resulting $r_\epsilon(\sigma_t)$ is nondecreasing in σ_t on the same range. At $\sigma_t = 0$, radius zero is exact.

Proof. We show that the upper bound in (15) is nondecreasing in σ_t for $0 < \sigma_t \le \sigma_{\mathrm{abs}}$. First, for $\sigma_t > 0$,

$$\frac{1}{\chi_t} \frac{d\chi_t}{d\sigma_t} = \frac{2\sigma_t (q_+ - q_-)}{(1 + \sigma_t^2 q_+)(1 + \sigma_t^2 q_-)} \ge 0,$$

and hence $d\chi_t / d\sigma_t \ge 0$. Next,

$$\begin{aligned} \frac{d}{d\sigma_t} \log \left[\frac{\left(1 - 1/\sqrt{\chi_t}\right)^2}{(1 + \sigma_t^2 q_-)} \right] &= \frac{1}{\sqrt{\chi_t} - 1} \left(\frac{1}{\chi_t} \frac{d\chi_t}{d\sigma_t} \right) - \frac{2\sigma_t q_-}{1 + \sigma_t^2 q_-} \\ &= \frac{2\sigma_t}{1 + \sigma_t^2 q_-} \left[\frac{q_+ - q_-}{(1 + \sigma_t^2 q_+)(\sqrt{\chi_t} - 1)} - q_- \right] \end{aligned}$$

is nonnegative when $q_+ - q_- \ge q_- (1 + \sigma_t^2 q_+)(\sqrt{\chi_t} - 1)$. Using

$$1 + \sigma_t^2 q_+ = \chi_t \frac{q_+ - q_-}{q_+ - q_- \chi_t}, \qquad q_+ - q_- \chi_t = \frac{q_+ - q_-}{1 + \sigma_t^2 q_-},$$

this condition is equivalent to $q_+ \ge q_- \chi_t^{3/2}$, which simplifes to $\sigma_t^2 \le \sigma_{\mathrm{abs}}^2$.

For the rest of the term in the upper bound, it is straightforward to see that $d\rho_t / d\chi_t \ge 0$, and since $d\chi_t / d\sigma_t \ge 0$, we have $d\rho_t / d\sigma_t \ge 0$. Consequently, if $0 < \sigma_1 \le \sigma_2 \le \sigma_{\mathrm{abs}}$, every radius whose upper bound is at most ϵ at σ_2 also has upper bound at most ϵ at σ_1. Hence $r_\epsilon(\sigma_2) \ge r_\epsilon(\sigma_1)$. □

C.3 Proof of Theorem 3

Proof. Fix m and t with $\sigma_t > 0$, and use the stacked-vector convention throughout. Under the empirical prior, let J be uniform on $[n]$ and set $X_0 = x^{(J)}$. Since $Y_t = X_0 + \sigma_t \varepsilon$, the linear aggregate satisfies $\mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t) \mid J = i \sim \mathcal{N}\big(S_{m,t} x^{(i)}, \sigma_t^2 S_{m,t} S_{m,t}^\top\big)$. The covariance is positive definite because $\sigma_t > 0$ and $S_{m,t}$ has full row rank. With the prescribed precision $\Gamma_{m,t} = (\sigma_t^2 S_{m,t} S_{m,t}^\top)^{-1}$, the Gaussian log-likelihood is $-\frac{1}{2} \|s - S_{m,t} x^{(i)}\|_{\Gamma_{m,t}}^2$ up to a term independent of i. The uniform prior and common Gaussian normalizing factor cancel in Bayes' rule, giving

$$\Pr_{\hat{p}_0} \left[J = i \mid \mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t) = s, t \right] = \mathrm{sm}_i \left\{ -\frac{1}{2} \left\| s - \mathcal{S}_{m,t}^{\mathrm{lin}}(x^{(j)}) \right\|_{\Gamma_{m,t}}^2 \right\}_{j \in [n]}.$$

Consequently,

$$\mathbb{E}_{\hat{p}_0} \left[X_{0,m} \mid \mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t) = s, t \right] = \sum_{i=1}^{n} \Pr_{\hat{p}_0} \left[J = i \mid \mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t) = s, t \right] x_m^{(i)} = g_{m,t}^{\mathrm{rbf}}(s).$$

Because the aggregate is a measurable function of $Y_{t,\mathcal{B}_{m,t}}$, the tower property gives the first equality below. The assumed sigma-field inclusion makes the inner conditional mean measurable with respect to the aggregate, giving the second equality:

$$\begin{aligned} \mathbb{E}_{\hat{p}_0} \left[X_{0,m} \mid \mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t), t \right] &= \mathbb{E}_{\hat{p}_0} \left[\mathbb{E}_{\hat{p}_0} \left[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}}, t \right] \mid \mathcal{S}_{m,t}^{\mathrm{lin}}(Y_t), t \right] \\ &= \mathbb{E}_{\hat{p}_0} \left[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}}, t \right]. \end{aligned}$$

Together with $\hat{D}_{m,t}^{\mathrm{pld}} = g_{m,t}^{\mathrm{rbf}} \circ \mathcal{S}_{m,t}^{\mathrm{lin}}$, this proves the theorem.

To make the inclusion condition concrete, restrict $S_{m,t}$ to its context columns, denoted by $[S_{m,t}]_{\mathcal{B}_{m,t}}$. Locality implies $S_{m,t}y = [S_{m,t}]_{\mathcal{B}_{m,t}} y_{\mathcal{B}_{m,t}}$. Bayes' rule for the full context gives

$$\begin{aligned}\Pr_{\hat{p}_0}\left[J = i \mid Y_{t,\mathcal{B}_{m,t}} = y_{\mathcal{B}_{m,t}}, t\right] &= \mathrm{sm}_i\left\{-\frac{1}{2\sigma_t^2}\left\|y_{\mathcal{B}_{m,t}} - x^{(j)}_{\mathcal{B}_{m,t}}\right\|_2^2\right\}_{j\in[n]} \\ &= \mathrm{sm}_i\left\{\frac{2y_{\mathcal{B}_{m,t}}^\top\left(x^{(j)}_{\mathcal{B}_{m,t}} - x^{(1)}_{\mathcal{B}_{m,t}}\right) - \|x^{(j)}_{\mathcal{B}_{m,t}}\|_2^2 + \|x^{(1)}_{\mathcal{B}_{m,t}}\|_2^2}{2\sigma_t^2}\right\}_{j\in[n]}.\end{aligned}$$

The second equality subtracts the log-likelihood of sample 1 from every logit, which leaves the softmax unchanged. Thus, the posterior weights depend on $y_{\mathcal{B}_{m,t}}$ only through its inner products with the training-context differences. Suppose that

$$V_{m,t} := \mathrm{span}\left\{x^{(j)}_{\mathcal{B}_{m,t}} - x^{(1)}_{\mathcal{B}_{m,t}} : j \in [n]\right\} \subseteq \mathrm{row}\big([S_{m,t}]_{\mathcal{B}_{m,t}}\big)\,. \tag{17}$$

For each j, there is then a vector $a_j \in \mathbb{R}^{d_h}$ such that $x^{(j)}_{\mathcal{B}_{m,t}} - x^{(1)}_{\mathcal{B}_{m,t}} = [S_{m,t}]^\top_{\mathcal{B}_{m,t}} a_j$. Hence $y^\top_{\mathcal{B}_{m,t}}(x^{(j)}_{\mathcal{B}_{m,t}} - x^{(1)}_{\mathcal{B}_{m,t}}) = a_j^\top [S_{m,t}]_{\mathcal{B}_{m,t}} y_{\mathcal{B}_{m,t}} = a_j^\top S_{m,t} y$. Every posterior weight is therefore a continuous function of $S_{m,t}y$, as is their weighted sum $\mathbb{E}_{\hat{p}_0}[X_{0,m} \mid Y_{t,\mathcal{B}_{m,t}} = y_{\mathcal{B}_{m,t}}, t]$. This proves the required sigma-field inclusion: the local conditional mean can be recovered from the aggregate even when the full context cannot.

This construction is strictly compressive when $0 < v_{m,t} := \dim V_{m,t} < |\mathcal{B}_{m,t}|d_p$: choose the $v_{m,t}$ rows of $[S_{m,t}]_{\mathcal{B}_{m,t}}$ to be an orthonormal basis of $V_{m,t}$, so that $d_h = v_{m,t} < |\mathcal{B}_{m,t}|d_p$. All clean training contexts have the same projection onto $V_{m,t}^\perp$. Under isotropic Gaussian corruption, the discarded coordinates consequently contribute the same likelihood factor for every sample index, which cancels from the posterior weights. For example, if $|\mathcal{B}_{m,t}|d_p = 2$ and $x^{(j)}_{\mathcal{B}_{m,t}} = (u_j, c)$ with fixed c and varying u_j, then $[S_{m,t}]_{\mathcal{B}_{m,t}} = (1, 0)$ retains only the first noisy coordinate and still determines all posterior weights. If $v_{m,t} = 0$, all training contexts coincide and the local conditional mean is constant. In general $v_{m,t} \le \min(n-1, |\mathcal{B}_{m,t}|d_p)$, but $v_{m,t}$ can equal $|\mathcal{B}_{m,t}|d_p$; in that case this row-space condition requires retaining the full context dimension and provides no strict compression. We note that the condition (17) is sufficient, not necessary for the theorem: it preserves the entire posterior over sample indices, whereas the theorem requires only preservation of the target-patch conditional mean. □

D From the aggregator–predictor construction to implementation

This section connects the aggregator–predictor construction in Secs. 4 to the implemented architectures. We first expand a single attention branch applied to the noisy patches Y_t, then explain how aggregation and patch prediction extend to residual stacks, and finally give the full predictor parameterizations. All patches, tokens, embeddings, biases, and aggregates are column vectors, consistent with Appendix B.1; projection matrices act from the left. The width d_h denotes the full token and aggregate dimension, and the number of attention heads $\tilde{H}$ divides d_h.

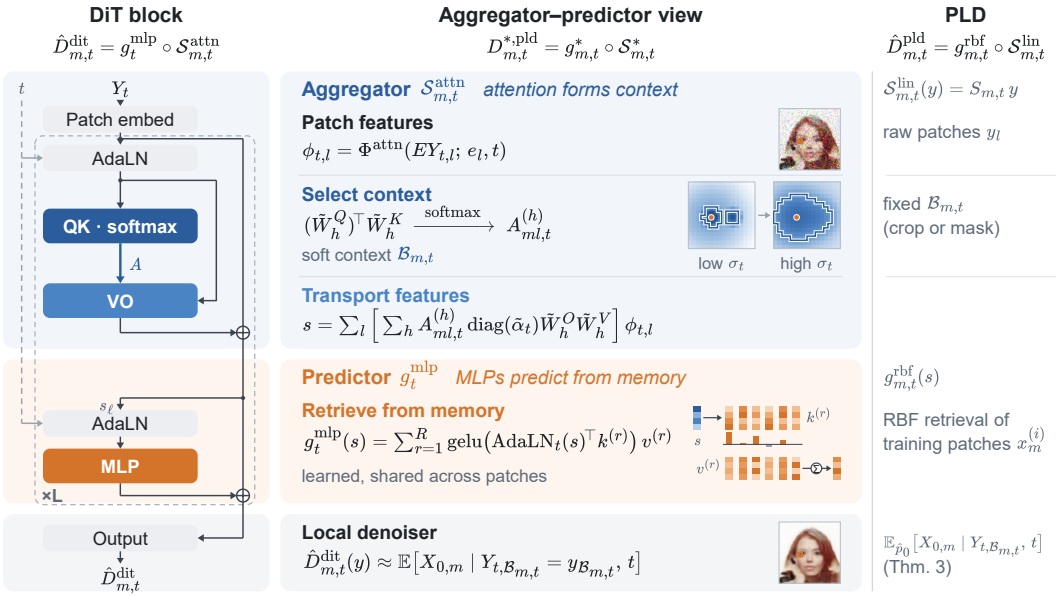

Figure 6: **Component map for the aggregator–predictor view.** At the branch level, AdaLN-conditioned QK supplies spatial weights, VO transforms the weighted features, and the MLP admits a learned-memory interpretation for tokenwise prediction. The patchwise linear denoiser (PLD) instead uses a prescribed context and retrieves training patches.

D.1 Patch, position, and time conditioning

The initial token at patch l is $EY_{t,l} + e_l$, where $E \in \mathbb{R}^{d_h \times d_p}$ and $e_l \in \mathbb{R}^{d_h}$. A shared patch-projection bias, if present, can be included in each e_l. If the network receives $X_t = a_t Y_t$ or another known scalar preconditioning of Y_t, include that time-dependent scalar before the patch projection; the same derivation applies.

Let $\tau(t) \in \mathbb{R}^{d_h}$ denote a fixed sinusoidal time embedding. The time-embedding and the following AdaLN parameters can be written as

$$e^{\mathrm{time}}_t = W^{\mathrm{time}}_2 \,\mathrm{SiLU}\big(W^{\mathrm{time}}_1 \tau(t) + b^{\mathrm{time}}_1\big) + b^{\mathrm{time}}_2, \tag{18}$$

$$\begin{bmatrix}\tilde{\beta}_t^\top & \tilde{\gamma}_t^\top & \tilde{\alpha}_t^\top & \beta_t^\top & \gamma_t^\top & \alpha_t^\top\end{bmatrix}^\top = W^{\mathrm{ada}} e^{\mathrm{time}}_t + b^{\mathrm{ada}}. \tag{19}$$

Here $W^{\mathrm{time}}_1 \in \mathbb{R}^{4d_h \times d_h}$, $W^{\mathrm{time}}_2 \in \mathbb{R}^{d_h \times 4d_h}$, and their biases have dimensions $4d_h$ and d_h, respectively. Also $W^{\mathrm{ada}} \in \mathbb{R}^{6d_h \times d_h}$ and $b^{\mathrm{ada}} \in \mathbb{R}^{6d_h}$, so every shift β, scale adjustment γ, and residual gate α is a column in $\mathbb{R}^{d_h}$. The tilded rows condition spatial attention; the untilded rows condition the predictor branch. Layer indices are suppressed in this single-layer expansion; in a stack, the AdaLN map and both sets of conditioning parameters are layer-specific.

With these notations, the feature entering attention can be written as

$$\phi_{t,l}(Y_t) := \Phi^{\mathrm{attn}}(EY_{t,l}; e_l, t) = \big(1 + \tilde{\gamma}_t\big) \odot \mathrm{LN}(EY_{t,l} + e_l) + \tilde{\beta}_t. \tag{20}$$

Layer normalization acts over the d_h channels of this token, with a positive numerical stabilizer and no learned affine parameters; $\odot$ denotes elementwise multiplication. Consequently, at fixed t, this feature depends only on patch l and its position. The gate $\tilde{\alpha}_t$ is applied after attention output mixing.

