ELROND: EXPLORING AND DECOMPOSING INTRINSIC CAPABILITIES OF DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A single text prompt passed to a diffusion model yields a wide range of visual outputs determined solely by a stochastic process, leaving users with no direct control over which semantic variations appear. Exploring this range is difficult: random search offers no guarantee of covering it, while prompt editing is coarse, as even a small change in wording can substantially alter the generated image. We argue that systematic exploration instead requires recovering how the model itself organizes the conditional distributions it can produce. We formalize this structure as a *generative manifold*, and present ELROND, a method for recovering its tangent space at a given conditioning. To that end, we collect gradients obtained by backpropagating the differences between stochastic realizations of a fixed prompt, and decompose them into interpretable directions using Principal Component Analysis or a Sparse Autoencoder. We show that our method recovers this subspace accurately in a controlled setting and validate it behaviorally on large-scale models. We also demonstrate that recovered structure enables broader exploration of the model's capabilities than methods relying on external representations, and mitigates mode collapse in distilled models without retraining.

1 INTRODUCTION

Text-to-image diffusion models (Rombach et al., 2022; Podell et al., 2023) can generate a remarkably diverse set of images from a single prompt. A prompt such as *"An image of an elf"* corresponds to anything from a Christmas creature to a majestic, immortal being, with the realized image determined entirely by the stochasticity. When a desired output is known in advance, existing methods can often narrow the generation through additional conditioning (Zhang et al., 2023), attention-based editing (Hertz et al., 2022), or learned attribute directions (Gandikota et al., 2024; Baumann et al., 2025). Creative exploration starts without such a target: users want to discover what the model can produce and then explore interesting variations further. Prompt engineering therefore offers limited help and random resampling provides no systematic way to cover the model's capabilities. What is missing is a structure organizing the variations the model can produce.

To see what such a structure should represent, consider the generation process itself. A diffusion model does not map a conditioning c directly to a single image but to a distribution $p_\theta(x \mid c)$, from which an individual image is sampled. Thus, conditioning determines *which distribution* is selected, while stochasticity determines *which realization* is sampled. For exploration, this distinction is important. Individual images are the objects we observe, but the distribution is the object the model controls. Exploring the model then means understanding how these distributions change as the conditioning changes. We therefore view the model through the family of conditional distributions it can realize (fig. 1). We refer to this family as the model's *generative manifold*, where each point is a distribution of images, and samples from a fixed conditioning are realizations from that point.

Access to this structure would make the exploration of a model's capabilities systematic allowing moving between distributions and, in principle, reaching every single one the model is able to produce. Although the manifold itself is not directly accessible, the conditioning space provides coordinates for it, since small changes in the conditioning induce local changes in the corresponding distribution.

Recovering semantic structure in such spaces is the established purpose of mechanistic interpretability tools, most notably sparse autoencoders and linear probes. Applying them here, however, exposes a fundamental limitation. These tools operate on activations gathered over diverse inputs and rely on

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

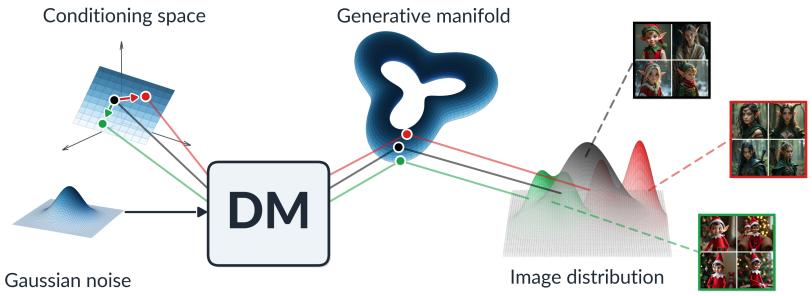

Figure 1: **Generative manifold of a diffusion model.** A conditioning c selects a conditional image distribution $p_\theta(x \mid c)$ on the *generative manifold*, while Gaussian noise determines the individual realization sampled from that distribution. Nearby conditionings induce local movements on the manifold. ELROND recovers its tangent space in the conditioning space.

the variance across them. Our setting admits none: the conditioning is fixed by construction, as the distribution available at a fixed prompt is exactly what we aim to characterize. Yet at a fixed prompt we have access to a different source of variation: stochasticity.

