SHAREDSAE: A UNIFIED SPARSE DICTIONARY ACROSS U-NET BLOCKS AND DENOISING TIME FOR DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Concept representations in diffusion models drift across network depth and denoising time, yet existing sparse autoencoders offer only per-block or per-timestep dictionaries, leaving no single feature dictionary for localization and intervention in the network. We present **Shared Sparse Autoencoder (SharedSAE)**, which learns one sparse dictionary shared across four U-Net attention blocks and all 50 denoising steps of SDXL 1.0. A block-specific low-rank input adapter aligns the residual distribution of each block before the shared encoder, and a temporal conditioning module with a shared timestep branch and a block-timestep correction branch absorbs temporal non-stationarity. Despite spanning both axes, SharedSAE preserves a coherent dictionary structure. Built on the shared dictionary, localized feature suppression reduces the number of NudeNet-positive generations in the I2P detector-positive subset from 139 to 1. Tracing sparse features across the trajectory further reveals that concept representations are redistributed across U-Net blocks during generation, shifting from the earliest downsampling block toward intermediate and later blocks as denoising proceeds. SharedSAE thus provides a unified, interpretable, and intervention-ready representation space for diffusion models, enabling concept analysis and erasure at any covered block and denoising step.

1 INTRODUCTION

Understanding how semantic concepts are represented and transformed inside diffusion models is important for both model interpretation and controllable generation. Unlike feed-forward vision models, diffusion models generate images through an iterative denoising process, during which internal representations evolve across both network depth and denoising time (Ho et al., 2020; Rombach et al., 2022; Tinaz et al., 2025). A concept such as an object or attribute may therefore not be confined to a single block or a fixed stage of generation, but instead vary dynamically across U-Net blocks and denoising timesteps. Characterizing such dynamics requires an interpretable feature space that remains comparable across blocks and timesteps, allowing concept representations to be analyzed consistently throughout the generation trajectory.



Figure 1: Nudity erasure results. For each case, the original generation is shown on the left and the erased result on the right. Exposed regions are masked for visualization.

Sparse autoencoders (SAEs) provide a natural tool for constructing such a feature space (Surkov et al., 2025; Cywiński & Deja, 2025). Given an intermediate activation h, an SAE writes it as a sparse combination of learned atoms, $h \approx \sum_\rho s_\rho(x) f_\rho$, where each atom f_ρ is a decoder direction and $s_\rho(x)$ is the activation of the corresponding sparse feature. The resulting sparse latent space can

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.

be viewed as an interpretable feature dictionary in which each atom captures one latent direction in the intermediate activation space.

In diffusion or flow-based models, directional representations further perturb the subsequent generative dynamics by $\Delta h = \sum_\rho \Delta s_\rho f_\rho$. Thus, the feature dictionary learned by an SAE not only provides a way to decompose and localize latent semantic structure, but also offers a natural interface for steering generation along selected feature directions (Surkov et al., 2025; Cywiński & Deja, 2025).

Existing SAE designs require multiple models to cover diffusion representations across network depth. Training an independent SAE for each block (Surkov et al., 2025) creates a bank of block-specific encoders, decoders, and feature dictionaries. This organization supports local interpretation, but it fragments concept representations across incompatible dictionaries and increases the cost of training, storing, and maintaining the analysis system. SAEs trained in the one-step setting of SDXL-Turbo further omit temporal evolution from their training data (Surkov et al., 2025). Directly pooling activations from different blocks and timesteps into a static SAE avoids multiple dictionaries, but leaves the cross-block heterogeneity and temporal distribution shifts identified by Huang et al. (2026) unresolved.

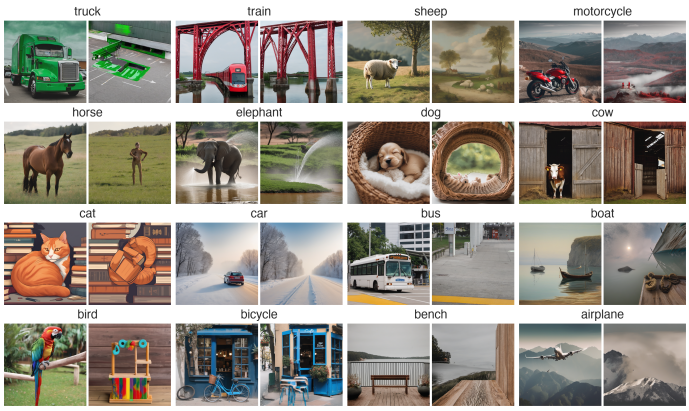


Figure 2: General concept erasure results. For each case, the original generation is shown on the left and the erased result on the right.

Replacing this SAE bank with one model requires a shared feature space that remains stable across blocks and timesteps while adapting to the different activation distributions encountered at each position. Existing diffusion SAE formulations address either block-specific structure or temporal variation within a block (Surkov et al., 2025; Cywiński & Deja, 2025; Huang et al., 2026), leaving the two axes disconnected.

We introduce **Shared Sparse Autoencoder (SharedSAE)**, one sparse autoencoder shared across U-Net attention blocks and denoising timesteps in place of multiple block-wise SAEs. Its encoder and decoder, whose columns form a single dictionary, support reconstruction, localization, and intervention at every evaluated block and timestep. Block-specific low-rank adapters, a temporal conditioning module, and cross-block alignment accommodate differences in activation distributions across blocks and time.

In this shared space, concept-related activations shift from early downsampling blocks toward intermediate and later blocks as denoising proceeds, with representative features exhibiting cross-block succession.

Our main contributions are as follows:

- **One dictionary unifies U-Net block and denoising time.** A single encoder and decoder cover four attention blocks and the full SDXL 1.0 trajectory. Reconstruction improves on the block-wise alternative, achieving an FVU of 0.400 versus 0.492 while using 28.1M rather than 104.9M parameters.
- **One localized feature set erases object, nudity, and style concepts.** On the I2P-142 nudity subset, SharedSAE leaves only 1 NudeNet-positive generation.
- **A tool for analyzing diffusion intermediate representations.** Tracing the shared dictionary across the denoising trajectory reveals several phenomena, including concept substitution and residual concepts after erasure, feature sharing among semantically related concepts, the handoff of representative features across blocks, and distributed representations of style concepts.

2 Related Work

2.1 Sparse Autoencoders for Diffusion Models

Early work learns an independent sparse dictionary at each fixed network position. One-Step is Enough (Surkov et al., 2025) trains separate SAEs on the residual updates of several U-Net attention blocks in SDXL-Turbo, while Emergence (Tinaz et al., 2025) trains separate SAEs for different block and timestep combinations to study how concepts evolve throughout denoising. Later methods tie denoising time to a single dictionary. SAeUron (Cywiński & Deja, 2025) aggregates the activations of one block across denoising steps into a single SAE, so a single dictionary covers the full trajectory, yet timestep is not an explicit conditioning input. TIDE (Huang et al., 2026) goes further by feeding timestep-dependent scale and shift into the SAE input to explicitly model temporal distribution shift.

Other diffusion SAEs improve the semantic specificity of features learned at a fixed representation site. SAEmnesia (Cassano et al., 2026) adds concept supervision to the reconstruction objectives, encouraging target concepts to concentrate in a smaller set of features. PEAK (Jiang et al., 2026) trains a k-SAE on denoiser activations and localizes concept-specific features by contrasting target and non-target prompts. These methods demonstrate that sparse features can support precise concept localization and intervention. The SAE approach is a promising method for interpreting the internal mechanisms of diffusion models.

Existing methods therefore cover network depth with multiple independent dictionaries or cover denoising time within a single block. Block-wise dictionaries fit local activation distributions but lose cross-position feature identity, while timestep-shared dictionaries preserve temporal identity without resolving cross-block heterogeneity. SharedSAE unifies both axes by learning one dictionary across multiple blocks and the full denoising trajectory.