D.2 Spatial weights and shared channel operators

For head h, the Q, K, and V matrices have shape $(d_h/\tilde{H}) \times d_h$, and the output projection matrix $\tilde{W}^O_h$ has shape $d_h \times (d_h/\tilde{H})$. Including the QK biases $\tilde{b}^Q_h, \tilde{b}^K_h \in \mathbb{R}^{d_h/\tilde{H}}$, we define the scalar spatial

weights

$$A^{(h)}_{ml,t}(Y_t) = \mathrm{sm}_l \left\{ \frac{\langle \tilde{W}^Q_h \phi_{t,m}(Y_t) + \tilde{b}^Q_h, \tilde{W}^K_h \phi_{t,j}(Y_t) + \tilde{b}^K_h \rangle}{\sqrt{d_h/\tilde{H}}} \right\}_{j\in[M]} . \tag{21}$$

These weights are positive and sum to one over l. Without QK biases, the numerator is $\phi_{t,m}(Y_t)^\top (\tilde{W}^Q_h)^\top \tilde{W}^K_h \phi_{t,l}(Y_t)$. The matrix $(\tilde{W}^Q_h)^\top \tilde{W}^K_h \in \mathbb{R}^{d_h \times d_h}$ therefore controls compatibility between query and context features.

Rearranging the finite head and patch sums gives

$$\begin{aligned} \mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t) &= \mathrm{diag}(\tilde{\alpha}_t) \sum_{h=1}^{\tilde{H}} \tilde{W}^O_h \left(\sum_{l=1}^{M} A^{(h)}_{ml,t}(Y_t) \tilde{W}^V_h \phi_{t,l}(Y_t) \right) \\ &= \sum_{l=1}^{M} \left[\sum_{h=1}^{\tilde{H}} A^{(h)}_{ml,t}(Y_t)\, \mathrm{diag}(\tilde{\alpha}_t) \tilde{W}^O_h \tilde{W}^V_h \right] \phi_{t,l}(Y_t). \end{aligned} \tag{22}$$

Each bracketed matrix in the second line is $d_h \times d_h$, and each summand is a column in $\mathbb{R}^{d_h}$. Setting the QK biases to zero recovers (7); with biases, the same decomposition holds using the weights in (21). If included, the value biases $\tilde{b}^V_h \in \mathbb{R}^{d_h/\tilde{H}}$ and output bias $\tilde{b}^O \in \mathbb{R}^{d_h}$ add the column offset $\mathrm{diag}(\tilde{\alpha}_t)\big(\sum_h \tilde{W}^O_h \tilde{b}^V_h + \tilde{b}^O\big)$, because each attention row sums to one. This offset depends on time but not on the noisy image or patch location, so it does not change the spatial selection mechanism.

As explained in Sec. 4.1, we can read off the roles of QK and VO from (22). At fixed t, every $S^{\mathrm{attn}}_{ml}(Y_t; t)$ lies in the linear subspace spanned by the time-conditioned VO operators, $\mathrm{span}\{\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}^O_h \tilde{W}^V_h : h \in [\tilde{H}]\}$, whose dimension is at most $\min(\tilde{H}, d_h^2)$. Thus, each head supplies a patch-shared, time-conditioned channel operator $\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}^O_h \tilde{W}^V_h$ and an *independently normalized* spatial weighting. Before output mixing, a head forms a convex combination of its projected patch features $\tilde{W}^V_h \phi_{t,l}(Y_t)$. However, the full branch (22) need not be a convex combination of the patch features, since the channel operators are unconstrained. With the value vectors and channel operators held fixed, the attention matrix A_h determines how the head distributes across the value vectors associated with different patches. QK therefore directly controls which patches receive more or less exposure in the constructed context, even though the downstream effect of an individual weight also depends on value magnitudes, channel operators, and residual interactions.

D.3 Soft locality of Attention Aggregator

Section D.2 explained how, for fixed features, QK controls the relative logits and hence which patches receive appreciable weight. Accordingly, low-scoring patches are suppressed smoothly by exponentiation and normalization, rather than removed by a literal threshold. This provides an input-adaptive analogue of a local selector, but without explicit thresholding unmasked attention remains globally supported and need not solely concentrate on a local context set.

To relate this interpretation to our definition of locality, fix a nonempty context $\mathcal{B}_{m,t}$ containing m and mask every head to that context before softmax. Define $\mathcal{S}^{\mathrm{attn},\mathcal{B}}_{m,t}$ using the same features, channel operators, and biases, but with weights proportional to $A^{(h)}_{ml,t}(Y_t)$ over $l \in \mathcal{B}_{m,t}$ and zero otherwise. The renormalization cancels the original global softmax denominator. Since each retained logit depends only on patches m and l, this masked branch is exactly $\mathcal{B}_{m,t}$-local. The following bound records the expected squared error of this hard-threshold masking approximation. The bound depends on discarded mass over the observation law, together with the feature and channel-operator magnitudes. The notation $\lesssim_t$ hides a constant depending only on the fixed network parameters and time t, not on the observation or the context.

Theorem 5 (Expected squared error of context masking). *Fix t, a patch m, the attention parameters, and a nonempty context $\mathcal{B}_{m,t} \subseteq [M]$ containing m. Then $\mathcal{S}^{\mathrm{attn},\mathcal{B}}_{m,t}$ is $\mathcal{B}_{m,t}$-local, and*

$$\mathbb{E}\left\|\mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t) - \mathcal{S}^{\mathrm{attn},\mathcal{B}}_{m,t}(Y_t)\right\|_2^2 \lesssim_t \sum_{h=1}^{\tilde{H}} \mathbb{E}\left[\left(\sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)^2\right]. \tag{23}$$

All expectations are over the observation law of Y_t at the fixed time t.

Proof. Before applying the time-conditioned scale and shift, the layer-normalized feature has norm at most $\sqrt{d_h}$, so (20) gives

$$\max_{l\in[M]} \|\phi_{t,l}(Y_t)\|_2 \le \sqrt{d_h}\|1+\tilde{\gamma}_t\|_\infty + \|\tilde{\beta}_t\|_2.$$

Fix an observation Y_t. For each head, let μ_h^{in} and μ_h^{out} be the attention-weighted averages of $\phi_{t,l}(Y_t)$ over the retained and discarded patches, respectively, each normalized by its total attention mass. The retained mass is positive because the context is nonempty and softmax weights are positive. If the discarded set is empty, take $\mu_h^{\mathrm{out}} = 0$, in which case the approximation is exact. Note that both averages satisfy the same norm bound as the features. Also, the masked head average is μ_h^{in}, since the retained weights are renormalized to sum to one. Thus the difference between the full and masked head averages is

$$\begin{aligned}
&\sum_{l=1}^{M} A^{(h)}_{ml,t}(Y_t)\phi_{t,l}(Y_t) - \mu_h^{\mathrm{in}} \\
&\quad= \left(1 - \sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)\mu_h^{\mathrm{in}} + \left(\sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)\mu_h^{\mathrm{out}} - \mu_h^{\mathrm{in}} \\
&\quad= \left(\sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)(\mu_h^{\mathrm{out}} - \mu_h^{\mathrm{in}}).
\end{aligned}$$

The value- and output-bias offsets are the same in both branches, since both the full and masked attention rows sum to one, so these offsets cancel. The triangle inequality and the spectral-norm bound therefore give

$$\begin{aligned}
&\left\|\mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t) - \mathcal{S}^{\mathrm{attn},\mathcal{B}}_{m,t}(Y_t)\right\|_2 \\
&\quad\le \sum_{h=1}^{\tilde{H}} \left\|\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}_h^O\tilde{W}_h^V\right\|_2 \left(\sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)\|\mu_h^{\mathrm{out}} - \mu_h^{\mathrm{in}}\|_2 \\
&\quad\le 2\left(\sqrt{d_h}\|1+\tilde{\gamma}_t\|_\infty + \|\tilde{\beta}_t\|_2\right)\sum_{h=1}^{\tilde{H}} \left\|\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}_h^O\tilde{W}_h^V\right\|_2 \sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t).
\end{aligned}$$

Squaring and taking expectations gives

$$\begin{aligned}
&\mathbb{E}\left\|\mathcal{S}^{\mathrm{attn}}_{m,t}(Y_t) - \mathcal{S}^{\mathrm{attn},\mathcal{B}}_{m,t}(Y_t)\right\|_2^2 \\
&\quad\le 4\left(\sqrt{d_h}\|1+\tilde{\gamma}_t\|_\infty + \|\tilde{\beta}_t\|_2\right)^2 \mathbb{E}\left[\left(\sum_{h=1}^{\tilde{H}} \left\|\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}_h^O\tilde{W}_h^V\right\|_2 \sum_{l\notin\mathcal{B}_{m,t}} A^{(h)}_{ml,t}(Y_t)\right)^2\right].
\end{aligned}$$

Applying Cauchy–Schwarz to the head sum yields (23), with implicit constant

$$4\left(\sqrt{d_h}\|1+\tilde{\gamma}_t\|_\infty + \|\tilde{\beta}_t\|_2\right)^2 \sum_{h=1}^{\tilde{H}} \left\|\mathrm{diag}(\tilde{\alpha}_t)\tilde{W}_h^O\tilde{W}_h^V\right\|_2^2.$$

Finally, after masking, the softmax denominator contains only retained logits. Each such logit depends only on patches m and $l \in \mathcal{B}_{m,t}$, and each retained feature depends only on its own patch. Thus the masked branch depends only on $Y_{t,\mathcal{B}_{m,t}}$, establishing locality. □

D.4 General predictor parameterizations

We now write the full formulation of the shared-memory patch predictors at layer $\ell \in [L]$, where L denotes the number of layers, expanding the AdaLN conditioning and restore the affine biases, output gates, separate output projections, and multiple heads, suppressed in Sec. 4.2. We let s denote the input to a patch predictor at layer ℓ. Its time-conditioned version is $\mathrm{AdaLN}_{\ell,t}(s) = (1+\gamma_{\ell,t}) \odot \mathrm{LN}_\ell(s) + \beta_{\ell,t}$. The AdaLN scale and shift, together with the channelwise output gate $\alpha_{\ell,t}$, are supplied by the untilded rows in (19).

For compact notation, suppressing the layer index and biases, define the retrieval functions

$$\kappa_t^{\mathrm{sm}}(s,k) := \exp\big(\mathrm{AdaLN}_t(s)^\top k\big), \qquad \kappa_t^{\mathrm{gelu}}(s,k) := \mathrm{gelu}\big(\mathrm{AdaLN}_t(s)^\top k\big).$$

Softmax retrieval normalizes κ_t^{sm} over memory entries, whereas GELU retrieval uses κ_t^{gelu} directly. The full expressions below make the headwise projections, scaling, biases, and output gates explicit.

Throughout the predictors, we fix the same token dimension d_h. The predictors are specified by the following design choices:

Notation	Design choice
H	Number of predictor heads
R	Number of learnable memory entries
d'_h	Width of learnable memory

The ordinary MLP illustrates these choices. It has a single predictor head $H = 1$, and $R = 4d_h$ layer-specific keys and values $\{k_\ell^{(r)}, v_\ell^{(r)} \in \mathbb{R}^{d'_h}\}_{r=1}^R$ of width $d'_h = d_h$:

$$g_{\ell,t}^{\mathrm{mlp}}(s) = \alpha_{\ell,t} \odot \left[\sum_{r=1}^{R} \mathrm{gelu}\Big(\mathrm{AdaLN}_{\ell,t}(s)^\top k_\ell^{(r)} + b_\ell^{(r)}\Big)\, v_\ell^{(r)} + b_\ell^O\right]. \tag{24}$$

These keys and values are shared across patches within a layer but are learned independently at each layer. If $H > 1$, as with multi-head attention, each predictor head projects queries, keys, and values to width d_h/H, assuming d_h is divisible by H.

Table 1 compares the memory banks and layerwise parameter counts for each predictor across all L layers. The spatial-attention backbone and AdaLN parameters are common to all four models and are excluded.

Predictor	Memory bank	Bank parameters	Layerwise parameters
LPLD $g_{\ell,m,t}^{\mathrm{lpld}}$	$\{z^{(r)}\}_{r=1}^R$	RMd'_h	$3d_h^2 + d_h d'_h + 4d_h$
SCA $g_{\ell,t}^{\mathrm{sca}}$	$\{k^{(r)}, v^{(r)}\}_{r=1}^R$	$2Rd'_h$	$2d_h(d_h + d'_h) + 5d_h + H$
SMLP $g_{\ell,t}^{\mathrm{smlp}}$	$\{k^{(r)}, v^{(r)}\}_{r=1}^R$	$2Rd'_h$	$2d_h(d_h + d'_h) + Hd'_h + 2d_h$
MLP $g_{\ell,t}^{\mathrm{dit}}$	$\{k_\ell^{(r)}, v_\ell^{(r)}\}_{\ell,r}$	$2LRd_h$	0

Table 1: Predictor parameter comparison. The layerwise column includes Q/K/V/output projections, their biases, and SCA's headwise gain parameters at each layer. For LPLD, a learned compatibility map adds $d_h d'_h$ parameters once; this cost is zero when $A = I$.