ELROND exploits this observation by extracting the local signal directly from visual variation within a single conditional distribution. Given two images generated from the same prompt under different random initializations, we construct a reconstruction objective that encourages the model's expected output to move from the characteristics of one realization toward those of the other. We then backpropagate this objective to the conditioning embedding, obtaining a direction in the conditioning space that corresponds to a local displacement on the generative manifold. Repeating this procedure over many pairs of samples provides a collection of local directions that span the tangent space at the distribution induced by the prompt.

We apply this construction to individual concepts by holding the remaining conditioning fixed and varying only the embedding of the token representing the concept. We then decompose the recovered directions spanning the corresponding tangent space using Principal Component Analysis (PCA) or Sparse Autoencoders (SAEs) (Olshausen & Field, 1997), obtaining interpretable directions that can be injected directly into the token embedding during sampling.

We validate our framework first in a controlled setting where the tangent space is known, showing that the recovered gradients recover the ground-truth subspace. On large-scale text-to-image models, we show that the model's sensitivity to its conditioning is concentrated in the recovered subspaces. We then demonstrate the practical value of the recovered structure for exploration, where our directions expose a wider range of concept-relevant variation while maintaining semantic consistency, and for mitigating mode collapse in distilled models. Our contributions can be summarized as follows:

- We introduce ELROND, a method for recovering and decomposing the tangent space of the generative manifold that a diffusion model has learned.
- We validate the recovery both against a known ground truth in a controlled setting and behaviorally on large-scale models, where we show that the recovered structure enables exploration of a model's capabilities beyond what existing approaches reach.
- We quantitatively and qualitatively evaluate how we can benefit from our approach in practical task of mitigating mode collapse in distilled models.

2 Related Work

Precise generation in text-to-image diffusion models Achieving desired control over generated images necessitates mechanisms that go beyond text prompting. Standard approaches introduce additional conditioning by extending the model through adapters (ControlNet (Zhang et al., 2023), T2I-Adapter (Mou et al., 2024)). Without affecting the structure of the model architecture, Rodriguez et al. (2024; 2025) have introduced a framework enabling fine-grained control by transporting the activations during inference. Alternatively, attention-based methods like Prompt-to-prompt (Hertz

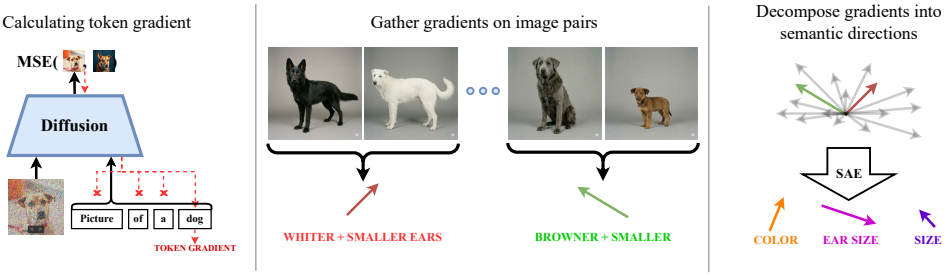

Figure 2: To extract the latent directions for a given concept, we backpropagate the differences between image instances generated from the same prompt to the given token. We then disentangle the latent directions by training SAE / computing PCA on the gradients.

et al., 2022) or Ledits++ (Brack et al., 2023) leverage cross-attention maps of a reference generation to localize changes, while Mokady et al. (2022) optimizes a text-conditioning space. Most relevant to our proposed methodology, Baumann et al. (2025) identifies semantic directions within the CLIP text embeddings by backpropagating the difference in noise predictions between contrastive prompts (e.g *"young person"* vs. *"old person"*). However, this reliance on manually constructing contrastive pairs necessitates *a priori* knowledge of the target attributes, rendering it infeasible for the unsupervised discovery of unknown semantic variations.

Unsupervised concept discovery Without supervision, Kwon et al. (2022) identify that U-net's bottleneck features possess a semantic structure, while Dalva & Yanardag (2024) employs contrastive learning on the directions added to the intermediate diffusion steps, to identify steerable directions. Similarly, Liu et al. (2023) decompose an image into a set of randomly initialized concepts that can sum up into the observed composition by the summation of individual score functions. On the other hand, building on the model adaptation techniques, Gandikota et al. (2024; 2025) automate the discovery process by applying PCA to CLIP embeddings of generated images, obtaining controllable basis vectors for a given concept without manual attribute specification. From a conceptually different direction, Dravid et al. (2024) interpret the semantic directions in the adapter's weight space.