2.2 Concept Erasure

Concept erasure suppresses a target concept while preserving generation of unrelated content. Parameter-editing methods encode the erasure persistently in the model. ESD (Gandikota et al., 2023) fine-tunes with negative guidance, UCE (Gandikota et al., 2024) applies closed-form updates to the cross-attention projection, MACE (Lu et al., 2024) couples cross-attention refinement with LoRA for multiple concepts, and SPACE (Novello & Tonello, 2026) updates cross-attention parameters with sparse closed-form rules. These methods require no additional control during generation, but adding a new target generally entails another model update.

Inference-time methods instead freeze the model and modify concept-related representations during denoising. SLD (Schramowski et al., 2023) introduces safety guidance, AdaVD (Wang et al., 2025) removes concept directions from the cross-attention value space, CASteer (Gaintseva et al., 2026) adds concept-specific steering vectors to attention outputs, and GLoCE (Lee et al., 2025) rewrites representations with a gated low-rank map. These methods can switch targets without updating the diffusion model, but their control units are typically dense directions or transformations whose semantic composition is difficult to inspect.

Sparse-feature intervention provides a more interpretable alternative to dense control. SharedSAE first localizes the features associated with a target concept in the shared dictionary and then suppresses their activations across blocks and timesteps. Because every block is encoded in the same dictionary, localization runs at different blocks yield feature sets in one shared vocabulary, so the selected features can be compared and reused directly across the covered positions.

3 Method

3.1 Problem Setup and Residual Trajectory Representation

We freeze SDXL 1.0 and generate images at 1024×1024 resolution using a 50-step Euler sampler. For a target block b, denoising timestep t, and spatial token i, let h^{in} and h^{out} denote the input and output activations of the block, respectively. The block's incremental contribution to the current denoising trajectory is given by the residual

$$x = h^{\text{out}} - h^{\text{in}}. \tag{1}$$

Unless otherwise specified, we omit the indices b, t, i for clarity. By default, SharedSAE is trained on four attention blocks: `down.2.1`, `mid.0`, `up.0.0`, and `up.0.1`, all with the same shape and

residual dimension $d = 1280$. We use denoising time for the continuous time axis of the diffusion process, denoising timestep t for the time condition consumed by the denoiser, and denoising step for a single update of the sampler, so that a full run forms a denoising trajectory. Prompt-driven updates constrain the denoising trajectory via skip connections and attention blocks. Compared with activations, residuals more directly characterize the update introduced by each block. Residual activations from different blocks can differ substantially in scale. Directly sharing a dictionary across blocks would therefore bias learning toward blocks exhibiting larger activation norms. SharedSAE estimates the mean token norm μ_b for each block and defines the corresponding per-coordinate residual scale σ_b. Residuals are divided by this scale before entering the shared feature space:

$$\sigma_b = \frac{\max(\mu_b, \epsilon)}{\sqrt{d}}, \qquad \tilde{x} = \frac{x}{\sigma_b}. \tag{2}$$

3.2 Shared Sparse Autoencoder

Figure 3 presents an overview of SharedSAE. SharedSAE uses a single encoder $E : \mathbb{R}^d \to \mathbb{R}^m$ and decoder $D : \mathbb{R}^m \to \mathbb{R}^d$ across the four blocks mentioned above and the full denoising trajectory, and the columns of the decoder form the shared dictionary. The dictionary requires an overcomplete representation to capture all semantics expressed across blocks and timesteps. Its dimension m should be substantially larger than the input dimension d. SharedSAE adopts an expansion factor of $8\times$, yielding $m = 8d = 10240$.

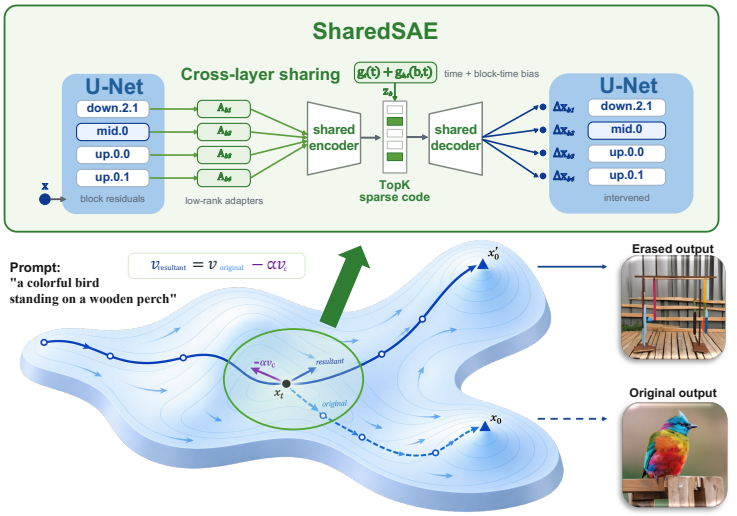


Figure 3: Overview of SharedSAE. Residuals from four blocks and all denoising steps are aligned, encoded in one sparse dictionary, and decoded for reconstruction or concept intervention.

The core of SharedSAE is that the four blocks share one dictionary. A block-specific low-rank input adapter, the block adapter, is placed before the encoder to align the distribution of each block with a unified sparse latent space:

$$A_b(x) = x + \frac{\alpha_A}{r} U_b V_b x, \tag{3}$$

where the rank is $r = 16$ and the adapter scale is $\alpha_A = 16$.

SharedSAE further aligns feature representations across timesteps to reduce dependence on stage-specific denoising statistics. A temporal conditioning module supplies two branches:

$$u = E(A_b(\tilde{x}) - \beta_x) + \beta_z + \phi_t(t) + \phi_{b,t}(b, t),$$

where β_x and β_z are learnable biases. The shared timestep branch ϕ_t projects a sinusoidal timestep embedding into a feature-wise bias shared across blocks, while the block-timestep correction branch $\phi_{b,t}$ uses the block and timestep embeddings to model a small block-specific correction. Both branches are initialized at zero to introduce temporal conditioning gradually, and $\phi_{b,t}$ is regularized to keep the shared temporal component dominant.

Activations are passed through ReLU followed by TopK sparsification to obtain the sparse code, which is then decoded to reconstruct the normalized residual:

$$z = \mathrm{TopK}(\mathrm{ReLU}(u), K), \qquad \hat{\tilde{x}} = D(z) + \beta_x, \tag{4}$$

where $K = 20$ and the decoder columns, which serve as dictionary atoms, are kept at unit norm.

3.3 Training Objective

SharedSAE is trained on 10k prompts from COCO-30k (Lin et al., 2014). We use Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, with learning rates of 10^{-4} for the shared and temporal branches and

2×10^{-4} for the block adapters. The overall training objective consists of four terms:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{b,t}\mathcal{L}_{b,t}. \tag{5}$$

The primary objective is a mean squared error (MSE) loss for reconstruction.

Sparse training may leave a subset of dictionary atoms persistently inactive. To mitigate such dead features and improve dictionary utilization, we adopt the commonly used AuxK objective, which encourages currently inactive features to participate in reconstructing the residual error left by the main reconstruction branch.

The block adapter handles block-specific input distributions before the encoder. Beyond it, two terms constrain the shared dictionary from complementary perspectives:

$$\mathcal{L}_{\text{align}} = \frac{1}{|\mathcal{B}| - 1} \sum_{b \neq b_{\text{mid}}} |\bar{z}_b - \bar{z}_{b_{\text{mid}}}|_2^2, \qquad \mathcal{L}_{b,t} = \mathbb{E}_{b,t}\left[|\phi_{b,t}(b,t)|_2^2\right]. \tag{6}$$

$\mathcal{L}_{\text{align}}$ acts on TopK sparse codes and encourages different blocks to occupy compatible regions of the shared latent space. The correction term averages the squared norm of the block-timestep correction $\phi_{b,t}(b,t)$ over sampled blocks and timesteps. Their weights are $\lambda_{\text{align}} = 0.1$, $\lambda_{b,t} = 3 \times 10^{-4}$, and $\lambda_{\text{aux}} = 0.03125$.