D.4.1 Learned PLD (LPLD)

At each layer, LPLD uses cross-attention to retrieve from a memory bank $\{z^{(r)} \in \mathbb{R}^{Md'_h}\}_{r=1}^R$ shared across layers. Each entry $z^{(r)} = [z_1^{(r)}, \ldots, z_M^{(r)}]$ contains M memory patches $z_m^{(r)} \in \mathbb{R}^{d'_h}$, whose width d'_h may differ from the token width d_h. The query is projected from the conditioned layer input $\mathrm{AdaLN}_{\ell,t}(s)$, while the value is projected directly from the raw memory patch $z_m^{(r)}$. To form the key, a layer-shared compatibility map $A \in \mathbb{R}^{d_h \times d'_h}$ first maps each memory patch to token width d_h. We then add positional embeddings, apply the same attention aggregator (including attention

AdaLN) used for the input tokens, and apply the same predictor AdaLN used to form the query. We write $\mathcal{S}^{\mathrm{attn}}_{\ell,m,t}(Az^{(r)})$ for this memory attention context. With H predictor heads, the full predictor is

$$\begin{aligned}
q_{\ell,h,t}(s) &= W^Q_{\ell,h}\,\mathrm{AdaLN}_{\ell,t}(s) + b^Q_{\ell,h},\\
k^{(r)}_{\ell,h,m,t} &= W^K_{\ell,h}\,\mathrm{AdaLN}_{\ell,t}\Big(\mathcal{S}^{\mathrm{attn}}_{\ell,m,t}(Az^{(r)})\Big) + b^K_{\ell,h},\\
v^{(r)}_{\ell,h,m} &= W^V_{\ell,h} z^{(r)}_m + b^V_{\ell,h},\\
g^{\mathrm{lpld}}_{\ell,m,t}(s) &= \alpha_{\ell,t} \odot \left[\sum_{h=1}^{H} W^O_{\ell,h}\left(\sum_{r=1}^{R} \mathrm{sm}_r\left\{\frac{q_{\ell,h,t}(s)^\top k^{(j)}_{\ell,h,m,t}}{\sqrt{d_h/H}}\right\}_{j\in[R]} v^{(r)}_{\ell,h,m}\right) + b^O_\ell\right].
\end{aligned} \tag{25}$$

Here A acts patchwise, $W^Q_{\ell,h}, W^K_{\ell,h} \in \mathbb{R}^{(d_h/H)\times d_h}$, and $W^V_{\ell,h} \in \mathbb{R}^{(d_h/H)\times d'_h}$. When $d'_h = d_h$, we may take $A = I$ without adapter parameters; otherwise, A is learned. The shared spatial bank costs RMd'_h parameters once, plus $d_h d'_h$ if the compatibility map A is learned.

D.4.2 Shared cross-attention (SCA)

SCA uses untied key and value banks $\{k^{(r)}, v^{(r)} \in \mathbb{R}^{d'_h}\}_{r=1}^R$ shared across all layers and patch positions. With H predictor heads, the full predictor is

$$\begin{aligned}
q_{\ell,h,t}(s) &= W^Q_{\ell,h}\,\mathrm{AdaLN}_{\ell,t}(s) + b^Q_{\ell,h},\\
k^{(r)}_{\ell,h} &= W^K_{\ell,h} k^{(r)} + b^K_{\ell,h},\\
v^{(r)}_{\ell,h} &= W^V_{\ell,h} v^{(r)} + b^V_{\ell,h},\\
g^{\mathrm{sca}}_{\ell,t}(s) &= \alpha_{\ell,t} \odot \left[\sum_{h=1}^{H} W^O_{\ell,h}\left(\eta_{\ell,h,t}(s)\sum_{r=1}^{R} \mathrm{sm}_r\left\{\frac{q_{\ell,h,t}(s)^\top k^{(j)}_{\ell,h}}{\sqrt{d_h/H}}\right\}_{j\in[R]} v^{(r)}_{\ell,h}\right) + b^O_\ell\right].
\end{aligned} \tag{26}$$

Here $W^Q_{\ell,h} \in \mathbb{R}^{(d_h/H)\times d_h}$ and $W^K_{\ell,h}, W^V_{\ell,h} \in \mathbb{R}^{(d_h/H)\times d'_h}$. The positive headwise gain is $\eta_{\ell,h,t}(s) = \mathrm{softplus}(a^\top_{\ell,h} q_{\ell,h,t}(s) + b^\eta_{\ell,h})$, with learned $a_{\ell,h} \in \mathbb{R}^{d_h/H}$ and scalar $b^\eta_{\ell,h}$. The shared libraries cost $2Rd'_h$ parameters once, while their projections and the query/output maps remain layer-specific.

D.4.3 Shared MLP (SMLP)

SMLP uses the same shared-library structure and H predictor heads as SCA but replaces normalized softmax retrieval with unnormalized GELU retrieval. With H predictor heads, the full predictor is

$$\begin{aligned}
q_{\ell,h,t}(s) &= W^Q_{\ell,h}\,\mathrm{AdaLN}_{\ell,t}(s) + b^Q_{\ell,h},\\
k^{(r)}_{\ell,h} &= W^K_{\ell,h} k^{(r)},\\
v^{(r)}_{\ell,h} &= W^V_{\ell,h} v^{(r)},\\
g^{\mathrm{smlp}}_{\ell,t}(s) &= \alpha_{\ell,t} \odot \left[\sum_{h=1}^{H} W^O_{\ell,h}\left(\sum_{r=1}^{R} \mathrm{gelu}\left\{\frac{q_{\ell,h,t}(s)^\top k^{(r)}_{\ell,h}}{\sqrt{d_h/H}} + (b^K_{\ell,h})^\top k^{(r)}\right\} v^{(r)}_{\ell,h}\right) + b^O_\ell\right].
\end{aligned} \tag{27}$$

K and V projections are bias-free; the Q and O projections have ordinary affine biases. $(b^K_{\ell,h})^\top$ is an additional learned row of the key projection that generates the GELU bias from the shared key prototype. Note that all predictor heads are evaluated for every token: there is no expert router, softmax normalization, or separate retrieval gain. Both the single-head and multihead forms retain the MLP's unnormalized GELU expansion while sharing the underlying representations across depth. The shared libraries cost $2Rd'_h$ parameters once. Including the layer-specific weight matrices but excluding biases and AdaLN, the cost is $2Rd'_h + 2Ld_h(d_h + d'_h)$, compared with $2LRd_h$ for independent width-R MLPs.

D.5 From a single layer to a residual stack

The construction in Sec. 4 isolates aggregation and prediction in a single layer without residual connections. In this section, we explain its extension to a *residual stack*, a multilayer architecture with residual connections.

Let h_ℓ denote the hidden token sequence entering layer ℓ at diffusion time t. We absorb normalization, AdaLN, and output gates into the attention branch $\mathcal{S}^{\mathrm{attn}}_{\ell,t}$ and the tokenwise predictor $g_{\ell,t}$ (an MLP in DiT). The two residual additions give

$$\begin{aligned} s_\ell &= h_\ell + \mathcal{S}^{\mathrm{attn}}_{\ell,t}(h_\ell), \\ h_{\ell+1} &= s_\ell + g_{\ell,t}(s_\ell) = h_\ell + \underbrace{\mathcal{S}^{\mathrm{attn}}_{\ell,t}(h_\ell) + g_{\ell,t}\big(h_\ell + \mathcal{S}^{\mathrm{attn}}_{\ell,t}(h_\ell)\big)}_{F_{\ell,t}(h_\ell)}. \end{aligned} \tag{28}$$

The aggregator interpretation also applies to s_ℓ: it combines the token's existing state with attention-selected context, so the identity path retains a contribution from the token itself. The predictor then maps this contextualized state to a residual feature update.

We emphasize that (28) motivates interpreting the residual stack as *iterative denoising in feature space*: successive aggregation and prediction updates refine the internal residual stream from which the final clean-image estimate is read out. This view is consistent with interpretations of residual networks as unrolled iterative estimation and feature refinement (Greff et al., 2017; Jastrzebski et al., 2018). This interpretation also motivates consulting a shared memory across depth, providing a reusable bank of learned representations at each refinement step.

However, the clean factorization in a single layer does not necessarily extend to a residual stack. Patch projection and positional embedding are applied once at network entry; later layers operate on already contextualized tokens. Although the shared-channel-operator decomposition of attention still holds layerwise, a token can depend on many original patches, and VO or predictor updates can alter later QK routing. The division of labor between QK and VO is therefore less clear-cut than in a single layer. Similarly, the MLP in one layer can affect attention in subsequent layers, and attention can in turn affect subsequent MLP updates.

E Experimental details

Datasets. Table 2 summarizes the seven datasets used in our real-image experiments. All real images are represented as 64×64 RGB arrays with pixel values scaled to $[-1, 1]$; training is unconditional and does not use class, identity, or attribute labels.

Dataset	Content	Training	Evaluation
CelebA (Liu et al., 2015)	Aligned celebrity photographs	162,770	19,962
FFHQ (Karras et al., 2019)	Photographic faces	60,000	5,000
AAHQ (Liu et al., 2021)	Artistic portraits	15,366	1,921
AFHQ-v2 (Choi et al., 2020; Karras et al., 2021)	Cat, dog, and wild-animal faces	14,335	1,467
iCartoonFace (Zheng et al., 2020)	Cartoon-face recognition crops	388,710	22,496
STL-10 (Coates et al., 2011)	Natural-object images	104,500	8,000
ImageNet64 (Chrabaszcz et al., 2017)	Downsampled natural images	1,281,167	50,000

Table 2: **Datasets used throughout the paper.** Counts refer to prepared images after filtering. Evaluation uses the held-out splits described below; the ImageNet64 evaluation set is its official validation split.

We retain CelebA's official train/test splits and center-crop its images to 178×178 before bilinear resizing. For FFHQ, we retain the official 60,000-image training partition and divide the remaining 10,000 images equally into validation and test sets. For AAHQ, crops from the same source artwork remain in the same split. AFHQ-v2 pools all three animal categories, retains the official evaluation split, and removes exact pixel duplicates. iCartoonFace uses its official train/test splits after filtering

invalid crops and exact pixel duplicates; identities may overlap across splits. STL-10 training combines 100,000 unlabeled images with 4,500 labeled training images, reserving the remaining 500 labeled images for validation and retaining the official test split. The synthetic GMRF experiment instead uses the analytically specified prior in Appendix E.1 and does not train a DiT.

Common training protocol. All real-image DiTs divide each image into $M = 256$ nonoverlapping 4×4 patches, giving a 16×16 patch grid with $d_p = 3 \cdot 4^2 = 48$ coordinates per patch. Following the direct-clean-image formulation in Sec. 2, we use the linear Gaussian interpolation

$$X_t = tX_0 + (1-t)\varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, I), \qquad \operatorname{logit}(t) \sim \mathcal{N}(-0.8, 0.8^2). \tag{29}$$

Thus $a_t = t$, $b_t = 1 - t$, and the normalized observation $Y_t = X_t/t$ has noise level $\sigma_t = (1-t)/t$. Let $f_\theta(X_t, t)$ denote the model's direct clean-image prediction. The implementation's VP velocity objective uses $c(t) = \max(1 - t, 0.05)$ and minimizes

$$\begin{aligned}\mathcal{L}(\theta) &= \frac{1}{Md_p}\mathbb{E}\left\| \frac{f_\theta(X_t,t) - X_t}{c(t)} - \frac{X_0 - X_t}{c(t)} \right\|_2^2 \\ &= \frac{1}{Md_p}\mathbb{E}\left[\frac{\|f_\theta(X_t,t) - X_0\|_2^2}{c(t)^2} \right].\end{aligned} \tag{30}$$

The denominator floor is applied to both predicted and target velocities. Away from the floored region, the target velocity is $X_0 - \varepsilon$. The floor changes the time weighting but retains the conditional mean as the pointwise population optimum. Unless specified otherwise below, we use AdamW with a constant learning rate of 2×10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, zero weight decay, gradient clipping at norm 1, and global batch size 32. We maintain an exponential moving average (EMA) of the weights with decay 0.999 and use EMA weights for all reported measurements and generated samples. The experiment-specific settings below give the model sizes, training durations, seeds, precision, and any departures from these defaults.

E.1 Experiment 1-1: Exact denoising context under synthetic GMRF priors

We conduct experiments to illustrate the concept of local context and examine the prediction of Cor. 2: within a specified noise interval, the radius certifying an upper bound on locality error, which we call the *certified radius*, is nondecreasing as the effective noise level increases.

E.1.1 Synthetic Gaussian MRF prior.

We consider the Gaussian MRF image prior introduced in Thm. 1, for a single-channel image consisting of 16×16 scalar patches of dimension $d_p = 1$. We draw a graph G_0 by sampling 384 distinct unordered pairs uniformly without replacement from the $\binom{256}{2}$ possible pairs, and independently draw each retained edge's connection strength as $\omega_{ij} \sim \mathrm{Uniform}(0.5, 1.5)$. We choose 384 edges to give an average of three neighbors per patch: each undirected edge contributes to the degree of both endpoints, so the mean degree is $2 \times 384/256 = 3$. Then, let $A_{ij} = A_{ji} = \omega_{ij}$ for each sampled edge $\{i, j\}$ and zero otherwise. We define the prior by

$$X_0 \sim \mathcal{N}(0, Q^{-1}), \qquad Q = 0.2I + \operatorname{diag}(A\mathbf{1}) - A. \tag{31}$$

Here $\mathbf{1}$ is the all-ones vector, so $A\mathbf{1}$ contains the row sums of A. We use the weighted graph Laplacian $\operatorname{diag}(A\mathbf{1}) - A$ so that the prior penalizes differences between connected patches, with larger edge weights encouraging stronger agreement. The positive diagonal shift ensures that Q is positive definite even if the graph is disconnected. We evaluate 41 logarithmically spaced noise levels from 10^{-2} to 10^2, together with zero and the corollary's endpoint σ_{abs}, giving 43 noise levels in total.

E.1.2 Exact and certified radii for absolute locality error

Since we know the population prior and its graph, we can compute the global population denoiser and the exact local denoiser for each graph-radius context $B_r(m) = \{l : d_{G_0}(m, l) \le r\}$. Here d_{G_0} is the unweighted shortest-path distance; vertices in different components have infinite distance and zero cross-component influence. This allows us to compare the minimum radius achieving a given

locality error with the radius certifying its upper bound in Cor. 2. Specifically, observe $Y_\sigma = X_0 + \sigma\varepsilon$ with independent $\varepsilon \sim \mathcal{N}(0, I)$, and write $J_\sigma = (I + \sigma^2 Q)^{-1}$ for the population denoiser's Jacobian. The global denoiser and the local denoiser on $B_r(m)$ are

$$D^*_{m,\sigma}(y) = (J_\sigma y)_m, \tag{32}$$

$$D^{*,r}_{m,\sigma}(y) = (Q^{-1})_{m,B_r(m)} \big[(Q^{-1})_{B_r(m),B_r(m)} + \sigma^2 I\big]^{-1} y_{B_r(m)}. \tag{33}$$

Gaussian conditioning gives the locality error in closed form:

$$\begin{aligned}&\mathbb{E}\big|D^*_{m,\sigma}(Y_\sigma) - D^{*,r}_{m,\sigma}(Y_\sigma)\big|^2 \\ &\quad = (J_\sigma Q^{-1})_{mm} - (Q^{-1})_{m,B_r(m)} \big[(Q^{-1})_{B_r(m),B_r(m)} + \sigma^2 I\big]^{-1} (Q^{-1})_{B_r(m),m}.\end{aligned} \tag{34}$$

Based on this expression, we compute the *exact radius* for an absolute tolerance $\epsilon > 0$ as

$$r_{\mathrm{ext}}(m, \sigma; \epsilon) := \min\left\{ r \in \mathbb{Z}_{\geq 0} : \mathbb{E}\big|D^*_{m,\sigma}(Y_\sigma) - D^{*,r}_{m,\sigma}(Y_\sigma)\big|^2 \leq \epsilon \right\}. \tag{35}$$

Corollary 2 also gives the certified radius. For this calculation we use $q_- = 0.2$ and $q_+ = 0.2 + 2\max_i \sum_j A_{ij} \approx 21.420774$. The former is the exact minimum eigenvalue of Q, while the latter is a conservative upper bound on its maximum eigenvalue; the theorem's bound remains valid with this spectral enclosure. Using χ_t and ρ_t as defined in Appendix C.1, evaluated at $\sigma_t = \sigma$, the *certified radius* is

$$r_{\mathrm{crt}}(\sigma; \epsilon) := \min\left\{ r \in \mathbb{Z}_{\geq 0} : \frac{(1 - 1/\sqrt{\chi_t})^2}{q_-(1 + \sigma^2 q_-)} \rho_t^{2r} \leq \epsilon \right\}. \tag{36}$$

At zero noise, both radii are zero. The certified radius is common for all target patches m. We use tolerances $\epsilon = 10^{-3}$ in the main figure.