Sparse Autoencoders in Diffusion Models Recent research has successfully adapted Sparse Autoencoders (SAE) to interpret text-to-image diffusion models. Works by Cywiński & Deja (2025); Surkov et al. (2024); Huang et al. (2025) demonstrate that SAEs can decompose activations of diffusion models into interpretable features, allowing for precise steering of the generative process. In parallel, some works apply SAEs to interpret CLIP embeddings (Pach et al., 2025; Dreyer et al., 2025) with additional applications when this model is used as conditioning for diffusion models (Kim et al., 2024; Ye et al., 2025), including precise editing as shown by Kamenetsky et al. (2025).

Geometry of learned representations The linear representation hypothesis holds that features are encoded as directions in activation space (Park et al., 2023), an assumption that underlies the use of sparse autoencoders for their recovery. Recent work argues that this picture is incomplete: Engels et al. (2025) identify irreducible multi-dimensional features, while Modell et al. (2025) develop a theory of features encoded as manifolds and show that cosine similarity in representation space reflects their intrinsic geometry.

3 METHOD

In this section, we introduce our framework. We first describe how to discover the distribution manifold by recovering its tangent space directly from sampled images and how to isolate this structure for individual concepts. We then explain how to decompose these recovered spaces into interpretable semantic directions, and how to use them for steering.

3.1 DISCOVERING THE MANIFOLD

A text-to-image diffusion model maps a text conditioning matrix $c \in \mathbb{R}^{L\times d}$, representing a sequence of L tokens, to an image probability distribution $p_\theta(x \mid c)$. Let

$$\mathcal{V} = \{c \in \mathbb{R}^{L\times d} \mid \mathbb{P}_{x\sim p_\theta(\cdot|c)}[x \in \mathcal{M}_{img}] \geq 1-\delta\}$$

be the set of conditionings that yield natural images, where $\mathcal{M}_{img}$ denotes the manifold of valid images. The family of distributions available to the model is then

$$\mathcal{M} = \{p_\theta(\cdot \mid c) \mid c \in \mathcal{V}\},$$

which, following the manifold hypothesis, we posit to be a low-dimensional manifold. Our goal is to characterize the neighbourhood of one of its points, that is, to recover the tangent space $T_{\Phi(c)}\mathcal{M}$ at the distribution induced by a fixed prompt, where $\Phi(c) = p_\theta(\cdot \mid c)$.

The points of $\mathcal{M}$ are distributions, which we cannot manipulate directly. We therefore use the parametrization Φ of $\mathcal{M}$ and work in the conditioning space instead. Since $T_{\Phi(c)}\mathcal{M} = \mathrm{im}(d\Phi_c)$ and $d\Phi_c$ is injective on $\ker(d\Phi_c)^\perp$, this subspace is in bijection with the tangent space. Recovering $T_{\Phi(c)}\mathcal{M}$ is thus equivalent to finding its representation in the conditioning space

$$\mathcal{T}_c = \ker(d\Phi_c)^\perp \subset \mathbb{R}^{L\times d}.$$

This subspace contains every direction in the conditioning space that changes the images the prompt produces. For *"An image of an elf"*, one such direction might shift the generated elves toward Christmas helpers, another toward armored warriors. Finding $\mathcal{T}_c$ therefore amounts to collecting enough such directions to span it. We propose to recover them from the sampled images.

Given a prompt with conditioning c, we first sample a dataset of N images, $\mathcal{X} = \{x^{(1)}, \ldots, x^{(N)}\}$. To extract a single semantic direction, we sample a pair of distinct images $(x^{(i)}, x^{(j)})$ from $\mathcal{X}$. We encode $x^{(i)}$ into the latent space to obtain $z_0^{(i)}$ and apply forward diffusion to a noise level t:

$$z_t^{(i)} = \alpha_t z_0^{(i)} + \sigma_t \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

At any timestep t, the network ϵ_θ predicts an estimate of the clean latent, $\hat{z}_0$. By Tweedie's formula, this prediction serves as an estimator for the conditional expectation of the clean image:

$$f_t(c, z_t^{(i)}) := \hat{z}_0(z_t^{(i)}, t, c) = \frac{z_t^{(i)} - \sigma_t \epsilon_\theta(z_t^{(i)}, t, c)}{\alpha_t} \approx \mathbb{E}[z_0 \mid z_t^{(i)}, c].$$

For a fixed $z_t^{(i)}$ and noise level t, the function f_t defines a parameterized smooth mapping from the conditioning into the visual latent space.