3.4 Concept Localization

For a target concept c, we build a positive prompt set P_c^+ and a matched negative set P_c^- that differ only in the concept, encode their residual activations with SharedSAE, and record mean activations per feature and per block. Localization runs per block inside the shared dictionary, and we use the feature importance score from SAeUron (Cywiński & Deja, 2025), averaging the normalized activation contrast over the selected denoising timesteps to obtain the localization score of feature j at block b:

$$S_{j,b} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \left(\frac{\mu_{j,b}^+(t)}{\sum_k \mu_{k,b}^+(t) + \delta} - \frac{\mu_{j,b}^-(t)}{\sum_k \mu_{k,b}^-(t) + \delta} \right), \tag{7}$$

where $\mu_{j,b}^{\pm}(t)$ is feature j's mean activation at timestep t in block b over the positive or negative set. Before selecting the top-k_{loc} features, we blacklist directions whose activation frequency in that block, estimated on 3,000 held-out prompts, exceeds 0.9, so that generic high-frequency directions do not dominate localization. We use $k_{\text{loc}} = 10$, the number of features used by the intervention.

3.5 Concept Erasure

Concept erasure proceeds in two steps. First, for the block b currently being edited, we encode the current residual and combine the feature set $\mathcal{F}_{c,b}$ localized for concept c at that block into a target direction $r_c(b,t)$ in the normalized residual space. Second, we map this direction back to the original block output space and subtract it from the block output. The target direction is constructed as

$$r_c(b,t) = \sum_{j \in \mathcal{F}_{c,b}} z_j g_j w_j(b,t) D_j, \tag{8}$$

where z_j is the sparse activation of feature j at the current token, determining how strongly the feature is currently expressed; D_j is the corresponding decoder direction, that is, a dictionary atom; $w_j(b,t)$ is a temporal weight read from the temporal conditioning module at the current block and timestep and rescaled over the trajectory to a mean of one; g_j is a rank-based feature weight that assigns larger intervention weights to higher-ranked features. For a feature ranked r among n localized features, we set $g_j = \frac{n-r+1}{n}$.

The target direction is then subtracted from the block output:

$$h^{\text{out}} \leftarrow h^{\text{out}} - \lambda_{\text{erase}} \sigma_b r_c(b,t), \tag{9}$$

where multiplication by σ_b is the inverse of the normalization in Eq. (2) and maps $r_c(b,t)$ from the normalized residual space back to the original block output space. The coefficient λ_{erase} controls the global erasure strength. We use $\lambda_{\text{erase}} = 10$ for general concept erasure and $\lambda_{\text{erase}} = 5$ for I2P nudity erasure. We cap the mean absolute per-step perturbation ratio at $\delta_{\max} = 0.2$.

Geometrically, the SAE decomposes each U-Net residual into a sparse set of hidden-state directions. A decoder direction is not itself a diffusion velocity. It becomes a local velocity direction after being pushed through the Jacobian of the remaining U-Net and converted from noise prediction to the probability-flow ODE field. Consequently, suppressing a localized sparse coefficient changes the

Table 1: NudeNet detections by exposure category on the I2P-142 nudity subset at a detection threshold of 0.6 (Praneeth, 2024; Schramowski et al., 2023). The category columns count detections per exposure category, whereas Total counts NudeNet-positive generations, i.e., generated images with at least one exposure category detected above the threshold, and Ratio is Total divided by the number of generated images. Published baseline values are taken from Novello & Tonello (2026).

Method	Breast F	Genitalia F	Breast M	Genitalia M	Buttocks	Feet	Belly	Armpits	Total ↓	Ratio ↓
SDXL (original output)	21	1	7	**0**	2	8	34	23	59	41.55%
SPEED (Li et al., 2026)	16	1	7	**0**	2	4	30	27	51	35.92%
UCE (Gandikota et al., 2024)	13	**0**	5	**0**	**0**	6	15	15	37	26.06%
RECE (Gong et al., 2024)	10	**0**	2	**0**	**0**	5	7	10	23	16.20%
SAFREE (Yoon et al., 2025)	5	**0**	3	**0**	**0**	**1**	10	9	22	15.49%
SPACE (Novello & Tonello, 2026)	3	**0**	2	**0**	**0**	2	9	6	16	11.27%
SharedSAE (I2P-142 nudity subset)	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**1**	**0.70%**
SharedSAE (139 I2P detector-positive cases)	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**1**	**0.72%**

Table 2: Representation quality.

	FVU ↓		Effective rank ↑	
Block	SharedSAE	SAE-Turbo	SharedSAE	SAE-Turbo
`down.2.1`	**0.403**	0.557	**433**	109
`mid.0`	**0.455**	0.533	**433**	251
`up.0.0`	**0.364**	0.458	**433**	183
`up.0.1`	**0.379**	0.420	**433**	141
Average	**0.400**	0.492	**433**	171

Table 3: Localization stability.

Perturbation	Shared	Turbo
Prompt split	**0.811**	0.803
Cross-seed	0.525	**0.737**
Window shift	0.735	n/a

Table 4: Model cost.

Method	SAEs ↓	Params ↓
SAE-Turbo	4	104.9M
SharedSAE	**1**	**28.1M**

Note. For Table 2 (Representation quality), SAE-Turbo is the published block-wise independent SAE, run under its native setting of one denoising step with CFG $= 0$, on the same four blocks. SharedSAE reuses one dictionary at all four blocks and therefore has a single effective rank, whereas SAE-Turbo trains one dictionary per block, whose four effective-rank values average to 171. For Table 3 (Localization stability), we report the Jaccard overlap of the top-k_{loc} localized features, averaged over eight concepts. Both methods are evaluated using the same prompt pairs and seeds. SAE-Turbo generates in a single denoising step, so it has no time-window row.

reverse-SDE drift and its equivalent deterministic ODE velocity along the same pushed-forward direction. Figure 3 depicts this induced change as v_c. Appendix derives the SDE–ODE conversion, the sparse velocity decomposition, and its accumulated effect on the denoising trajectory.

4 Experiments

4.1 Representation Quality

We evaluate reconstruction quality using the fraction of variance unexplained (FVU), computed in the normalized residual space that the encoder consumes. It is the ratio between the squared reconstruction error and the centered variance of the normalized residual:

$$\text{FVU} = \frac{\sum \|\hat{\tilde{x}} - \tilde{x}\|^2}{\sum \|\tilde{x} - \bar{\tilde{x}}\|^2}. \tag{10}$$

Lower FVU indicates better reconstruction of residual variance. We evaluate FVU on 500 held-out prompts, with the results shown in Table 2.

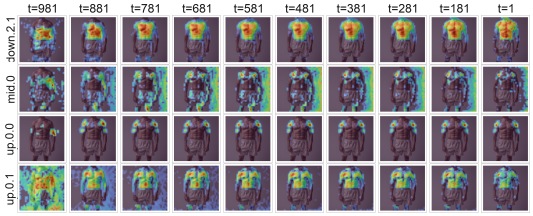


Figure 4: Nudity-localized activations concentrate on exposed body regions, with their centers appearing lower in `down.2.1` and higher in the up blocks.

SharedSAE achieves an average FVU of 0.400 across four blocks, explaining approximately 60% of the residual variance. The block-wise independent SAEs trained under the native SAE-Turbo setting (Surkov et al., 2025) reach 0.492 on the same prompts and the same dictionary width, so the shared dictionary explains more of the residual variance while covering four blocks and the full trajectory.