E.1.3 Sensitivity radius as a measure of statistical dependence

Along with these exact and certified radii, we define a sensitivity radius based on the input–output Jacobian, which measures how perturbing each observed patch affects the predicted target patch. For this Gaussian prior, $(J_\sigma)_{ml} = \partial D^*_{m,\sigma}(y)/\partial y_l$, with $J_\sigma = (I + \sigma^2 Q)^{-1}$. We square its entries to measure sensitivity energy and normalize over input patches as $(J_\sigma)^2_{ml} / [\sum_j (J_\sigma)^2_{mj}]$. This normalized derivative energy is the fraction of the target m patch's total squared sensitivity attributable to input patch l. It measures relative influence, so a common reduction in the magnitude of all derivatives does not by itself change the normalized map. Using this normalization, we define the *q-sensitivity radius*:

$$r_{\mathrm{sns}}(m, \sigma; q) := \min\left\{ r \in \mathbb{Z}_{\geq 0} : \sum_{l \in B_r(m)} (J_\sigma)^2_{ml} \Big/ \left[\sum_j (J_\sigma)^2_{mj}\right] \geq q \right\}. \tag{37}$$

The self-patch is included, and we evaluate $q \in \{0.90, 0.95\}$. Unlike the exact and certified radii, this radius concerns relative derivative energy rather than an absolute locality-error tolerance, so the three radii need not agree. For any prior with finite second moments, the population denoiser satisfies the posterior-covariance identity (Dytso et al., 2021)

$$\frac{\partial D^*_{m,t}(y)}{\partial y_l} = \sigma_t^{-2} \operatorname{Cov}(X_{0,m}, X_{0,l} \mid Y_t = y, t), \qquad \sigma_t > 0. \tag{38}$$

Thus, its Jacobian reflects the dependence between clean patches that remains after observing the noisy image. For the Gaussian prior considered here, this Jacobian is independent of y.

E.1.4 Interpretation of Fig. 2

Fig. 2(a) Local context need not be spatially contiguous. The first panel shows the radius-3 local contexts of the two selected patches, containing 12 and 50 nodes, respectively, including the targets. Because these sets are defined by graph distance, they need not be square or spatially contiguous: a patch that is distant on the image grid can still be nearby in G_0. Each ball is nevertheless connected in the graph. Thus, locality is relative to the dependence structure of the prior, rather than necessarily to proximity on the image grid.

Fig. 2(b) The certified radius is nondecreasing over a bounded noise interval. The second panel compares the exact r_{ext} and certified radii r_{crt} at the absolute error tolerance $\epsilon = 10^{-3}$. At low noise, the target patch's own observation is accurate, so little additional context is needed. As noise increases, observations of correlated patches become more useful, and both radii are nondecreasing over the specified interval in this realization. r_{crt} is conservative: its maximum over the sweep is 16 hops, compared with 4 hops for each exact radius. The dotted line marks $\sigma_{\mathrm{abs}} \approx 1.128962$, the endpoint specified in Cor. 2. The corollary guarantees monotonicity of the certified radius on $[0, \sigma_{\mathrm{abs}}]$; the exact-radius behavior is an empirical observation in this example. Beyond this interval, the radii eventually decrease as the observations provide progressively less information about the clean signal. Indeed, $J_\sigma \sim \sigma^{-2} Q^{-1}$ as $\sigma \to \infty$, so both the global and local posterior means approach the zero prior mean and their absolute difference shrinks. Note that this does not mean that reconstruction improves: locality error measures the discrepancy between the two denoisers, not their error relative to the clean image.

Fig. 2(c) Relative sensitivity reveals how the denoiser distributes influence across patches. The third panel shows that the sensitivity radius r_{sns} can remain broad even when the absolute locality error becomes small. As noise increases from low levels, a larger graph neighborhood is needed to capture the same fraction of the total derivative energy, consistent with the initial growth of r_{ext} and r_{crt}. At extreme noise, however, the absolute response to every observation weakens, while its relative contribution approaches a constant:

$$\frac{(J_\sigma)_{ml}^2}{\sum_j (J_\sigma)_{mj}^2} \longrightarrow \frac{((Q^{-1})_{ml})^2}{\sum_j ((Q^{-1})_{mj})^2} \qquad \text{as } \sigma \to \infty. \tag{39}$$

At extreme noise, the posterior covariance approaches the prior covariance Q^{-1}, while the factor σ^{-2} in the Jacobian makes its absolute entries vanish. This common scale cancels under normalization, leaving a pattern determined by the prior's covariances, which can extend to distant patches. These results show why a broad sensitivity radius can coexist with a small absolute locality error: the radius describes the distribution of influence, while the error also depends on its absolute strength. Note that in this synthetic setting, we do not claim that the sensitivity radius must increase monotonically with noise, or that observations outside the corresponding neighborhood can be removed without affecting denoising accuracy.

This interpretation provides a link to denoisers trained on real-image datasets: their input–output Jacobians also describe responses to infinitesimal perturbations, so the sensitivity radius r_{sns} can be defined in the same way to characterize local functional context even when the population prior and exact local conditional means are unknown. Furthermore, due to (38), to the extent that learned denoisers approximate the population denoisers, their Jacobian-based sensitivity provides a measure of statistical dependence relevant to denoising.

E.2 Experiment 1-2: Denoising context in DiTs trained on real-image datasets

We ask whether the denoising-relevant statistical dependence in image datasets exhibits a similar pattern to the exponentially local GMRF model studied in Appendix E.1. We use trained DiTs as estimators of the population denoiser and its Jacobian; through the posterior-covariance identity (38), their sensitivity fields provide an empirical view of this dependence. We use a DiT with approximately 20M parameters, comprising $L = 10$ transformer layers, token dimension $d_h = 320$, $\tilde{H} = 10$ attention heads, and MLP hidden width 1280. We denote its clean-patch prediction by $D^{\mathrm{dit}}_{m,t}$ and use the common patch representation and training protocol at the start of Appendix E. Following the normalized-observation convention in Sec. 2, $D^{\mathrm{dit}}_{m,t}(y)$ denotes the prediction obtained by feeding ty and t to the trained network; thus, all input derivatives below are with respect to y.

Experiment-specific training. We train models on CelebA, iCartoonFace, ImageNet64, and STL-10 using the datasets and splits described above. We analyze checkpoints at epochs 400, 400, 800, and 800, respectively. The non-ImageNet runs use batch size 32 and FP32 on single GPUs, mostly RTX 2080 Ti GPUs; the ImageNet checkpoint uses batch size 256 and BF16 on an H200.

E.2.1 SENSITIVITY CONTEXT SET BASED ON INPUT-OUTPUT JACOBIAN

Our primary measurement is the Jacobian of output patch m with respect to input patch l, extending the scalar-patch sensitivity in Appendix E.1. For real-image priors, the population derivative $\partial D^*_{m,t}(y)/\partial y_l$ is unavailable, so we estimate it using the trained denoiser's Jacobian,

$$(J_t(y))_{ml} := \frac{\partial D^{\mathrm{dit}}_{m,t}(y)}{\partial y_l} \in \mathbb{R}^{d_p \times d_p}. \tag{40}$$

Since each patch contains multiple spatial and color coordinates, we measure its sensitivity by the squared Frobenius norm of this block, which sums the squared derivatives over all input–output coordinate pairs. The squared Frobenius norm extends the squared scalar Jacobian entries used in Appendix E.1 to vector-valued patches. This also follows the use of input-gradient sensitivity to study denoiser locality in Niedoba et al. (2025, Sec. 3.1).

For each dataset, we evaluate 1,024 held-out images at 12 noise levels from $\sigma_t = 0.02$ to 49. In practice, we estimate the squared Frobenius norm using vector–Jacobian products with four independent Rademacher output probes per target patch, following Hutchinson-type Monte Carlo estimation (Avron & Toledo, 2011). We divide by d_p to obtain sensitivity per output coordinate, average over held-out clean images and Gaussian corruption noise, and then normalize over input patches to define the *normalized sensitivity weight*:

$$w_{ml}(t) := \frac{d_p^{-1}\mathbb{E}_{X_0,\varepsilon}\|(J_t(Y_t))_{ml}\|_F^2}{\sum_{j=1}^{M} d_p^{-1}\mathbb{E}_{X_0,\varepsilon}\|(J_t(Y_t))_{mj}\|_F^2}. \tag{41}$$

Thus, $w_{ml}(t)$ is the fraction of patch m's expected squared sensitivity attributable to input patch l, with $\sum_l w_{ml}(t) = 1$ whenever the total sensitivity is positive.

For real images, the prior's dependence graph is unknown so we cannot compute the exact radius of a local context as in the synthetic case. Instead, we describe local context in terms of spatial distance and patch count. Specifically, we use the Chebyshev grid distance, a patch hop, defined as $d_{\mathrm{grid}}(m,l) = \max\{|i_m - i_l|, |j_m - j_l|\}$, where i_m, j_m denote the row and column indices of patch m, and define $B_r(m) = \{l : d_{\mathrm{grid}}(m,l) \le r\}$. These neighborhoods are square regions clipped to the image boundary, with radius measured in patch hops. Using the normalized derivative energy in (41), we define the *sensitivity radius* by

$$r_{\mathrm{sns}}(m,t;q) := \min\left\{ r \in \mathbb{Z}_{\ge 0} : \sum_{l \in B_r(m)} w_{ml}(t) \ge q \right\}. \tag{42}$$

The radius is the smallest centered Chebyshev ball containing at least a fraction q of the normalized derivative energy. This ball measures the spatial extent of the context, but the functionally relevant local context set need not be ball-shaped. To allow arbitrary shapes, we also define a context set containing the fewest patches that together account for at least a fraction q of the normalized derivative energy and the *sensitivity count* as

$$\mathcal{B}_{m,t}(q) \in \operatorname*{arg\,min}_{B \subseteq [M]} \left\{ |B| : \sum_{l \in B} w_{ml}(t) \ge q \right\}, \qquad K_q(m,t) := |\mathcal{B}_{m,t}(q)|. \tag{43}$$

We construct this set by adding patches in descending order of $w_{ml}(t)$ until their cumulative energy reaches q, breaking ties by patch index. The selected patches may be scattered or disconnected; the target patch is eligible for selection. For the population curves, we compute each target's radius and count from the image-averaged energy in (41), then average these quantities over all 256 target patches. For the displayed sensitivity heatmaps, we apply the same construction to the normalized derivative energy of the individual noisy image.

Together, these quantities distinguish spatially broad context from sparse influence carried by a few distant patches. The posterior-covariance identity (38) connects these measurements to statistical dependence remaining after observing the noisy image. Under the Jacobian approximation described above, these sensitivity-based contexts estimate the patches carrying most of the expected squared posterior cross-covariance with the target. They thus characterize the dataset's denoising-relevant dependence through a learned predictor.

E.2.2 Attention context set based on attention mass

Additionally, we average the normalized attention matrices computed from the residual-stream features of $D^{\mathrm{dit}}_{m,t}$ over heads and layers, and use the resulting attention mass in place of $w_{ml}(t)$. Specifically, let $A^{(\ell,h)}_{ml,t}(y)$ denote the softmax attention weight from query patch m to source patch l in head h of layer ℓ, evaluated when the network receives ty and time t. These are the spatial weights in (21), computed from each layer's features after layer normalization and AdaLN modulation, as described in Appendix D.5. Averaging over the same held-out images and corruption noise gives the *mean attention weight*:

$$w^{\mathrm{attn}}_{ml}(t) := \mathbb{E}_{X_0,\varepsilon}\left[\frac{1}{L\tilde{H}}\sum_{\ell=1}^{L}\sum_{h=1}^{\tilde{H}} A^{(\ell,h)}_{ml,t}(Y_t)\right]. \tag{44}$$

Each attention row sums to one over source patches, so $\sum_l w^{\mathrm{attn}}_{ml}(t) = 1$ without further normalization. We average the softmax weights themselves, without squaring them or weighting them by residual gates or value/output projections; this measures how attention distributes mass across token positions along the residual stream. Note that both $w^{\mathrm{attn}}_{ml}(t)$ and $w_{ml}(t)$ assign normalized weights to input patches l for a given target patch m, but they measure different quantities: the former describes how attention distributes mass within the network, whereas the latter measures the relative squared sensitivity of the final denoised prediction to each input patch. In this sense, although both weights lie in $[0,1]$ and sum to one, equal values do not imply equal influence.

Using the same neighborhoods and coverage threshold as for the sensitivity radius, we define the *attention radius* by

$$r_{\mathrm{attn}}(m,t;q) := \min\left\{ r \in \mathbb{Z}_{\geq 0} : \sum_{l\in B_r(m)} w^{\mathrm{attn}}_{ml}(t) \geq q \right\}. \tag{45}$$

Mirroring (43), we additionally define the attention context set and *attention count* as

$$\mathcal{B}^{\mathrm{attn}}_{m,t}(q) \in \operatorname*{arg\,min}_{B\subseteq[M]} \left\{ |B| : \sum_{l\in B} w^{\mathrm{attn}}_{ml}(t) \geq q \right\}, \qquad K^{\mathrm{attn}}_{q}(m,t) := |\mathcal{B}^{\mathrm{attn}}_{m,t}(q)|. \tag{46}$$

The radius measures the smallest centered square capturing a fraction q of the mean attention weight, while the count measures the fewest patches carrying that fraction, allowing arbitrary spatial arrangements. We obtain the attention context set by selecting patches in descending order of $w^{\mathrm{attn}}_{ml}(t)$, with the same tie-breaking rule as in (43). These quantities are computed after averaging the attention maps over heads, layers, and images; the population curves then average the resulting radii or counts over all 256 target patches. For an individual-image heatmap, we average only over heads and layers before selecting its context set.