Because f_t approximates the expected clean image, we can formulate an objective to steer this expectation toward a target visual state. We compute the standard diffusion reconstruction loss (MSE) between the prediction and the latent representation of the second image $x^{(j)}$:

$$\mathcal{L}_{sem} = \|f_t(c, z_t^{(i)}) - z_0^{(j)}\|_2^2$$

(see App. L for alternative formulations), and backpropagate it to obtain the gradient with respect to the conditioning, $g_{i,j} = \nabla_c \mathcal{L}_{sem}$.

Intuitively, minimizing this loss forces the model to alter the conditioning such that the expected generated output shifts from the characteristics of $x^{(i)}$ toward $x^{(j)}$. Formally, let $J_{c,t} = \nabla_c f_t(c, z_t^{(i)})$ be the Jacobian matrix of the denoising mapping. Applying the chain rule to the semantic loss yields

$$g_{i,j} = 2J_{c,t}^\top (f_t(c, z_t^{(i)}) - z_0^{(j)}).$$

The visual difference vector $(f_t(c, z_t^{(i)}) - z_0^{(j)})$ represents a macroscopic transformation toward the target image. When multiplied by $J_{c,t}^\top$, any component of this difference lying in the left null space of the Jacobian - representing changes that the conditioning cannot control - is mapped to zero. Consequently, $g_{i,j} \in \ker(J_{c,t})^\perp$, and since a displacement that leaves $\Phi(c)$ unchanged also leaves every $f_t(\cdot, z_t)$ unchanged, we have $\ker(d\Phi_c) \subseteq \ker(J_{c,t})$. Each gradient therefore lies in the tangent space we set out to recover, $g_{i,j} \in \mathcal{T}_c$.

Repeating this process for many pairs gives a dataset of gradients $\mathcal{G} = \{g_1, \ldots, g_M\}$, representing the semantic directions that span the tangent space of the concept in the embedding space.

3.2 Restricting to a Concept

The construction above recovers the variations of the entire generated image. In practice, however, one is more often interested in something more specific, such as the variations of a single concept within a scene, while the rest of it stays intact. Prior work shows that the influence of individual tokens is largely localized, with cross-attention maps attributing image content to the corresponding words (Hertz et al., 2022; Tang et al., 2023). Following these observations, we assume that the characteristics of a concept are controlled predominantly by its own tokens.

Under this assumption, fixing the remaining tokens and varying only the embedding $e \in \mathbb{R}^d$ of the token of interest traces a submanifold of $\mathcal{M}$, which we refer to as the *concept manifold*. Its tangent space is a subspace of $\mathcal{T}_c$ and, since $\nabla_e \mathcal{L}_{sem}$ is the block of $\nabla_c \mathcal{L}_{sem}$ corresponding to that token, the derivation above carries over unchanged: we collect gradients exactly as before, restricting the backward pass to the concept tokens, and obtain $\mathcal{G} \subset \mathcal{T}_e$.

3.3 Discovering Latent Directions

The raw gradients in $\mathcal{G}$ often contain entangled information about several semantic changes together with unimportant noise. To extract disentangled, semantic directions, we decompose the set of gradients with two approaches:

Principal Component Analysis (PCA). We perform PCA on $\mathcal{G}$. The resulting latent directions are defined as the principal components (eigenvectors) v_k. To filter out noise, we retain only components with explained variance above a threshold τ.

Sparse Autoencoder Decomposition. For more interpretable directions, we train a Top-k Sparse Autoencoder (SAE) (Bussmann et al., 2024) on the set of gradients $\mathcal{G}$. To that end, we use a model composed of two neural networks: an encoder and a decoder trained using a reconstruction loss with sparsity constraints. Given an input vector $v \in \mathbb{R}^d$, the encoder projects it to a higher-dimensional latent space $h \in \mathbb{R}^n$ using a linear transformation followed by a TopK activation function: $h = \text{TopK}(W_{enc}(v - b_{pre}))$, where $W_{enc} \in \mathbb{R}^{n \times d}$ is the encoder weight matrix, $b_{pre} \in \mathbb{R}^d$ is a pre-encoder bias, and $\text{TopK}(\cdot)$ is a non-linearity that retains only the k largest activation values, setting the rest to zero. This architectural constraint enforces a strict sparsity level of k active features per input. The decoder then reconstructs the input from the sparse latent code h, and minimizes the reconstruction error:

$$\mathcal{L}_{SAE} = \|v - (W_{dec}h + b_{pre})\|_2^2,$$

where $W_{dec} \in \mathbb{R}^{d \times n}$ is the decoder weight matrix. Post-training, we identify the set of "alive" latent features (with non-zero activation density). The decoder weights corresponding to these features, $d_k = (W_{dec})_{:,k}$, represent the disentangled semantic directions. The sparsity constraint ensures that these directions represent fundamental, primitive attributes of the concept.