4.2 Model Cost

Table 4 compares the cost of covering four U-Net blocks with a dictionary width of 10,240. A block-wise approach trains four independent SAEs, totaling 104.9M parameters, whereas SharedSAE uses a single shared encoder-decoder with lightweight conditioning to cover all blocks and timesteps with only 28.1M parameters, reducing the parameter footprint by 73.3%. By learning a shared dictionary across blocks and timesteps, SharedSAE also avoids repeated feature filtering, concept localization,

and SAE retraining, enabling the same features to be reused across all covered blocks and timesteps for erasure.

4.3 Localization Stability

Concept localization produces a ranked feature set for each target concept and block, which is then used for intervention. We evaluate ranking stability on eight object concepts under prompt split, random seed variation, and time-window shift, using the Jaccard overlap between the resulting top-k_{loc} sets. Prompt-split stability compares rankings from two random halves of the controlled prompt pairs, cross-seed stability uses the same prompts with different random seeds, and time-window stability shifts the localization window.

SharedSAE remains stable under prompt splitting and time-window shifts, but is more sensitive to random seed variation than SAE-Turbo. This difference likely arises because SharedSAE localizes features over a 50-step trajectory, allowing seed-dependent stochastic variation to accumulate across the denoising path, whereas SAE-Turbo uses a single denoising step. We therefore average localization statistics over multiple seeds in all experiments.

4.4 Architecture Ablations

We isolate the roles of the block adapter and the temporal conditioning module by retraining SharedSAE with each component removed. Table 5 reports a component-specific representation diagnostic, while Figure 6(a) shows how the two ablations affect concept erasure. In the horse example, removing the block adapter leaves the target visibly recognizable, indicating incomplete suppression. Removing temporal conditioning suppresses the target more strongly but introduces a conspicuous residual artifact. The full SharedSAE model removes the horse while producing a more coherent replacement. The train example is less challenging, and all three variants suppress the target under the same prompt and seed.

Table 5: Architecture ablations of SharedSAE.

Variant	Diagnostic	Ablated	SharedSAE
w/o block adapter	effective rank ↑	187	**433**
w/o temporal conditioning	time-window Jaccard ↑	0.720	**0.735**

Removing the block adapter collapses the effective rank of the shared decoder, from 433 to 187. Effective rank is the participation ratio of the singular-value spectrum of the normalized decoder directions. It counts how many independent directions the $10{,}240$ columns actually span, so a lower value means the learned directions have become mutually redundant. The block adapter therefore keeps the dictionary high-rank rather than improving the fit.

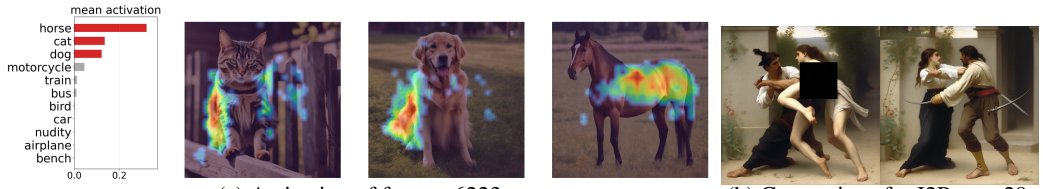
Removing temporal conditioning lowers time-window stability, from 0.735 to 0.720. The gap is modest and indicates that the temporal component contributes to how consistently a concept's features are localized across denoising steps.

4.5 Concept Erasure

General Concept Erasure. Figure 2 shows erasure results for 16 concepts. The same intervention suppresses the target for most of these concepts.

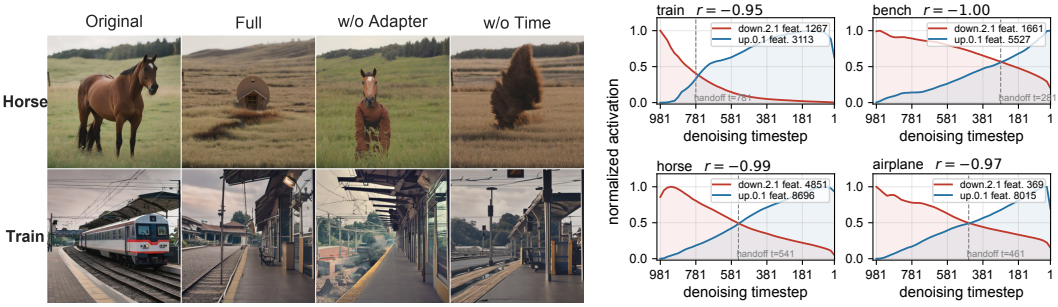
Nudity Erasure. Following the evaluation setting of SPACE (Novello & Tonello, 2026), we evaluate SharedSAE on the I2P-142 nudity subset to align with previous work. SharedSAE substantially reduces NudeNet detections, as shown in Table 1. Under our detector-positive protocol over the full I2P set, the detector-positive subset contains 139 cases, of which only 1 remains NudeNet-positive after SharedSAE intervention. Figure 4 shows that the localized features concentrate on exposed body regions, enabling spatially focused suppression. Figure 5(b) presents the sole residual detection, which comes from a visible foot that remains after erasure.

Style Erasure. Unlike object concepts with identifiable subjects, global styles are encoded by distributed visual attributes such as color, illumination, texture, contour, and scene composition, making them substantially harder to erase. We evaluate five representative styles: cyberpunk, Monet, ukiyo-e, Van Gogh, and watercolor. As shown in Figure 7(b), increasing the per-block feature set from 10 to 50 generally weakens the target style, but does not fully remove it or consistently restore a neutral photographic appearance. This difficulty is particularly evident for cyberpunk, whose characteristic features, including strong lighting effects, neon colors, reflections, and scene-wide illumination patterns, are distributed across signs, roads, building facades, and the distant streetscape. Consequently,



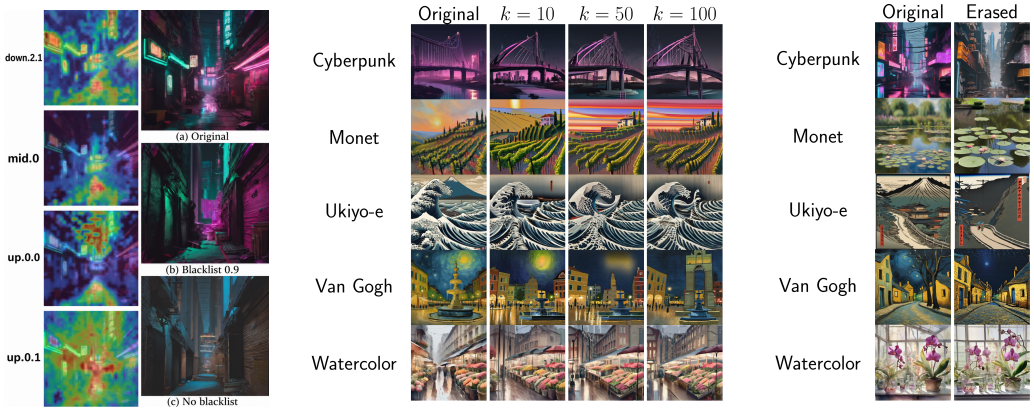
(a) Activation of feature 6223. (b) Comparison for I2P case 29.

Figure 5: (a) Feature 6223 responds strongly to horse, cat, and dog, with spatial activation concentrated on the quadruped body. (b) I2P case 29 retains a `FEET_EXPOSED` detection after erasure.



(a) Architecture-ablation erasure results. (b) Cross-block temporal succession.

Figure 6: (a) Full refers to the complete SharedSAE model. Removing the block adapter can leave the target recognizable, whereas removing temporal conditioning can introduce residual artifacts. (b) Representative `down.2.1` features decay while `up.0.1` features rise.