E.2.3 Interpretation of Fig. 3

Estimated dependence fields resemble the exponentially local GMRF setting. Figure 3 reveals a pattern of denoising-relevant statistical dependence similar to that of the synthetic GMRF prior. Sensitivity is concentrated at lower noise, while its radius generally increases as noise grows, echoing the broadening of relative sensitivity in Fig. 2. Through the posterior-covariance interpretation in (38), this suggests that predicting a patch relies on a broader set of statistical dependencies when the observation is noisier. Figures 7 and 8 corroborate this overall broadening across additional datasets and at both 90% and 95% coverage curves. The analytic GMRF setting thus provides a useful model for interpreting the estimated dependence fields of real-image datasets, beyond its role as a tractable synthetic example.

The context set need not be spatially contiguous. The estimated dependence fields in the heatmaps show that patches carrying most of the sensitivity can form irregular or disconnected regions rather than a centered square. At lower noise, influence is concentrated near the target and selected nearby structures; at higher noise, the displayed context sets spread over a larger part of the image. The radius and patch count therefore describe complementary aspects of context: its spatial extent and the number of patches carrying a specified fraction of influence.

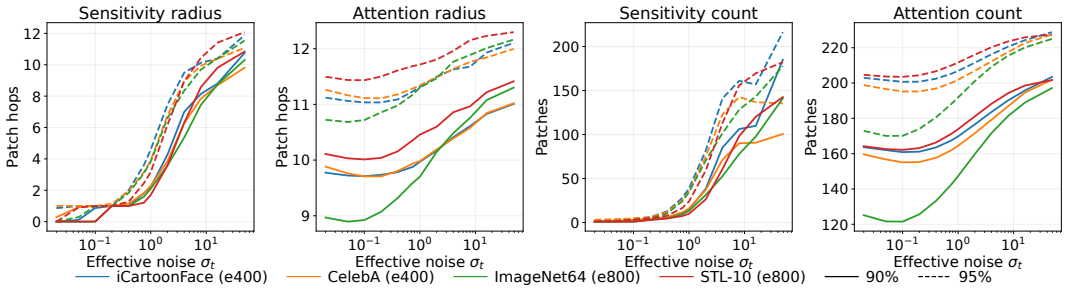

Figure 7: Sensitivity and attention radii and patch counts on four datasets. Solid/dashed curves show 90%/95% coverage, computed from population maps over 1,024 held-out images and averaged over all 256 target patches.

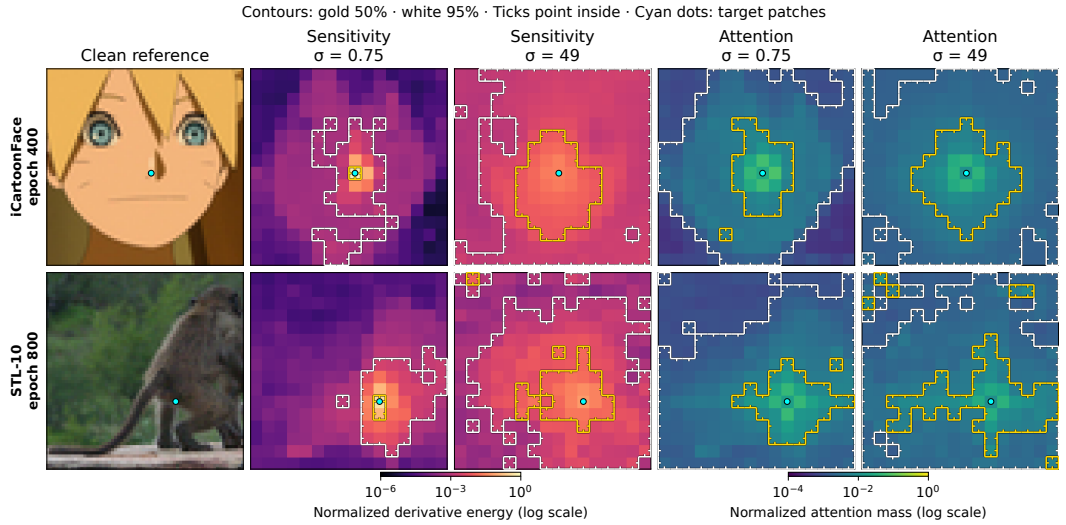

Figure 8: Additional context maps for iCartoonFace and STL-10 at $\sigma_t = 0.75, 49$, complementing the CelebA and ImageNet examples in Fig. 3. Colors are logarithmic; gold/white contours mark 50%/95% coverage, ticks point inward, and cyan dots mark target patches.

Attention exhibits a related pattern of context formation. Head- and layer-averaged attention also tends to spread over a broader region at higher noise, although its mass is already more diffuse than sensitivity at low noise. Attention weights and input–output Jacobians are different quantities: the former are softmax-normalized logits between intermediate tokens, whereas the latter measure how the final denoised prediction responds to the original input. There is therefore no requirement that their spatial patterns agree. Their observed agreement in overall broadening is consistent with attention adapting its aggregation to the dataset's denoising-relevant dependence structure and contributing to the formation of noise-dependent local context, as suggested by the aggregator interpretation in Sec. 4.1 and Appendix D.5. The additional radius/count curves and heatmaps in Figs. 7 and 8 support this interpretation. Note that the comparison concerns a shared qualitative trend rather than equality of context sets: 95% attention mass and 95% squared derivative energy describe different distributions.

E.3 Experiment 2: Replacing MLPs with Shared-Memory Predictors

E.3.1 Motivation

Our aggregator–predictor framework decomposes the local conditional mean into a local aggregator and a patch predictor (Def. 2). Sections 4.1 and 4.2 associate these operations with attention and the DiT MLP, respectively. This interpretation motivates replacing the MLP with more interpretable predictors based on retrieval from a reusable bank of learned representations, as introduced in Sec. 4.2,

with the aim of attaining comparable denoising and generative performance at a similar trainable-parameter budget. If the proposed alternatives were to fail to do so, this would indicate that their representation or training omits useful properties of an ordinary MLP for denoising. Conversely, a successful replacement provides evidence that the framework supports a useful, explicitly structured alternative. Additionally, this comparison lets us examine the effects of three design choices on empirical performance: patch-specific versus patch-shared storage, softmax cross-attention versus unnormalized GELU retrieval, and a base memory shared across layers versus independent layerwise memory. We take a baseline DiT and replace only its patch predictor with an alternative predictor, leaving everything else fixed; the following section provides the details.

E.3.2 SETTINGS

Dataset and common backbone. We use the CelebA splits, preprocessing, and common training protocol described at the start of Appendix E, without horizontal-flip augmentation. All models use token width $d_h = 256$, depth $L = 8$, and $\tilde{H} = 8$ spatial-attention heads. The patch embedding, spatial-attention architecture, AdaLN conditioning, residual structure, and image-output head are common across families. Each model is trained jointly from initialization; the common backbone specification does not mean that attention weights are frozen or shared between runs.

Predictor configurations and training. We use the implementations in Appendix D.4, with the residual composition in Appendix D.5. Table 3 lists all 21 configurations. For each shared-memory family, we compare $H = 1$ and $H = 8$ predictor heads at the same stored width, with projected width d_h/H per head. Thus, the total projected query, key, and value width stays at 256 rather than growing eightfold with the number of heads. We vary stored width d'_h and choose MLP hidden widths to obtain three approximate parameter budgets around 8.6–8.9M, 10.2M, and 13.0–13.4M parameters. The table reports exact counts, including the common backbone and all predictor parameters; matching is approximate, rather than exact equality or equal computation.

For training, we use the common objective in Eq. (30) and train all configurations to 400 epochs, with 5,086 optimizer updates per recorded epoch and 2,034,400 updates in total. The global batch size is 32, and the default learning rate is constant except for the LPLD recovery intervention below. All runs use training seed 0.

Predictor	R	d'_h	Parameters, $H = 1$	Parameters, $H = 8$
LPLD	32	128	8,935,472	8,935,472
LPLD	32	256	10,213,424	10,213,424
LPLD	32	512	12,965,936	12,965,936
SCA	1024	256	8,642,616	8,642,672
SCA	1024	512	10,215,480	10,215,536
SCA	1024	1024	13,361,208	13,361,264
SMLP	1024	256	8,638,512	8,652,848
SMLP	1024	512	10,213,424	10,242,096
SMLP	1024	1024	13,363,248	13,420,592
MLP	640	256	8,639,536	—
MLP	1024	256	10,215,472	—
MLP	1792	256	13,367,344	—

Table 3: **Complete patch-predictor grid.** Each numeric entry in the last two columns specifies one trained configuration. All models use $d_h = 256$, $L = 8$, and $\tilde{H} = 8$; predictor head width is 256 for $H = 1$ and 32 for $H = 8$. R counts image-shaped memories for LPLD, shared prototype pairs for SCA/SMLP, and hidden units for MLP.

LPLD computation and noise groups. LPLD's key path applies spatial attention to each of the R full memory grids at every layer and noise level. This is substantially more expensive than projecting a patch-shared prototype bank: distinct sampled times require distinct conditioned memory computations. We therefore draw four independent times per global batch and share each time among eight images, retaining independent Gaussian noise for each image. This allows the memory-key com-

putation to be reused within a time group, while leaving per-image marginal objective in Eq. (30) unchanged.

During training, the original $H = 8, d'_h = 256$ LPLD run developed severe generation artifacts and degraded denoising during training. We exclude that run from the displayed comparisons and use a separate recovery that resumes its epoch-225 checkpoint, including optimizer and EMA state, at learning rate 10^{-4}. The recovery reaches cumulative epoch 400 after 175 additional epochs and has FID 29.197 at that endpoint. Only completed post-restart measurements are drawn for its trajectory; they are not spliced onto the original curve.

Evaluation. All measurements use EMA weights and the same 19,962 held-out test images. We evaluate the velocity objective above and the raw, unclipped clean-image MSE across 32 logarithmically spaced normalized noise levels $\sigma \in [0.05, 4]$. The denoising score is the trapezoidal integral of this MSE with respect to $\log \sigma$, divided by $\log(4/0.05)$, so it is an average error over the specified noise range. At each noise level we use $t = 1/(1 + \sigma)$ to map the normalized observation to the model's input convention. FID-50k compares 50,000 generated samples with the common real-test feature statistics. Generation uses 50 uniform time intervals from noise to data, with Heun updates and an Euler update on the last interval, and the same denominator floor of 0.05. We evaluate the standard training trajectories every 25 epochs. Figure 4 uses epoch 400 for every displayed metric and model in the current snapshot, rather than selecting each model's best intermediate FID.

E.3.3 Results and Interpretation

Figure 9 shows that all reported families learn substantial denoising improvements, but their attained errors and generation trajectories differ. Held-out losses generally decrease smoothly, whereas FID is less monotone. The following comparisons use the common epoch-400 snapshot in Fig. 4.

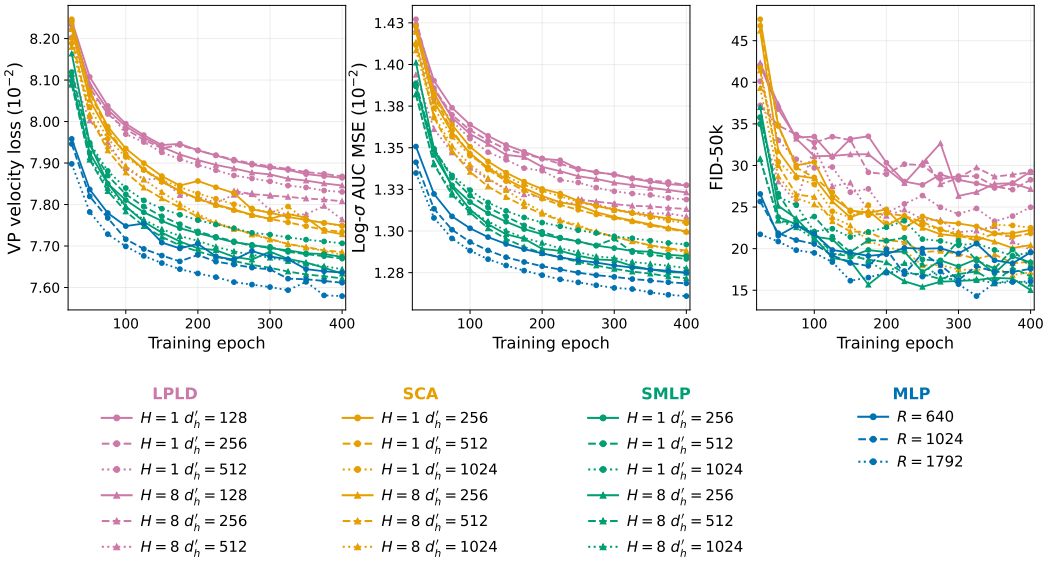

Figure 9: **Training trajectories for the patch-predictor comparison.** Held-out velocity loss, denoising log-σ AUC MSE, and FID-50k on CelebA64. Colors identify predictor families; circles and triangles indicate $H = 1$ and $H = 8$. Solid, dashed, and dotted curves denote increasing d'_h within each shared-memory family, or increasing hidden width R for MLP.

(1) Different patches can retrieve from a shared patch memory. The patch-shared SCA and SMLP banks are more competitive than LPLD's position-specific image-shaped memory in the corresponding parameter-budget groups. At the largest budget with $H = 8$, LPLD has denoising AUC 0.013087 and FID 19.848, compared with 0.012873 and 17.145 for SCA, and 0.012777 and 15.767 for SMLP. Thus, allocating a separate stored value vector to every patch position is not necessary for competitive denoising in this setting. Sharing prototypes across positions permits a much larger

bank—1,024 entries rather than 32—at a similar total parameter count. Overall, the weaker LPLD results show that the most literal learned-image retrieval construction is not automatically the easiest or most effective mechanism for patch predictor.

(2) GELU retrieval is more effective here than softmax retrieval. At every tested (H, d'_h) pair, SMLP has lower held-out velocity loss, denoising AUC, and FID than SCA. For example, at $H = 8, d'_h = 512$, SCA and SMLP have FIDs 18.413 and 16.652, respectively, with denoising AUCs 0.012882 and 0.012716. Although SCA already includes the positive, query-dependent headwise gain $\eta_{\ell,h,t}(s)$ specified in Appendix D.4, its within-head softmax coefficients remain nonnegative and coupled by normalization. In contrast, GELU allows unnormalized, potentially signed coefficients, providing a more flexible way to combine the shared values. The observed advantage is consistent with this flexibility being useful during training.