3.4 Semantic Steering

Given a discovered latent direction d (derived either from PCA or SAE), we can precisely steer the generation process to obtain a final generation with a characteristic defined by a given direction. For a given prompt, let e be the original embedding of the concept token. We modulate the embedding by injecting the latent direction: $e' = e + \lambda \cdot d$, where $\lambda \in \mathbb{R}$ is a strength parameter. The modified embedding is then passed to the diffusion model. This method scales naturally to complex prompts with multiple subjects, as directions are derived relative to specific tokens. Consequently, we can apply distinct directions to different subjects simultaneously with limited cross-concept interference.

4 Experiments

For the large-scale experiments, we use SDXL (Podell et al., 2023) and its distilled version SDXL-DMD (Yin et al., 2024), and show in App. G that ELROND generalizes to the transformer-based Flux Schnell (BlackForestLabs, 2024). The text conditioning acts as a concatenation of embeddings: from two distinct CLIP models for SDXL, and from a CLIP and T5 encoder for Flux-Schnell. Consequently, our method operates directly on this joint embedding space (dim=2048 for SDXL, dim=4096 for Flux). To analyze a specific concept c, we employ a generic prompt template $P =$ "A picture of $\langle c \rangle$".

For each concept, we sample $N = 10,000$ images using the base model and collect $10,000$ semantic gradients as described in Section 3. For the synthetic model, we collect this set independently at each diffusion timestep to fully characterize the recovered subspace. For the large-scale models, collecting gradients at every timestep would be computationally prohibitive, and unnecessary at the same time, as the conditioning signal is mosty used at the beginning of the sampling sequence. So, we restrict the backward pass to the highest-noise timestep ($t = T$) (see App. F for ablation). We provide the used intervention strengths λ and an ablation of its effect in App. D. For the PCA decomposition, we use a significance threshold of $\tau = 2.5/d_{embed}$, and for SAE training $k = 32$ and $d = d_{embed}$ (see App. C for ablations). The architecture and training details of the toy model used in Sec. 4.1 are provided in App. A.2.

4.1 Validating the recovered subspace

The validity of our method relies on the recovered subspace matching the concept's tangent space. Since $\mathcal{T}_c$ of a large-scale model is unknown, we first verify the recovery on an artificial dataset.

Synthetic dataset validation We train a diffusion model on a synthetic dataset of 64×64 images, each containing a single geometric shape with known attributes: size, color, and position. Each shape has a different set of attributes that change between instances. The model is conditioned on a vector $u \in \mathbb{R}^{3+A}$ mapped into 64-dimensional space by a linear encoder E. The first three coordinates of u one-hot encode the shape. Each of the remaining A coordinates, taking values in $[0, 1]$, controls the distribution one attribute is drawn from rather than setting its value directly. At 0.5, the attribute is drawn from a wide window centered in the middle of its range; moving the coordinate toward either end slides and narrows that window, and near the extremes it closes entirely, fixing the attribute to a single value. This mirrors text conditioning, where *"a cat"* admits every cat the model has learned and *"a big cat"* restricts the distribution to a given subset. We assume that the model has learned the dataset faithfully and, given any conditioning, generates shapes with correct distributions of attributes, which, given the simplicity of the dataset, we take to hold approximately. Consequently, the subspace $\mathcal{T}_c$ at a given shape c is exactly the image under E of the coordinate directions of its varying attributes.

Table 1: Subspaces recovered on the synthetic dataset: dimension estimated by VAEase, and precision and recall of the PCA subspace with respect to the ground truth.