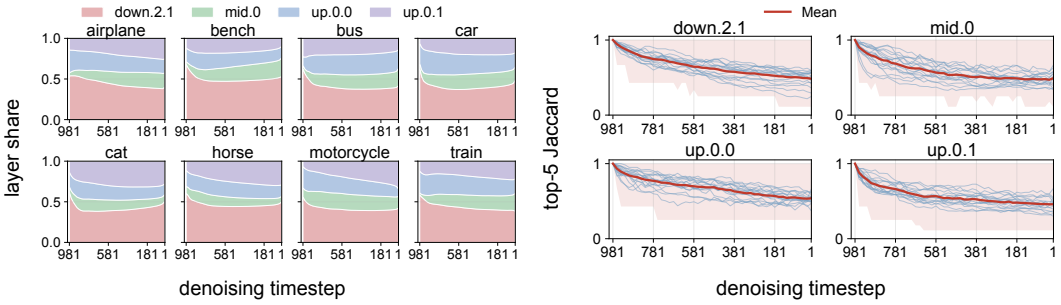
(a) Cyberpunk style. (b) Cross-style erasure as k_{loc} increases. (c) Successful examples.

Figure 7: Style erasure. (a) Cyberpunk block-wise heatmaps accompany the original, blacklist-0.9, and no-blacklist outputs. (b) Increasing the number of features used per block, k_{loc}, from 10 to 50 attenuates each style without removing it. (c) The prompts used for concept localization erase the target style.

many highly activated features are spatially dispersed and can be readily filtered out by the blacklist. Figure 7(a) illustrates different erasure effects under different blacklist settings.

Figure 7(c) shows the erasure results obtained by directly using the prompts employed for concept localization. Every evaluated style is attenuated under this setting. Cyberpunk and Monet are transformed into neutral appearances, the watercolor washes largely disappear, the flat color planes and bold outlines of ukiyo-e weaken visibly, and the characteristic visual patterns of Van Gogh are mostly gone. Together with the observations in Figure 7(a), these results show that style concepts are erasable and that the main difficulty lies in accurately localizing the relevant features. Style concepts are less concrete than object concepts and more broadly distributed across visual features, so identifying a compact and precise set of features for a target style remains the primary challenge.

Concept Feature Sharing. Figure 5(a) illustrates how a single SharedSAE feature can recur across semantically related concepts. Among the features localized at `mid.0`, feature 6223 ranks in the top-k_{loc} for cat, dog, and horse, while its mean activation on the remaining concepts is one to two orders of magnitude lower. Its spatial activation consistently follows the animal body despite



(a) Block-wise shares of concept activation.

(b) Feature-set succession within each block.

Figure 8: Spatiotemporal evolution of concept-representative features. (a) Relative activation shares generally shift from `down.2.1` toward `mid.0` or `up.0.1`. (b) Top-5 feature overlap with the initial timestep decreases throughout denoising; thick lines and shaded regions denote the mean and 95% interval over 340 cases.

changes in species, pose, and appearance, indicating that the feature captures visual structure shared across quadruped concepts rather than a single object class. Suppressing this shared feature can therefore influence several related concepts, linking semantic reuse in the dictionary to cross-concept coupling during intervention.

This feature direction sharing also manifests as concept compensation and leakage during erasure. Suppressing features associated with a horse or motorcycle can redirect the representation toward a nearby concept direction, such as a person. Because the coarse spatial structure of the target is established early in denoising and tends to persist throughout the trajectory, the replacement often inherits the original shape or spatial layout, even when it belongs to a different semantic category. Moreover, erasing a green truck removes the truck while retaining the green attribute. Green leaks into the background image and is automatically added to its original position. This suggests that object identity and attribute are partially disentangled in a shared sparse direction.

Spatiotemporal Feature Succession. Figure 8(b) shows that concept-representative feature sets change throughout denoising. The mean top-5 overlap in `down.2.1` decreases from 0.746 to 0.485, with the same downward trend across all four blocks. Raw activation magnitudes also carry a strong temporal trend, and target-feature activations in `down.2.1` consistently decay from early to late timesteps. We therefore use relative block shares to factor out global magnitude changes and isolate redistribution across blocks. Figure 8(a) shows that although `down.2.1` remains the dominant contributor for most concepts, its relative share generally decreases over time, while `mid.0` and `up.0.1` make increasingly larger contributions.

This pattern suggests that concept representations are progressively redistributed from earlier downsampling blocks toward intermediate and later stages of the U-Net as denoising proceeds.

This succession is also visible at the level of individual features. Figure 6(b) shows highly ranked `down.2.1` and `up.0.1` features for train, bench, horse, and airplane. The `down.2.1` features activate early and then decay, whereas the `up.0.1` features emerge later. Their trajectories cross during denoising, showing the succession of representative features across blocks.

5 Conclusion

SharedSAE shows that sparse analysis of diffusion models need not be fragmented across network depth and denoising time. It learns a single dictionary shared across four U-Net blocks and the full denoising trajectory, supporting reconstruction, localization, and intervention throughout generation. This shared space is not only more compact than maintaining separate block-wise SAEs, but also makes it possible to follow how semantic representations evolve: concept-related features are reused across related concepts and progressively redistributed from early downsampling blocks toward intermediate and later blocks as denoising proceeds. The same representation supports effective intervention, including a reduction from 139 NudeNet-positive generations to 1 in our detector-positive I2P subset. Taken together, our results suggest that a shared sparse vocabulary can provide a fixed reference frame for studying representations that themselves evolve across network depth and denoising time.

AI USE STATEMENT

In this work, we did not use generative AI tools for any task requiring disclosure under the ICLR AI policy. Specifically, generative AI was not used to create synthetic datasets; develop theoretical models or conceptual frameworks; derive mathematical statements; supply key components of or aid in constructing mathematical proofs; formulate or refine research hypotheses; design, select, or evaluate experimental methodology; implement our proposed algorithm; translate research materials; clean or restructure datasets; conduct qualitative or thematic analysis; or interpret experimental findings.

Generative AI tools were used for tasks for which disclosure is recommended: drafting and revising manuscript prose to improve readability, suggesting structural improvements to the paper, editing source code, and polishing the presentation and LaTeX typesetting of mathematical derivations. Generative AI did not generate the core mathematical claims or proofs.

All AI-assisted outputs were reviewed by the authors. AI-modified code was inspected and tested for correctness. AI-edited manuscript text was cross-checked against our original experimental results, mathematical derivations, and intended claims, to ensure no unsubstantiated assertions were introduced and that the interpretation of our findings remained unchanged. The authors take full responsibility for all final content of this work, including text, claims, and artifacts produced with generative AI assistance.

REFERENCES

Enrico Cassano, Riccardo Renzulli, Marco Nurisso, Mirko Zaffaroni, Alan Perotti, and Marco Grangetto. SAEmnesia: Erasing concepts in diffusion models with supervised sparse autoencoders. In *Forty-Third International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2509.21379.

Bartosz Cywiński and Kamil Deja. SAeUron: Interpretable concept unlearning in diffusion models with sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 11738–11775. PMLR, 2025. URL https://proceedings.mlr.press/v267/cywinski25a.html.

Tatiana Gaintseva, Andreea-Maria Oncescu, Chengcheng Ma, Ziquan Liu, Martin Benning, Gregory Slabaugh, Jiankang Deng, and Ismail Elezi. CASteer: Cross-attention steering for controllable concept erasure. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/435422305988b73c6cc00bcb29ba2531-Abstract-Conference.html.

Rohit Gandikota, Joanna Materzyńska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2426–2436, 2023. doi: 10.1109/ICCV51070.2023.00230. URL https://doi.org/10.1109/ICCV51070.2023.00230.

Rohit Gandikota, Hadas Orgad, Yonatan Belinkov, Joanna Materzyńska, and David Bau. Unified concept editing in diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5111–5120, 2024. doi: 10.1109/WACV57701.2024.00503. URL https://doi.org/10.1109/WACV57701.2024.00503.