(3) Multiple predictor heads are usually beneficial. Holding the stored width fixed, $H = 8$ improves both held-out denoising measures over $H = 1$ at every tested width in all three shared-memory families. For SCA and SMLP, it also improves FID at every width. For example, at $d'_h = 1024$, increasing H from one to eight reduces FID from 22.508 to 17.145 for SCA and from 19.556 to 15.767 for SMLP. LPLD FID improves from 29.269 to 27.167 at $d'_h = 128$ and from 24.976 to 19.848 at $d'_h = 512$. This suggests that multiple projections of the shared bank can be useful while keeping the total projected width fixed.

(4) The base memory can be shared across layers. The central comparison is between SMLP, whose base key–value bank is reused at every layer, and the ordinary MLP, whose vectors are independently stored in each layer. For $H = 8$ SMLP at $d'_h \in \{256, 512, 1024\}$, epoch-400 FIDs are 15.017, 16.652, and 15.767, compared with 19.550, 17.574, and 16.060 for the respective MLP controls. At the smallest budget, their denoising AUCs are 0.0127499 and 0.0127496, and their velocity losses are 0.076336 and 0.076360. At the middle and largest budgets, SMLP has approximately 0.24% and 1.33% higher denoising AUC, respectively, despite its lower FID. Thus the shared-memory predictor is competitive in both denoising and generation, with a small denoising tradeoff at the larger budgets. We emphasize that the result is a constructive demonstration that independently stored memory at every layer is not required for this level of performance.

Scope. We close by noting the scope of this experiment. It is a controlled study of unconditional face generation at 64×64 resolution, with modest model sizes and one training seed per configuration. These limits constrain how broadly the findings of our experiment can be applied. We therefore do not claim that all DiT MLPs should share memory. Instead, the results support considering an explicit shared memory architecture when a common representation bank would be useful for inspection or future intervention. Our experiment shows that this architecture is feasible at least in the setting studied here.

E.4 Experiment 3: Component compatibility in cross-dataset adaptation

Motivation and design. The aggregator–predictor framework separates forming a patch's local context from predicting its clean value using learned representations. Section 4.1 associates the first operation with attention and Sec. 4.2 associates the second with the MLP. Experiment 1 finds that DiT sensitivity and attention maps exhibit related noise-dependent broadening on real images; Experiment 2 shows that explicit retrieval from a shared memory supports competitive denoising and generation quality in a controlled CelebA64 study. We now ask whether component compatibility across image domains is consistent with the proposed attention and MLP roles. This question also matters when target data or adaptation compute are limited: fine-tuning methods must decide which pretrained parameters to update, and the best choice can depend on the source and target distributions (Kumari et al., 2023; Choi et al., 2024).

The attention decomposition in (7) gives a more specific hypothesis. Within an isolated branch, QK controls spatial weights and VO transforms and transports features; the MLP admits a key–value representation for tokenwise updates. This motivates the hypothesis that source QK may remain compatible with adapted VO and MLP parameters when source and target have similar spatial context, with larger shifts associated with reduced compatibility. Source-QK compatibility may also be directional: a broadly trained source may support contextual relations needed by a narrower target,

whereas the reverse transfer need not hold. We probe these hypotheses by adapting QK, VO, and MLP jointly to each target, then replacing one or more adapted groups with their source weights without further optimization. The eight resulting hybrids show how much each subset of jointly learned changes matters for reproducing the jointly adapted model.

Training and endpoint selection. We use the 20M-class DiT of Appendix E.2 on 64×64 RGB images: $L = 10$, $d_h = 320$, $\tilde{H} = 10$, and MLP hidden width 1280. The twelve directions are CelebA→AAHQ/FFHQ/AFHQ-v2/STL-10, STL-10→AAHQ/FFHQ/AFHQ-v2/CelebA, and ImageNet64→AAHQ/FFHQ/AFHQ-v2/CelebA. Source training uses seed 0; adaptation starts from EMA checkpoints at source epochs 400, 800, and 800, respectively, and is repeated with two adaptation seeds. Only the attention QK and VO projections along with MLP block weights are updated jointly. Patch, position, and time embeddings, AdaLN, the output head, and other parameters stay fixed. Freezing AdaLN keeps its learned time conditioning while holding the denoising schedule constant; it assumes that this aspect of time dependence transfers across datasets.

Training uses the common VP objective, global batch 32, AdamW with learning rate 2×10^{-4}, and a fresh optimizer for adaptation. For source and target training-set sizes $N_{\mathrm{src}}, N_{\mathrm{tgt}}$, one reference epoch is $s_{\mathrm{ref}} = \min(\lfloor N_{\mathrm{src}}/32 \rfloor, \lfloor N_{\mathrm{tgt}}/32 \rfloor)$ updates; adaptation stops by 200 reference epochs, with checkpoints every 25. For each direction, we select the checkpoint minimizing the joint model's two-seed mean FID-50k on the monitored target split, with earlier epochs breaking ties, and use that same reference epoch for both seeds (Table 4). Each eight-state hybrid cube uses the selected checkpoints, EMA weights, and the same 1,024 Gaussian initial latents per state, sampled in FP32 with 50-step VP Heun. Note that since the three architectural components are adapted jointly, these eight-state swaps measure compatibility among jointly adapted components; they do not measure what independently fine-tuned restricted models could learn.

Scope. Component transplantation diagnoses compatibility among parameter groups obtained by joint adaptation; it neither determines their performance when optimized in isolation nor uniquely assigns computation to components in a multilayer residual network. Separate restricted-training comparisons are reported in Appendix E.4.5. Spatial MMD, EA, and SA are descriptive proxies for dataset geometry and endpoint agreement, not direct measurements of denoising context. The twelve transfer directions share datasets and pretrained sources, and the component contrasts share cube terms; their correlations are descriptive rather than independent-sample estimates of a population relationship.

Transfer	Epoch	FID	Transfer	Epoch	FID
CelebA→AAHQ	75	40.767	STL-10→AAHQ	150	40.592
CelebA→FFHQ	100	25.252	STL-10→FFHQ	100	28.231
CelebA→AFHQ-v2	200	44.821	STL-10→AFHQ-v2	200	49.119
CelebA→STL-10	200	68.050	STL-10→CelebA	200	17.262
ImageNet64→AAHQ	150	34.493	ImageNet64→FFHQ	100	22.640
ImageNet64→AFHQ-v2	200	38.840	ImageNet64→CelebA	100	14.620

Table 4: Selected joint-adaptation endpoints. Epochs use the smaller source/target training set as reference; FID is the two-seed mean FID-50k used for checkpoint selection. STL-10 and ImageNet64 source checkpoints are both from epoch 800.

E.4.1 Spatial MMD for dataset similarity

To describe spatial similarity between unpaired real-image datasets, we compare distributions of whole-image patch self-similarity descriptors using maximum mean discrepancy (MMD). For an individual clean image x_0, let $u_m(x_0)$ be its final DINOv2 ViT-B/14 feature at patch position m after square resize to 224×224, yielding a 16×16 patch grid (Oquab et al., 2024). For each feature channel, subtract its mean over the 256 patch positions and divide by its spatial sample standard deviation plus 10^{-6}; then normalize each patch vector to unit length, obtaining $\widehat{u}_m(x_0)$. For patch positions $m, l \in [256]$, we form $K_{x_0}(m,l) = \widehat{u}_m(x_0)^\top \widehat{u}_l(x_0)$ and let $v(x_0) \in \mathbb{R}^{32640}$ collect the upper off-diagonal entries of K_{x_0} in a fixed raster order. Thus each descriptor retains *where* pairs of patches have similar features. Patch self-similarity has been used to describe image structure (Tumanyan et al., 2022), and spatially normalized patch features have been studied in diffusion

representation alignment (Singh et al., 2026); applying MMD to these descriptors for cross-dataset comparison is our construction.

For descriptors v, v', we use the Gaussian kernel $\kappa^{\mathrm{sp}}(v, v') = \exp[-\|v - v'\|_2^2/(2 \cdot 32640\,\tau^2)]$. We fix τ^2 to the median of $\|v - v'\|_2^2/32640$ over all distinct pairs in a balanced calibration bank of 128 held-out images from each of six datasets, separately for normalized and raw patch descriptors. Let $A = \{a_i\}_{i=1}^{n_A}$ and $B = \{b_j\}_{j=1}^{n_B}$ be independent evaluation banks of real-image descriptors. For distinct datasets, $n_A = n_B = 1000$; for a diagonal cell, the banks are disjoint halves of one dataset and $n_A = n_B = 500$. Conditional on the calibrated kernel bandwidth, the following is an unbiased estimator of the population squared MMD between the corresponding distributions of $v(X_0)$ (Gretton et al., 2012):

$$\widehat{\mathrm{MMD}}^2_{\mathrm{sp}}(A, B) = \frac{\sum_{i \neq j} \kappa^{\mathrm{sp}}(a_i, a_j)}{n_A(n_A - 1)} + \frac{\sum_{i \neq j} \kappa^{\mathrm{sp}}(b_i, b_j)}{n_B(n_B - 1)} - \frac{2\sum_{i,j} \kappa^{\mathrm{sp}}(a_i, b_j)}{n_A n_B}. \tag{47}$$

Smaller values indicate more similar distributions of these fixed-grid descriptors, not necessarily more similar physical geometry or semantic content. The estimate may be slightly negative for finite banks. Figure 10 shows the spatially normalized version, with each diagonal comparing two disjoint halves of the same dataset.

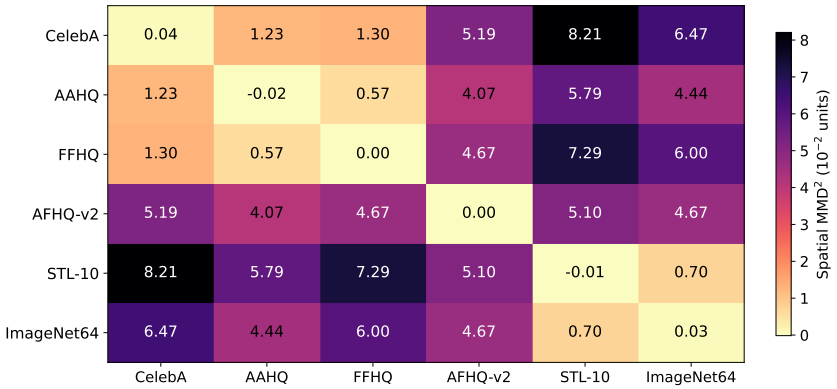

Figure 10: **Spatial self-similarity distance between real datasets.** Unbiased Gaussian-kernel MMD^2 of spatially normalized, fixed-grid DINOv2 patch descriptors. Cell and colorbar values are in units of 10^{-2}. Off-diagonal cells compare 1,000 unpaired images from each dataset; diagonal cells compare disjoint 500-image halves. Lower means more similar descriptor distributions.

E.4.2 Endpoint agreement and component contrasts

For each transfer direction and adaptation seed, write $\mathbf{b} = (b_Q, b_V, b_M) \in \{0, 1\}^3$ for a hybrid state in QK/VO/MLP order: 0 restores source weights and 1 retains jointly adapted weights. A bit string such as 011 abbreviates $\mathbf{b} = (0, 1, 1)$. Let $x_0^{(\mathbf{b})}(\xi_i)$ be the clean image generated by hybrid $\mathbf{b}$ from the ith shared Gaussian sampling initialization ξ_i; $x_0^{(000)}(\xi_i)$ is the source output and $x_0^{(111)}(\xi_i)$ is the joint-adaptation output. Visual comparisons use the same latents but can be ambiguous, so we also measure their agreement with the paired joint endpoint. Let $f_{\mathrm{g}}(x_0)$ be the unit-normalized final DINOv2 class token and let $v(x_0)$ be the spatial descriptor above. For $N = 1024$ matched latents, define

$$\mathcal{E}^{\mathrm{g}}_{\mathbf{b}} = \frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{g}}(x_0^{(\mathbf{b})}(\xi_i)) - f_{\mathrm{g}}(x_0^{(111)}(\xi_i))\|_2^2, \qquad \mathrm{EA}_{\mathbf{b}} = 1 - \frac{\mathcal{E}^{\mathrm{g}}_{\mathbf{b}}}{\mathcal{E}^{\mathrm{g}}_{000}}, \tag{48}$$

$$\mathcal{E}^{\mathrm{sp}}_{\mathbf{b}} = \frac{1}{N}\sum_{i=1}^{N} \frac{\|v(x_0^{(\mathbf{b})}(\xi_i)) - v(x_0^{(111)}(\xi_i))\|_2^2}{32640}, \qquad \mathrm{SA}_{\mathbf{b}} = 1 - \frac{\mathcal{E}^{\mathrm{sp}}_{\mathbf{b}}}{\mathcal{E}^{\mathrm{sp}}_{000}}. \tag{49}$$

We compute EA only when $\mathcal{E}^{\mathrm{g}}_{000} \geq 0.01$ and SA only when $\mathcal{E}^{\mathrm{sp}}_{000} \geq 10^{-6}$. Both scores set the source to 0 and the joint endpoint to 1; negative values are retained if a hybrid is farther from the joint endpoint than the source. EA describes global feature agreement and SA describes agreement of patch self-similarity at fixed image coordinates. SA and the dataset MMD use the same spatial

descriptor but answer different questions: SA compares matched generated images, while MMD compares distributions of unpaired real images.

For either score $\mathrm{Agr}_{\mathbf{b}} \in \{\mathrm{EA}_{\mathbf{b}}, \mathrm{SA}_{\mathbf{b}}\}$, finite differences across cube states quantify component dependence. The penalty for reverting one adapted group from the joint model is $R_Q = \mathrm{Agr}_{111} - \mathrm{Agr}_{011}$, $R_V = \mathrm{Agr}_{111} - \mathrm{Agr}_{101}$, or $R_M = \mathrm{Agr}_{111} - \mathrm{Agr}_{110}$; we also use $R_{\min} = \min(R_Q, R_V, R_M)$ to ask whether all three reversions are costly. The gain beyond adding the three isolated changes is

$$\mathrm{JNA}(\mathrm{Agr}) = \mathrm{Agr}_{111} - \mathrm{Agr}_{100} - \mathrm{Agr}_{010} - \mathrm{Agr}_{001} + 2\mathrm{Agr}_{000},$$
$$\mathrm{TWI}(\mathrm{Agr}) = \mathrm{Agr}_{111} - \mathrm{Agr}_{110} - \mathrm{Agr}_{101} - \mathrm{Agr}_{011} + \mathrm{Agr}_{100} + \mathrm{Agr}_{010} + \mathrm{Agr}_{001} - \mathrm{Agr}_{000}.$$

Here JNA measures joint non-additivity relative to the three isolated gains, while TWI isolates the three-way interaction after subtracting pairwise interactions. Positive $R_{\min}$ means each single reversion lowers endpoint agreement; positive JNA means the joint gain exceeds the sum of isolated gains. All contrasts are computed within each seed's cube before averaging seeds. Figure 11 gives the two agreement scores for every hybrid state and transfer direction; interpretation of their component patterns follows in the next section.