Shape	**Square**	**Triangle**	**Circle**
Ground-truth dim.	3	4	6
Estimated dim.	3	4	8
Recall	0.956	0.939	0.968
Precision	0.956	0.939	0.726

We collect 10 000 gradients from each of 20 uniformly spaced timesteps described in Sec. 3 and first estimate the dimensionality of the recovered subspace using VAEase (Lu et al., 2025). As shown in Tab. 1, VAEase estimates only 3-8 dimensions across the different shapes, indicating that the recovered variation is concentrated in a low-dimensional subspace. Moreover, the estimated dimensionalities closely match the ground-truth dimensions and preserve their relative ordering across shapes. The only discrepancy occurs for the circle, for which VAEase estimates 8 dimensions compared to the 6-dimensional ground truth. We attribute them to imperfections in model training; see App. A.4 for further discussion.

We next test whether these estimated dimensions correspond to the correct subspace. We perform PCA on the collected gradients and retain the top $\hat{d}_c$ components, where $\hat{d}_c$ is the VAEase estimate, yielding an orthonormal basis $\{u_i\}$ of the recovered subspace $\hat{\mathcal{T}}_c$. Letting P and $\hat{P}$ denote the orthogonal projections onto the ground-truth subspace $\mathcal{T}_c$ and recovered subspace $\hat{\mathcal{T}}_c$, respectively, and $\{v_j\}$ an orthonormal basis of $\mathcal{T}_c$, we measure precision and recall defined as:

$$\text{precision} = \frac{1}{\dim \hat{\mathcal{T}}_c} \sum_i \|Pu_i\|^2, \qquad \text{recall} = \frac{1}{\dim \mathcal{T}_c} \sum_j \|\hat{P}v_j\|^2.$$

Intuitively, recall measures how much of the ground-truth subspace is spanned by the recovered one, while precision measures how much of the recovered subspace lies within the ground-truth one. As shown in Tab. 1, recall exceeds 90% for all shapes, indicating that the recovered directions span the ground-truth tangent space well. Precision is likewise high, showing that the recovered subspace is

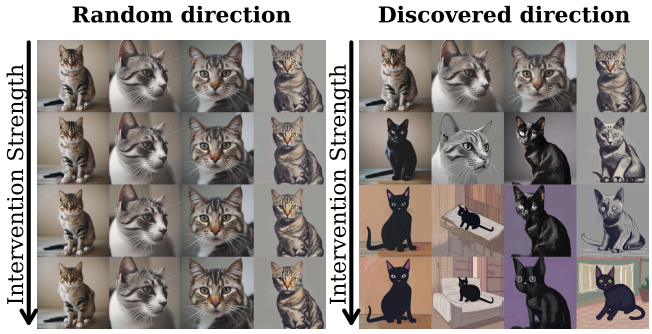

Figure 3: The diffusion model responds far more strongly to directions found by ELROND than to random directions of the same norm.

Table 2: Mean DreamSim distance (↑) between images generated before and after perturbing the embedding.

Concept	Random	Discovered
Cat	0.068	**0.230**
Monster	0.095	**0.288**
Car	0.085	**0.249**
Dog	0.073	**0.223**
Person	0.091	**0.261**

Table 3: Decomposition of the measured variation. VLM ratings (1–5) of how much each factor differs within a pair, and the share of pairs attributed to each primary driver.

Method	Targ. attr. ↑	Comp. ↓	Backgr. ↓	Style ↓	Artif. ↓	Primary driver ↑
ELROND-PCA	4.76	1.89	1.87	1.84	1.27	96.7%
ELROND-SAE	**4.79**	**1.62**	**1.61**	**1.52**	**1.10**	**97.7%**
SliderSpace	4.67	1.68	1.65	1.55	1.25	94.0%

predominantly concentrated within the correct tangent space. The only notable deviation is the circle, whose precision falls below 90%, consistent with the overestimate of its dimensionality by VAEase.

Validation on Large-Scale Models On large-scale text-to-image models, no ground truth is available, and we therefore validate the recovery behaviourally. If the collected gradients span $\mathcal{T}_c$, the model's sensitivity to its conditioning should be concentrated there: displacements within the recovered subspace should change the generated images, while displacements of equal magnitude outside it should leave them unchanged, as the embedding space is high-dimensional and the manifold occupies a small part of it.

As illustrated in Fig. 3, modulating the embedding with a discovered direction significantly alters the semantic content of the image, while adding a random vector of the same norm results in negligible visual change. To quantify this, we sample 2,000 image pairs per concept, generating a base image and a perturbed one for each, and measure the perceptual shift using DreamSim (Fu et al., 2023). Tab. 2 reports that the mean distance under our directions is substantially higher than under random perturbations of equal magnitude.