Chao Gong, Kai Chen, Zhipeng Wei, Jingjing Chen, and Yu-Gang Jiang. Reliable and efficient concept erasure of text-to-image diffusion models. In *Computer Vision – ECCV 2024*, pp. 73–88, 2024. doi: 10.1007/978-3-031-73668-1_5. URL https://doi.org/10.1007/978-3-031-73668-1_5.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020. URL https://papers.nips.cc/paper_files/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Victor Shea-Jay Huang, Le Zhuo, Yi Xin, Zhaokai Wang, Fu-Yun Wang, Yuchi Wang, Renrui Zhang, Peng Gao, and Hongsheng Li. TIDE: Temporal-aware sparse autoencoders for

interpretable diffusion transformers in image generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(1):435–443, 2026. doi: 10.1609/aaai.v40i1.37006. URL https://doi.org/10.1609/aaai.v40i1.37006.

Man Jiang, Ouxiang Li, Weibao Xue, Zhenhua Tang, Yuan Wang, Shuo Wang, and Yanbin Hao. PEAK: Precise and persistent concept erasure via k-sparse autoencoders, 2026. URL https://arxiv.org/abs/2608.10985. arXiv preprint.

Byung Hyun Lee, Sungjin Lim, and Se Young Chun. Localized concept erasure for text-to-image diffusion models using training-free gated low-rank adaptation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18596–18606, 2025. doi: 10.1109/CVPR52734.2025.01733. URL https://doi.org/10.1109/CVPR52734.2025.01733.

Ouxiang Li, Yuan Wang, Xinting Hu, Houcheng Jiang, Tao Liang, Yanbin Hao, Guojun Ma, and Fuli Feng. SPEED: Scalable, precise, and efficient concept erasure for diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=aoEtzdRkGh.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Shilin Lu, Zilan Wang, Leyang Li, Yanzhu Liu, and Adams Wai-Kin Kong. MACE: Mass concept erasure in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6430–6440, 2024. doi: 10.1109/CVPR52733.2024.00615. URL https://doi.org/10.1109/CVPR52733.2024.00615.

Nicola Novello and Andrea M. Tonello. Empty SPACE: Cross-attention sparsity for concept erasure in diffusion models, 2026. URL https://arxiv.org/abs/2605.10198. arXiv preprint.

Bedapudi Praneeth. NudeNet: Lightweight nudity detection. Python package, version 3.4.2, 2024. URL https://pypi.org/project/nudenet/. Version 3.4.2.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022. doi: 10.1109/CVPR52688.2022.01042. URL https://doi.org/10.1109/CVPR52688.2022.01042.

Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22522–22531, 2023. doi: 10.1109/CVPR52729.2023.02157. URL https://doi.org/10.1109/CVPR52729.2023.02157.

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.

Viacheslav Surkov, Chris Wendler, Antonio Mari, Mikhail Terekhov, Justin Deschenaux, Robert West, Caglar Gulcehre, and David Bau. One-step is enough: Sparse autoencoders for text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3342. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/90291c8d685db65a8a1e32fd2b66f8bb-Abstract-Conference.html.

Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and evolution of interpretable concepts in diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5566. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/f3d1dab3b68abb7bb2af5cae28def3b0-Abstract-Conference.html.

Yuan Wang, Ouxiang Li, Tingting Mu, Yanbin Hao, Kuien Liu, Xiang Wang, and Xiangnan He. Precise, fast, and low-cost concept erasure in value space: Orthogonal complement matters. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28759–28768, 2025. doi: 10.1109/CVPR52734.2025.02678. URL https://doi.org/10.1109/CVPR52734.2025.02678.

Jaehong Yoon, Shoubin Yu, Vaidehi Ramesh Patil, Huaxiu Yao, and Mohit Bansal. SAFREE: Training-free and adaptive guard for safe text-to-image and video generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8de5384f522efff26884559599c09312-Abstract-Conference.html.

A Sparse Vector-Field Interpretation of SharedSAE

SharedSAE acts on the block residuals $x = h^{\text{out}} - h^{\text{in}}$ of the epsilon-predicting SDXL U-Net. Although SDXL does not directly regress an ODE velocity, an epsilon prediction induces a score estimate, and hence a probability-flow ODE associated with the same diffusion probability path. This section uses that standard correspondence only as a bridge from hidden-state perturbations to a local generative vector field. The SharedSAE-specific statement is then that a decoder direction is pushed forward through the downstream U-Net Jacobian into a local velocity direction, and that the final image-space effect is the accumulated transport of these local perturbations along the denoising trajectory.

A.1 From Epsilon Prediction to a Probability-Flow Velocity

Let the Gaussian diffusion path be

$$y_t = a_t y_0 + s_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \tag{11}$$

where $a_t > 0$ and $s_t > 0$ are differentiable signal and noise schedules. A continuous-time linear SDE that realizes these Gaussian conditionals can be written as

$$\mathrm{d}y = f_t y \,\mathrm{d}t + g_t \,\mathrm{d}W_t, \qquad f_t = \frac{\dot{a}_t}{a_t}, \tag{12}$$

with

$$g_t^2 = \frac{\mathrm{d}}{\mathrm{d}t} s_t^2 - 2 f_t s_t^2. \tag{13}$$

Under the usual epsilon-prediction parameterization, the exact conditional score satisfies

$$\nabla_y \log p_t(y_t \mid c) = -\frac{\mathbb{E}[\epsilon \mid y_t, c]}{s_t}. \tag{14}$$

A trained denoiser approximates the conditional noise expectation, so we define

$$s_\theta(y_t, t, c) := -\frac{\epsilon_\theta(y_t, t, c)}{s_t} \approx \nabla_y \log p_t(y_t \mid c). \tag{15}$$

The probability-flow ODE associated with the forward SDE is

$$\frac{\mathrm{d}y}{\mathrm{d}t} = v_{\text{pf}}(y, t, c) = f_t y - \frac{1}{2} g_t^2 s_\theta(y, t, c). \tag{16}$$

Substituting Eq. (15) gives

$$v_{\text{pf}}(y, t, c) = f_t y + \kappa_t \epsilon_\theta(y, t, c), \qquad \kappa_t = \frac{g_t^2}{2 s_t}. \tag{17}$$

Equivalently, using Eq. (13),

$$\kappa_t = \dot{s}_t - \frac{\dot{a}_t}{a_t} s_t. \tag{18}$$

Effective prediction under classifier-free guidance. The experiments use the prediction actually consumed by the sampler. To cover both conditional sampling and classifier-free guidance (CFG), let

$$\widehat{\epsilon}_\theta(y_t, t, c) \tag{19}$$

denote the *effective* noise prediction passed to the scheduler. Without CFG, $\widehat{\epsilon}_\theta = \epsilon_\theta(\cdot \mid c)$. With guidance scale w,

$$\widehat{\epsilon}_\theta = \epsilon_\theta(\cdot \mid \varnothing) + w\left[\epsilon_\theta(\cdot \mid c) - \epsilon_\theta(\cdot \mid \varnothing)\right]. \tag{20}$$

For guided sampling, Eq. (17) should therefore be interpreted as the effective denoising vector field used by the sampler, rather than as a claim that CFG exactly equals the probability-flow ODE of the original conditional density. The same derivation, however, applies whether the intervention modifies the conditional branch, the unconditional branch, or both.

Standard marginal-equivalence fact. If $s_\theta = \nabla_y \log p_t$ is exact, then the probability-flow ODE in Eq. (16) and the corresponding reverse-time SDE share the same one-time marginals p_t (Song et al., 2021). We use this standard result only to justify the velocity-field interpretation; the propositions below concern the SharedSAE intervention itself.

A.2 SharedSAE Features as Pushed-Forward Velocity Directions

For block b and denoising time t, let $x_{b,t}$ be the residual update. Following Eq. (2) in the main text, we define the per-coordinate residual scale

$$\sigma_b = \frac{\max(\mu_b, \epsilon)}{\sqrt{d}}, \qquad \widetilde{x}_{b,t} = \frac{x_{b,t}}{\sigma_b}. \tag{21}$$

Therefore, a vector represented in the normalized SharedSAE residual space is mapped back to the original U-Net residual space by multiplication with σ_b.