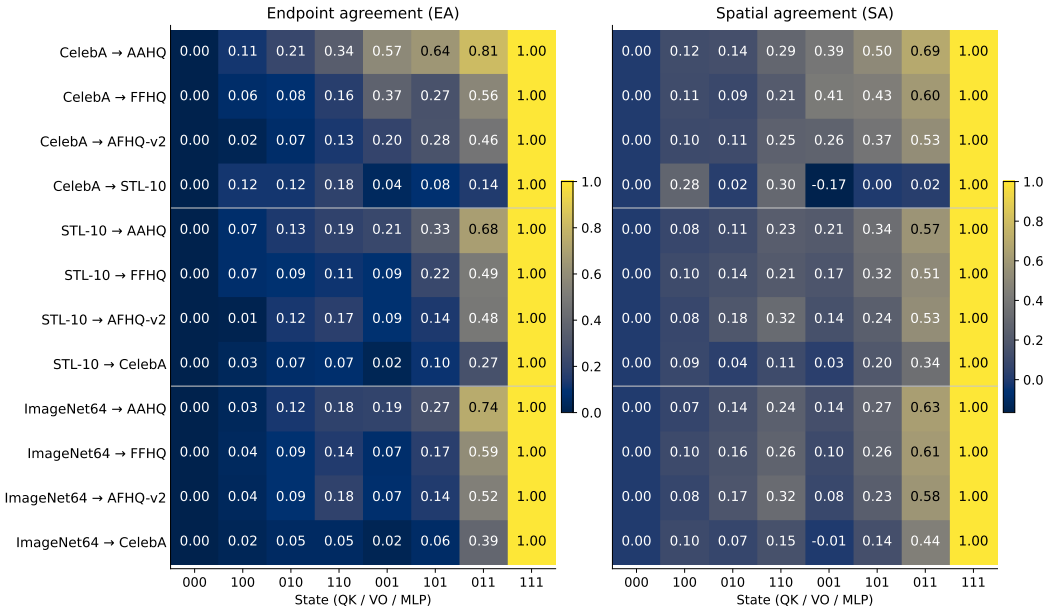

Figure 11: **Agreement of all hybrid states with the jointly adapted endpoint.** Global-feature endpoint agreement EA (left) and relative spatial endpoint agreement SA (right), averaged over two adaptation seeds. Each row is a transfer direction, and columns give QK/VO/MLP bits (0: source, 1: jointly adapted). The same 1,024 noise latents are used for all eight states of each cube. Each score is 0 for the source and 1 for the joint endpoint; negative values are possible.

E.4.3 Results and interpretation

(1) Higher spatial similarity is associated with QK compatibility; larger shifts accompany coadaptation. Figure 5(a) shows that spatial MMD^2 correlates with the QK reversion difference $\mathrm{Agr}_{111} - \mathrm{Agr}_{011}$ ($\rho_{\mathrm{EA}} = 0.739$, $\rho_{\mathrm{SA}} = 0.844$): source QK remains more compatible with adapted VO+MLP when dataset geometry is similar. The selected CelebA→AAHQ/FFHQ images retain recognizable faces in state 011, whereas CelebA→STL-10 has little recognizable structure until joint adaptation (111; Figs. 1 and 13). Across all twelve nested state differences, this QK difference ranks first by the weaker of its two absolute correlations (Table 5). Figure 12 extends the pattern to joint non-additivity, the minimum single-component reversion penalty, and three-way interaction: all increase with spatial distance ($\rho_{\mathrm{EA}}/\rho_{\mathrm{SA}} = 0.634/0.655$, $0.739/0.844$, and $0.641/0.774$, respectively). These observations are consistent with stronger component coadaptation across larger spatial shifts, subject to the dependence limitations noted above.

Table 5: Spatial distance and nested state differences. Spearman correlations with dataset spatial MMD^2 use 12 two-seed direction means for each score. Each row changes one component with its partners fixed. Rows are sorted by decreasing $\min(|\rho_{\mathrm{EA}}|, |\rho_{\mathrm{SA}}|)$.

Interpretation	State difference	ρ_{EA}	ρ_{SA}
QK reversion penalty†	$\mathrm{Agr}_{111} - \mathrm{Agr}_{011}$	**0.739**	**0.844**
MLP adaptation cost (VO adapted)†	$\mathrm{Agr}_{010} - \mathrm{Agr}_{011}$	0.718	0.809
VO adaptation cost (QK adapted)	$\mathrm{Agr}_{100} - \mathrm{Agr}_{110}$	0.771	0.683
MLP-only adaptation cost	$\mathrm{Agr}_{000} - \mathrm{Agr}_{001}$	0.746	0.673
MLP adaptation cost (QK adapted)	$\mathrm{Agr}_{100} - \mathrm{Agr}_{101}$	0.644	0.673
VO reversion penalty	$\mathrm{Agr}_{111} - \mathrm{Agr}_{101}$	0.641	0.673
VO-only adaptation cost	$\mathrm{Agr}_{000} - \mathrm{Agr}_{010}$	0.455	0.424
QK adaptation cost (VO adapted)	$\mathrm{Agr}_{010} - \mathrm{Agr}_{110}$	0.676	0.378
MLP reversion penalty	$\mathrm{Agr}_{111} - \mathrm{Agr}_{110}$	0.602	0.375
QK adaptation gain (MLP adapted)	$\mathrm{Agr}_{101} - \mathrm{Agr}_{001}$	0.266	0.676
QK-only adaptation gain	$\mathrm{Agr}_{100} - \mathrm{Agr}_{000}$	0.081	0.179
VO adaptation cost (MLP adapted)	$\mathrm{Agr}_{001} - \mathrm{Agr}_{011}$	0.168	0.018

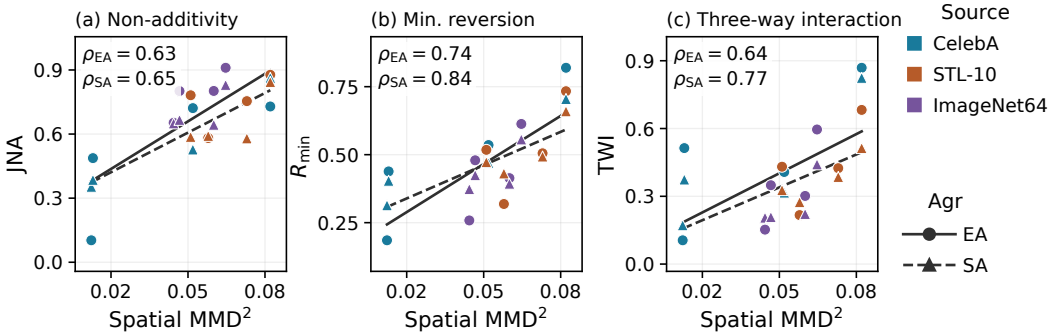

Figure 12: **Spatial distance and functional coordination.** Spatial MMD^2 versus (a) joint non-additivity, (b) minimum reversion penalty, and (c) three-way interaction, defined in Appendix E.4.2. Points average two adaptation seeds; colors, markers, and lines follow Fig. 5.

(2) VO dependence increases with functional coadaptation. In Fig. 5(b), joint non-additivity correlates strongly with the VO reversion difference $\mathrm{Agr}_{111} - \mathrm{Agr}_{101}$ ($\rho_{\mathrm{EA}} = 0.832$, $\rho_{\mathrm{SA}} = 0.930$). Table 6 shows that this association exceeds those for reverting QK or MLP in both scores. The generated images illustrate the same dependence: reverting VO in state 101 leaves fragmented texture for CelebA→STL-10 and distorted outputs for STL-10/ImageNet64→CelebA, compared with the coherent joint endpoints in 111 (Figs. 1, 14, and 15).

(3) Broad-domain source QK retains partial compatibility on face targets. Figure 5(c) shows partial source-QK agreement across AAHQ, FFHQ, and AFHQ for all three sources ($\mathrm{EA}_{011} = 0.465$–$0.815$, $\mathrm{SA}_{011} = 0.507$–$0.686$). For STL-10→CelebA, retained agreement $(\mathrm{EA}_{011}, \mathrm{SA}_{011}) = (0.267, 0.340)$ exceeds the reciprocal CelebA→STL-10 values $(0.143, 0.021)$; ImageNet64→CelebA also retains partial agreement $(0.387, 0.444)$. The selected natural-source images show varying face or animal structure with source QK in 011, while 111 generally yields clearer endpoints; some STL-10→CelebA examples remain weak in 011 (Fig. 14).

Table 6: Component dependence and joint non-additivity. Spearman correlations between $\mathrm{JNA}_{\mathrm{Agr}}$ and each displayed difference use 12 two-seed direction means, separately for $\mathrm{Agr} = \mathrm{EA}, \mathrm{SA}$. Note that since all quantities share cube terms, the differences between correlations are descriptive.

Reverted component	State difference	ρ_{EA}	ρ_{SA}
VO	$\mathrm{Agr}_{111} - \mathrm{Agr}_{101}$	**0.832**	**0.930**
QK	$\mathrm{Agr}_{111} - \mathrm{Agr}_{011}$	0.601	0.573
MLP	$\mathrm{Agr}_{111} - \mathrm{Agr}_{110}$	0.664	0.063

E.4.4 ADDITIONAL GENERATED-IMAGE CUBES

Figures 13–15 extend Fig. 1 with more latents and all eight component states per target. They use the same layout and selected endpoints (Table 4), with EMA weights, FP32 sampling, and 50-step VP Heun.

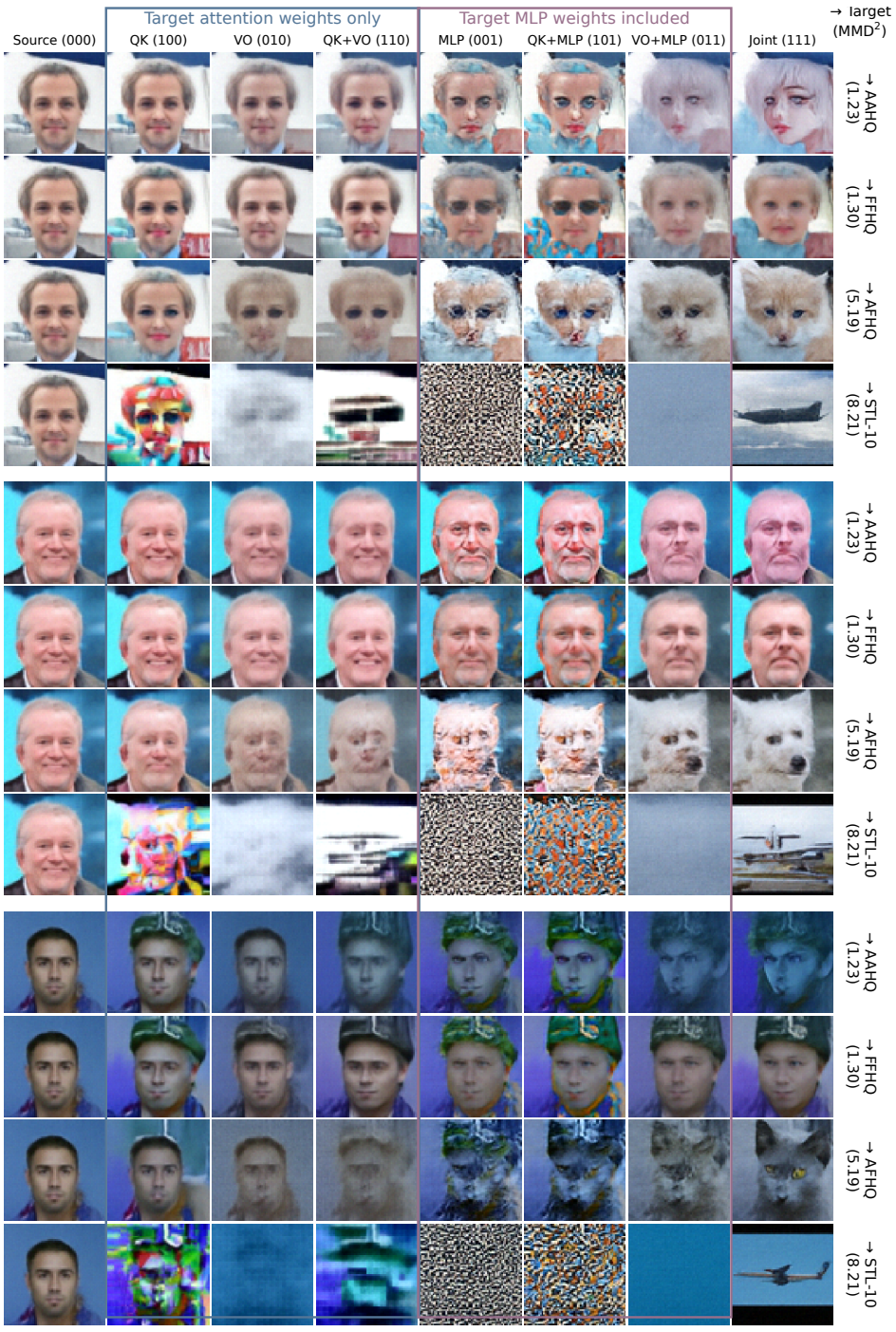

Figure 13: CelebA source: three additional latents across four targets.