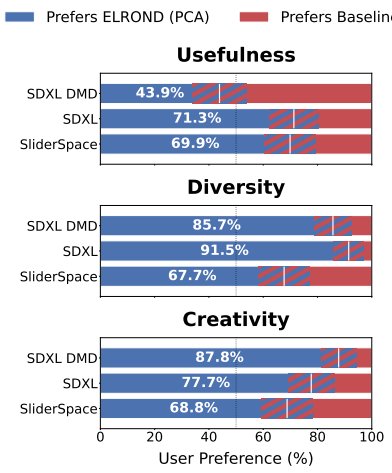

Figure 4: User study results evaluating ELROND against baselines across usefulness, diversity, and creativity.

4.2 Exploring the Model's Capabilities

Having established that the recovered subspace is valid, we turn to its practical use. Following Gandikota et al. (2025), we apply our method to a set of concepts, extract the interpretable latent directions, and use them to steer the generation process. As visualized in Fig. 5, the discovered directions correspond to distinct, semantically coherent attributes, with further examples in App. J.

A useful decomposition should not only identify attributes but also expose the model's full range of variation. To quantify this, we generate $10,000$ images per concept, modifying the concept token embedding of each by a linear combination of 3 randomly selected latent directions. We measure intra-

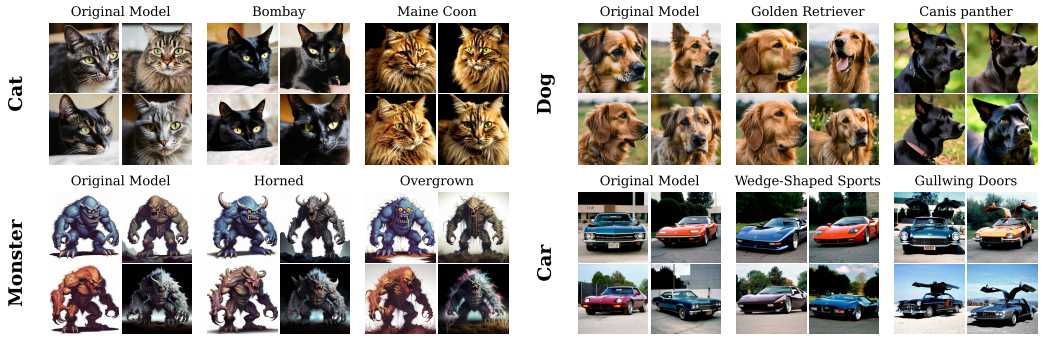

Figure 5: ELROND decomposes the visual capabilities of the diffusion model. We show latent directions discovered in SDXL-DMD.

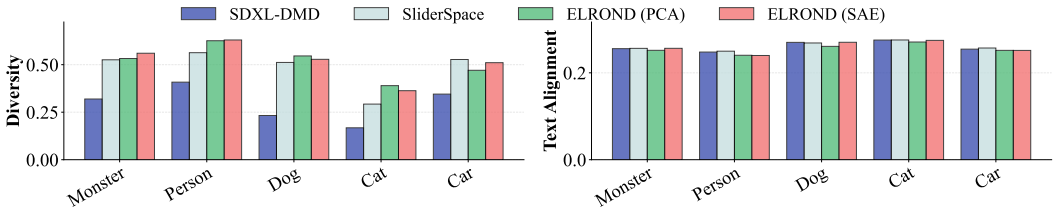

Figure 6: ELROND generates visually diverse variations of concepts (left) as measured by mean DreamSim distance (higher is better). Despite this, it maintains similar text-to-image alignment (right), as measured by CLIP score (higher is better).

class diversity using DreamSim (Fu et al., 2023) and semantic consistency via CLIP Distance (Hessel et al., 2021), comparing against the base SDXL-DMD model and SliderSpace (Gandikota et al., 2025). While all methods maintain high semantic consistency with the original prompt (Fig. 6, right), ELROND yields a significantly higher diversity score than the base model across all tested concepts (Fig. 6, left) and outperforms SliderSpace.