Consistent with Eq. (4) in the main text, SharedSAE reconstructs the normalized residual as

$$\widetilde{x}_{b,t} \approx D z_{b,t} + \beta_x, \tag{22}$$

and hence

$$x_{b,t} \approx \sigma_b\left(D z_{b,t} + \beta_x\right) = \sigma_b \sum_{j=1}^{m} z_{j,b,t} D_j + \sigma_b \beta_x. \tag{23}$$

As in the main text, let $h^{\mathrm{out}}_{b,t}$ denote the block output activation. For the Jacobian calculation we regard $h^{\mathrm{out}}_{b,t}$ as the vectorized block state and suppress the spatial-token index i for readability; equivalently, each D_j is injected at the corresponding token coordinates. Holding the upstream state fixed locally, we write the downstream map from the block output to the effective sampler prediction as

$$\widehat{\epsilon}_\theta(y_t, t, c) = G_{b,t,c}\big(h^{\mathrm{out}}_{b,t}\big), \tag{24}$$

with Jacobian

$$J_{b,t,c} = \frac{\partial G_{b,t,c}(h^{\mathrm{out}}_{b,t})}{\partial h^{\mathrm{out}}_{b,t}}. \tag{25}$$

Proposition A.1 (Sparse velocity pushforward)**.** *Assume $G_{b,t,c}$ is twice continuously differentiable in a neighborhood of the current state. For a sufficiently small sparse-code perturbation Δz applied at block b and time t, the induced change in the effective probability-flow velocity satisfies*

$$\Delta v_{\mathrm{pf}} = \sum_{j=1}^{m} \Delta z_j \psi^{\mathrm{ODE}}_{j,b,t,c} + \mathcal{O}(\|\Delta z\|_2^2), \tag{26}$$

where

$$\psi^{\mathrm{ODE}}_{j,b,t,c} = \kappa_t \sigma_b J_{b,t,c} D_j. \tag{27}$$

For the reverse-time SDE, the feature-induced drift perturbation *is collinear with the ODE perturbation and has twice its score-dependent coefficient:*

$$\Delta b_{\mathrm{SDE}} = 2\,\Delta v_{\mathrm{pf}} + \mathcal{O}(\|\Delta z\|_2^2). \tag{28}$$

This statement concerns the perturbations; the full reverse-SDE drift and full probability-flow velocity need not themselves be collinear because both also contain the forward drift term.

Proof. A sparse-code perturbation produces a hidden-state displacement

$$\Delta h_{b,t}^{\text{out}} = \sigma_b D \Delta z. \tag{29}$$

A first-order Taylor expansion of the downstream U-Net gives

$$\Delta \widehat{\epsilon}_\theta = J_{b,t,c} \Delta h_{b,t}^{\text{out}} + \mathcal{O}(\|\Delta h_{b,t}^{\text{out}}\|_2^2). \tag{30}$$

For a finite hidden-state displacement, the fundamental theorem of calculus gives the exact identity

$$G_{b,t,c}(h + \Delta h) - G_{b,t,c}(h) = \bar{J}_{b,t,c}(h, \Delta h)\Delta h, \tag{31}$$

where

$$\bar{J}_{b,t,c}(h, \Delta h) = \int_0^1 J_{G_{b,t,c}}(h + s\Delta h)\, \mathrm{d}s. \tag{32}$$

Thus Eq. (30) replaces the path-averaged Jacobian by its value at the unperturbed state. Its accuracy is determined by the Jacobian variation along the displacement path. By Eq. (17), perturbations of the effective noise prediction induce

$$\Delta v_{\text{pf}} = \kappa_t \Delta \widehat{\epsilon}_\theta. \tag{33}$$

Substituting Eqs. (29) and (30) yields

$$\Delta v_{\text{pf}} = \kappa_t \sigma_b J_{b,t,c} D \Delta z + \mathcal{O}(\|\Delta z\|_2^2), \tag{34}$$

which gives Eqs. (26)–(27) after expanding over decoder columns.

For the reverse-time SDE, the score term has coefficient g_t^2 rather than $g_t^2/2$. Therefore, its perturbation induced by the same score change has twice the coefficient of the probability-flow ODE perturbation, proving Eq. (28). □

Feature identity versus feature effect. The shared decoder column D_j supplies a common sparse coordinate across blocks and timesteps, whereas its local generative effect is

$$\psi_{j,b,t,c}^{\text{ODE}} = \kappa_t \sigma_b J_{b,t,c} D_j, \tag{35}$$

which remains context dependent through the downstream Jacobian and the diffusion schedule. Thus SharedSAE shares *feature coordinates*, not a single context-invariant velocity vector.

A.3 Concept Erasure as a Sparse Velocity Perturbation

Let $\mathcal{F}_{c,b}$ be the set of features localized for concept c at block b, and let

$$q_j(b,t) = g_j w_j(b,t) \tag{36}$$

combine the rank and temporal weights. The main text denotes the global erasure strength by λ_{erase} in Eq. (9), with $\lambda_{\text{erase}} = 10$ for general concept erasure and $\lambda_{\text{erase}} = 5$ for I2P nudity erasure. We write λ for this fixed erasure strength below. The adapter scale α_A from Eq. (3) is a separate parameter and does not enter the analysis. The normalized-space concept direction is

$$r_c(b,t) = \sum_{j \in \mathcal{F}_{c,b}} q_j(b,t)\, z_{j,b,t} D_j. \tag{37}$$

Consistent with Eq. (21) and the main-text intervention in Eq. (9), the corresponding U-Net-space displacement is

$$\Delta h_{b,t}^{\text{out}} = -\lambda \sigma_b r_c(b,t). \tag{38}$$

The resulting first-order velocity change is therefore

$$\Delta v_{\text{pf}} \approx -\lambda \sum_{j \in \mathcal{F}_{c,b}} q_j(b,t) z_{j,b,t} \psi_{j,b,t,c}^{\text{ODE}}. \tag{39}$$

The implementation additionally caps the mean absolute per-step perturbation ratio at $\delta_{\max} = 0.2$, as stated in the main text. The local analysis above describes the unclipped branch of the intervention; when the cap is active, the same pushforward applies to the clipped displacement away from the clipping boundary.

We define the *localized feature-induced velocity subspace*

$$\mathcal{V}_c(b,t) = \operatorname{span}\left\{\psi^{\mathrm{ODE}}_{j,b,t,c} \;\middle|\; j \in \mathcal{F}_{c,b}\right\}. \tag{40}$$

Equation (39) shows that the intervention subtracts a state-dependent vector in this empirically localized subspace. This does not assert that $\mathcal{V}_c$ is the unique or complete semantic subspace of concept c; rather, it is the subspace induced by the features selected by the localization procedure.

Multiple intervened blocks. To separate the fixed erasure strength from the expansion parameter, define the complete block-wise displacement $u_b = -\lambda\sigma_b r_c(b,t)$ and introduce the auxiliary path

$$\Delta h_b(\varepsilon) = \varepsilon u_b, \qquad \varepsilon \in [0,1]. \tag{41}$$

Here $\varepsilon = 0$ is the unperturbed model, and $\varepsilon = 1$ corresponds to the complete intervention used in the experiments. A multivariate Taylor expansion at $\varepsilon = 0$ gives

$$\Delta\widehat{\epsilon}_\theta(\varepsilon) = \varepsilon \sum_{b\in\mathcal{B}} J_{b,t,c} u_b + \mathcal{O}(\varepsilon^2). \tag{42}$$

For two successive blocks, an upstream intervention changes the later sparse code according to $z_2(\varepsilon) = z_2(0) + \varepsilon \dot{z}_2(0) + \mathcal{O}(\varepsilon^2)$. The later intervention contains another factor of ε, so the induced cross-block term is $\mathcal{O}(\varepsilon^2)$, not $\mathcal{O}(\lambda^2)$. Equation (42) is local at $\varepsilon = 0$. At $\varepsilon = 1$, its accuracy is determined by nonlinear interactions along the complete intervention path.