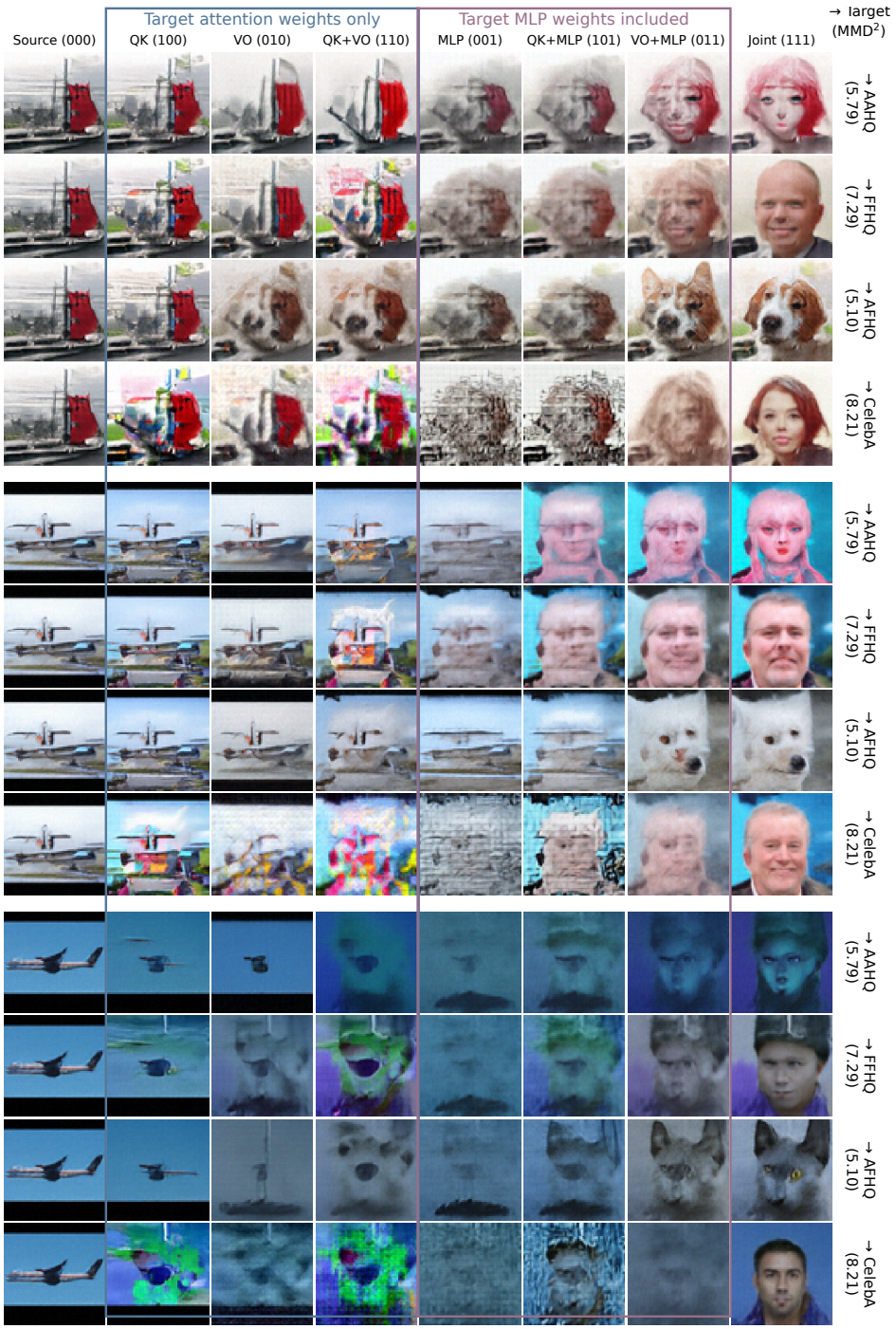

Figure 14: STL-10 source: three latents across four targets.

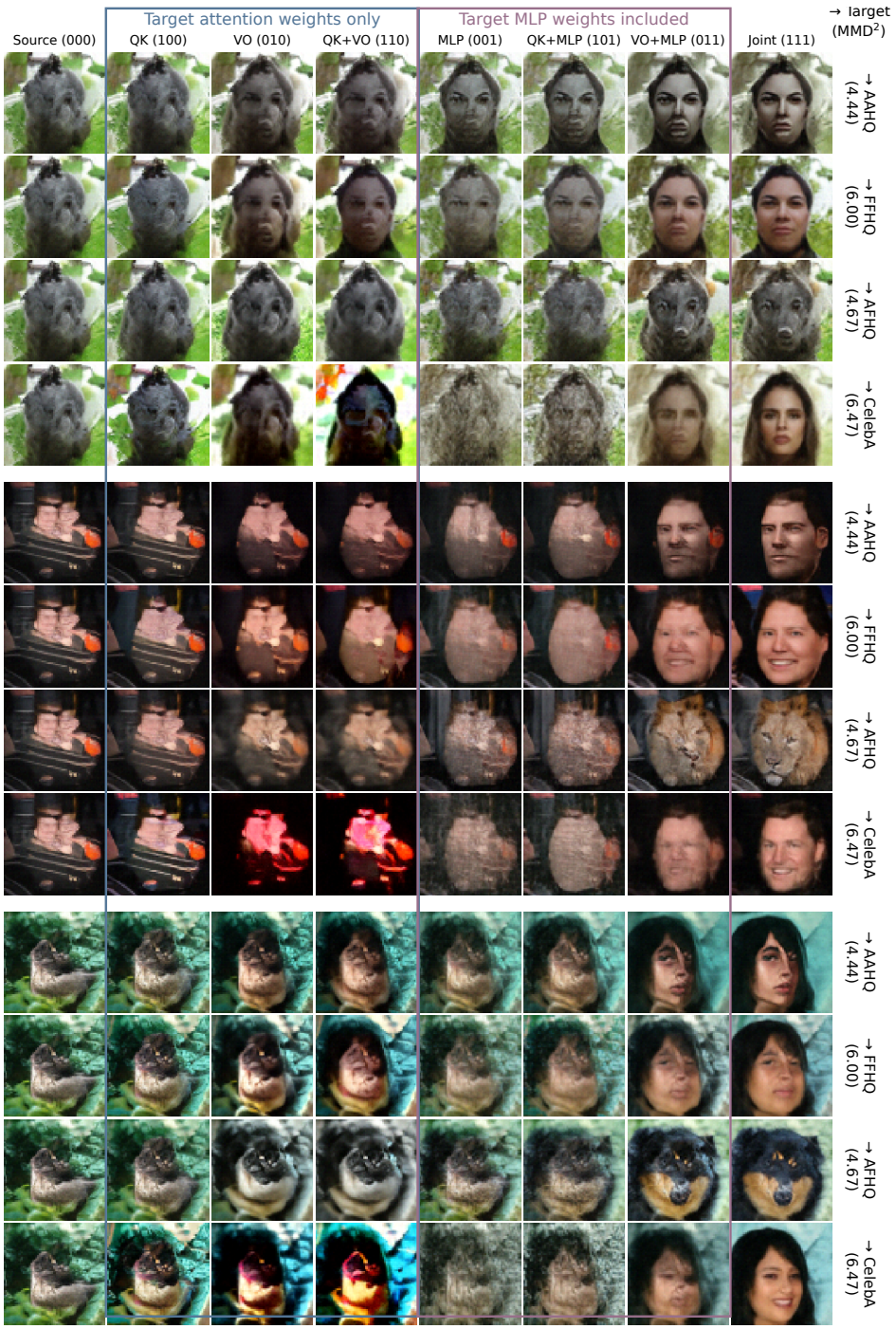

Figure 15: ImageNet64 source: three latents across four targets.

E.4.5 TRAINING TRAJECTORIES AND RESTRICTED FINE-TUNING

We compare joint QK/VO/MLP fine-tuning (111) with training all parameters from scratch and with independent fine-tuning restricted to QK or QK+VO, using the same target-domain update budget (Figs. 16–18). Joint fine-tuning is comparable to or better than scratch training in held-out denoising and usually in FID-50k; STL-10→CelebA is an exception for FID at the selected joint endpoint. QK-only and QK+VO-only fine-tuning generally lag further behind on larger domain shifts under the tested training protocol and update budget. Each target has one row, with denoising AUC, VP velocity loss, and FID-50k in columns; lower is better. The two loss axes use 10^{-2} units, as in Fig. 9. Solid and dashed curves denote adaptation seeds 0 and 1, and dotted scratch curves use seed 0. Vertical lines mark the selected joint endpoints; source pretraining is excluded from the target-domain update count.

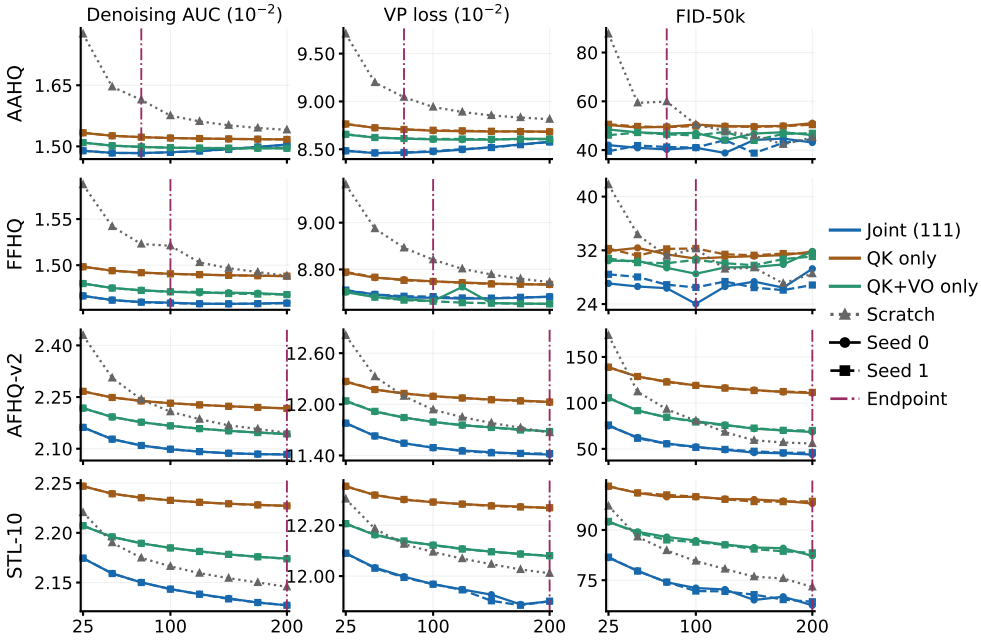

Figure 16: **CelebA-source adaptation trajectories.** Rows are AAHQ, FFHQ, AFHQ-v2, and STL-10 targets.

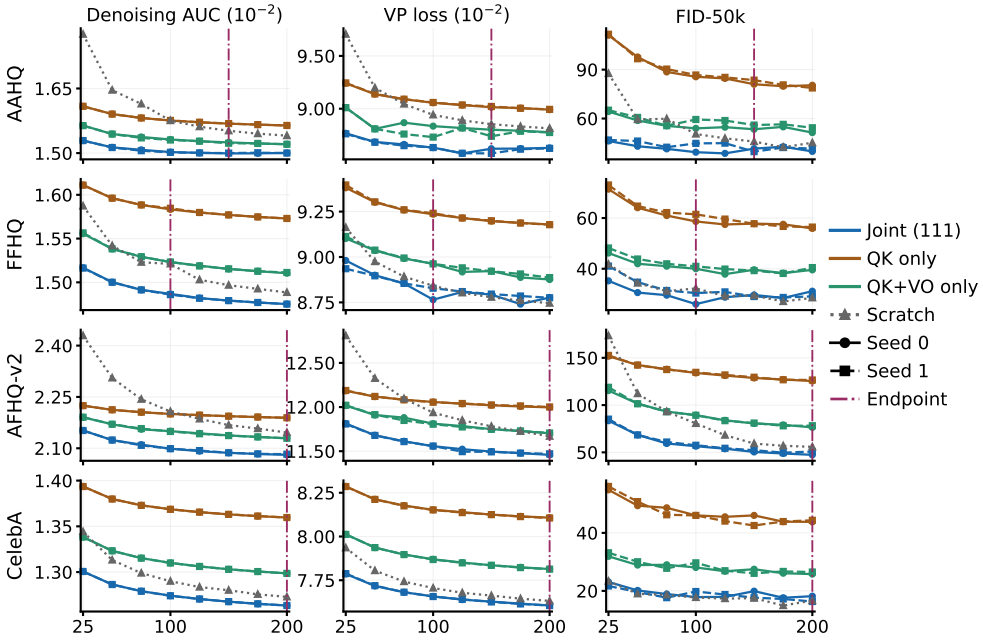

Figure 17: **STL-10-source adaptation trajectories.** The source checkpoint is from epoch 800.

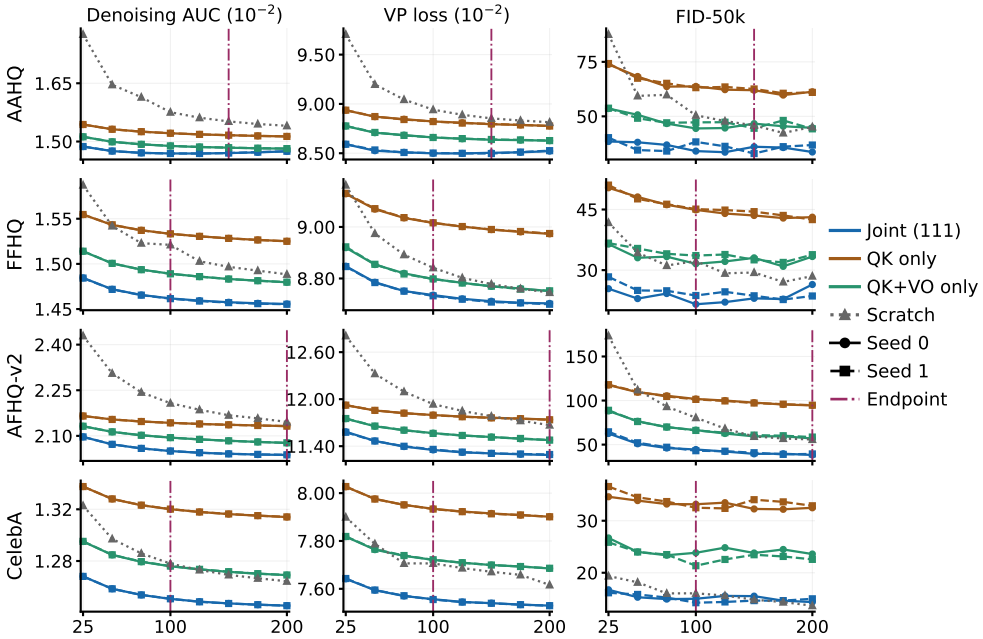

Figure 18: **ImageNet64-source adaptation trajectories.** The source checkpoint is from epoch 800; rows are AAHQ, FFHQ, AFHQ-v2, and CelebA targets.