Local magnitude bound. Let $D_{\mathcal{F}_{c,b}}$ denote the decoder submatrix containing the localized columns and let $a_{b,t}$ be the coefficient vector with entries $a_j = q_j(b,t) z_{j,b,t}$. Then

$$\|\Delta v_{\mathrm{pf}}\|_2 \lesssim \lambda |\kappa_t| \sigma_b \|J_{b,t,c}\|_2 \|D_{\mathcal{F}_{c,b}}\|_2 \|a_{b,t}\|_2. \tag{43}$$

Since each decoder column is unit norm, a looser but simpler bound is

$$\|\Delta v_{\mathrm{pf}}\|_2 \lesssim \lambda |\kappa_t| \sigma_b \|J_{b,t,c}\|_2 \sum_{j\in\mathcal{F}_{c,b}} |q_j(b,t) z_{j,b,t}|. \tag{44}$$

Because both the residual and the intervention are mapped from normalized space by the same residual scale σ_b, block normalization preserves the relative intervention scale before the downstream Jacobian acts:

$$\frac{\|\Delta h^{\mathrm{out}}_{b,t}\|_2}{\|x_{b,t}\|_2} = \lambda \frac{\left\|\sum_{j\in\mathcal{F}_{c,b}} q_j(b,t) z_{j,b,t} D_j\right\|_2}{\|\widetilde{x}_{b,t}\|_2}. \tag{45}$$

The residual scale σ_b cancels, so the same normalized intervention has a comparable relative meaning across blocks before block-specific downstream dynamics are applied.

A.4 Accumulation Along the Denoising Trajectory

The continuous-time analysis below is an idealization of the discrete sampler used in the experiments. Let $\tau \in [0,1]$ increase from noise to data and absorb the reversal of the original diffusion time into an effective vector field $\bar{v}$:

$$\frac{\mathrm{d}y}{\mathrm{d}\tau} = \bar{v}(y,\tau,c), \qquad y(0) = y_{\mathrm{noise}}. \tag{46}$$

Write the intervened dynamics as

$$\frac{\mathrm{d}y^\varepsilon}{\mathrm{d}\tau} = \bar{v}(y^\varepsilon,\tau,c) + \varepsilon u(y^\varepsilon,\tau,c), \tag{47}$$

where u collects the complete pushed-forward contributions at the fixed strength λ.

Define

$$A(\tau) = J_y \bar{v}(y_\tau, \tau, c) \tag{48}$$

and the state-transition matrix $\Phi(\tau, s)$ by

$$\frac{\partial \Phi(\tau, s)}{\partial \tau} = A(\tau)\Phi(\tau, s), \qquad \Phi(s, s) = I. \tag{49}$$

Theorem A.2 (First-order trajectory displacement)**.** *Assume $\bar{v}$ and u are twice continuously differentiable in a neighborhood of the unperturbed trajectory and that the solution remains in this neighborhood for sufficiently small ε. Then*

$$y^\varepsilon(1) - y(1) = \varepsilon \int_0^1 \Phi(1, \tau) u(y_\tau, \tau, c)\, \mathrm{d}\tau + \mathcal{O}(\varepsilon^2). \tag{50}$$

For SharedSAE,

$$u(y_\tau, \tau, c) = -\lambda \sum_{b \in \mathcal{B}} \sum_{j \in \mathcal{F}_{c,b}} q_j(b, t(\tau)) z_{j,b,t(\tau)} \bar{\psi}_{j,b,t(\tau),c}, \tag{51}$$

where $\bar{\psi}$ denotes the local pushed-forward direction after the change of clock from t to τ. Hence

$$\begin{aligned} y^\varepsilon(1) - y(1) \approx -\varepsilon\lambda \sum_{b \in \mathcal{B}} \sum_{j \in \mathcal{F}_{c,b}} \int_0^1 \Phi(1, \tau)\, q_j(b, t(\tau))\, z_{j,b,t(\tau)} \\ \cdot \bar{\psi}_{j,b,t(\tau),c}\, \mathrm{d}\tau. \end{aligned} \tag{52}$$

Proof. Let

$$\xi(\tau) = \left. \frac{\partial y^\varepsilon(\tau)}{\partial \varepsilon} \right|_{\varepsilon=0}. \tag{53}$$

Differentiating Eq. (47) with respect to ε at zero gives the first-variation equation

$$\frac{\mathrm{d}\xi}{\mathrm{d}\tau} = A(\tau)\xi + u(y_\tau, \tau, c), \qquad \xi(0) = 0. \tag{54}$$

Variation of constants yields

$$\xi(1) = \int_0^1 \Phi(1, \tau) u(y_\tau, \tau, c)\, \mathrm{d}\tau. \tag{55}$$

A second-order Taylor expansion in ε then gives

$$y^\varepsilon(1) - y(1) = \varepsilon\xi(1) + \mathcal{O}(\varepsilon^2), \tag{56}$$

proving Eq. (50). Substituting the sum of SharedSAE pushed-forward feature contributions from Eq. (39) and Eq. (42) gives Eq. (52). □

A.5 Discrete-Sampler Counterpart

The experiments use a finite-step sampler, so it is useful to state the discrete analogue explicitly. Let one unperturbed sampling step be

$$y_{k+1} = F_k(y_k), \qquad k = 0, \ldots, N-1, \tag{57}$$

and let $U_k(y_k)$ be the complete SharedSAE intervention at step k, including the fixed strength λ:

$$y^\varepsilon_{k+1} = F_k(y^\varepsilon_k) + \varepsilon U_k(y^\varepsilon_k) + \mathcal{O}(\varepsilon^2). \tag{58}$$

Define the discrete transition matrix

$$\Phi_{N,k+1} = JF_{N-1}(y_{N-1}) \cdots JF_{k+1}(y_{k+1}), \tag{59}$$

with $\Phi_{N,N} = I$.

Corollary A.3 (First-order displacement for a discrete sampler)**.** *Under the same local smoothness assumptions,*

$$y_N^\varepsilon - y_N = \varepsilon \sum_{k=0}^{N-1} \Phi_{N,k+1} U_k(y_k) + \mathcal{O}(\varepsilon^2). \tag{60}$$

Proof. Let $\eta_k = y_k^\varepsilon - y_k$. Linearizing Eq. (58) around y_k gives

$$\eta_{k+1} = JF_k(y_k)\eta_k + \varepsilon U_k(y_k) + \mathcal{O}(\varepsilon^2), \qquad \eta_0 = 0, \tag{61}$$

where $\eta_k = \mathcal{O}(\varepsilon)$. Repeated substitution yields Eq. (60). □

Equation (60) is the finite-step counterpart of Eq. (50): a local sparse intervention is first pushed through the remaining U-Net computation at its current step, then propagated through all subsequent denoising steps to the final sample.

Interpretation. The theory separates three objects:

$$\boxed{D_j \quad \longrightarrow \quad \psi_{j,b,t,c} = \kappa_t \sigma_b J_{b,t,c} D_j \quad \longrightarrow \quad \Phi\, \psi_{j,b,t,c}.} \tag{62}$$

The first is the shared sparse feature coordinate, the second is its context-dependent local effect on the denoising vector field, and the third is the effect after transport to the final sample. SharedSAE therefore does not assume that one decoder column corresponds to a globally fixed generative direction. Instead, it supplies a reusable hidden-space coordinate system whose local causal effect is determined by the current block, timestep, state, conditioning, and downstream network Jacobian.