Since DreamSim measures perceptual distance without regard to what changed, a higher score could in principle reflect drift in background, composition, or style rather than variation of the concept itself. To rule this out, we sample 60 image pairs per concept (300 per method) and ask Qwen2-VL (Wang et al., 2024) to rate each pair on five independent axes - how much the concept's target attribute changed, and separately how much the composition, background, and style changed - together with an absolute artifact rating, and to make a single forced choice of the primary driver of the difference. As reported in Tab. 3, both variants of ELROND rate the target attribute far above every off-target axis, artifacts remain low throughout, and the forced choice identifies the target attribute in over 96% of pairs. The measured diversity gain therefore reflects genuine, concept-relevant variation rather than incidental drift.

A user study (Fig. 4) confirms the advantage of our method: 40 human evaluators significantly preferred ELROND in terms of diversity and creativity, while on image fidelity, our method's win rate falls within the confidence interval of being statistically equivalent to the unperturbed model, indicating no degradation in quality.

We posit that this advantage stems from the source of the semantic signal. Unlike ELROND, SliderSpace derives its axes from CLIP embeddings of generated images, and thus organizes the variations according to a model other than the one producing them. Consequently, the method is prone to missing modes of diversity that the external model struggles with, while detecting ones that the DM may be unable to produce (see App. B for discrepancies between the models).

4.3 Diversity Enhancement and Mode Collapse Mitigation

Having demonstrated that ELROND is able to span the rich semantic manifold of learned concepts, we now apply it to the problem of mode collapse in distilled diffusion models (Yin et al., 2024), which

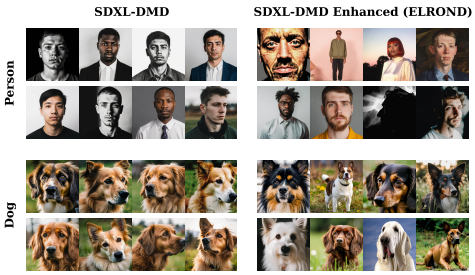

Figure 7: Adding latent directions discovered by ELROND (SAE) in the SDXL-DMD model, mitigates mode collapse in distilled model.

Table 4: Comparison of FID scores calculated between samples generated by SDXL and the distilled SDXL-DMD.

Concept	ELROND (SDXL)	ELROND (SDXL-DMD)	ELROND PCA (SDXL-DMD)	Slider Space	SDXL-DMD
Car	**25.48**	26.35	26.57	31.80	37.55
Cat	**19.72**	19.94	23.60	21.69	38.79
Dog	**24.06**	25.13	32.95	38.25	50.25
Monster	**95.05**	99.63	107.46	136.89	103.95
Person	**33.61**	38.90	47.35	41.52	86.94

manifests as repetitive outputs with reduced variation compared to their teachers. We discover latent directions for a set of concepts using both the teacher model (SDXL) and the distilled student (SDXL-DMD). To undo mode collapse, we sample images using SDXL-DMD while modifying the prompt embedding with a linear combination of 3 random directions. We evaluate three settings: using directions discovered within SDXL-DMD by PCA and SAE, and transferring directions discovered by SAE in the base SDXL model. We extend our evaluation to SDXL-Turbo in App. K.

Qualitative results in Fig. 7 show that ELROND reverses the mode collapse. We quantify this in Tab. 4 by computing the FID between a reference distribution of $10,000$ images sampled from the teacher and the distributions generated by SDXL-DMD under various control methods. Our SAE variant using student directions achieves a lower FID than both the base distilled model and SliderSpace, while directions transferred from the teacher perform best overall. This indicates that the teacher retains a richer tangent space than the student, and that our method can transfer it without retraining.

The SAE-based variant also consistently outperforms the PCA-based one. We attribute this to the SAE's ability to capture the *probability density* of the concept manifold, rather than just its geometric span. While PCA identifies a linear subspace where random sampling implies a uniform distribution over the manifold's extent, the SAE dictionary adapts to the non-uniform density of the semantic gradients. Formally, because the SAE minimizes the expected reconstruction error $\mathcal{L} = \mathbb{E}_{\mathbf{g}\sim p(\mathbf{g})}\|\mathbf{g} - \hat{\mathbf{g}}\|^2$ over the gradient distribution $p(\mathbf{g})$, it naturally allocates a higher density of feature vectors to regions of the manifold where $p(\mathbf{g})$ is high. Consequently, sampling random directions from the SAE dictionary effectively approximates sampling from the true underlying density $p(\mathbf{g})$, yielding a generated distribution that better matches the reference than the geometry-only approximation of PCA.

5 Limitations

ELROND, while effective, is not without limitations. First, while our approach avoids retraining, data collection is computationally extensive due to backpropagating through the diffusion model. However, even under an equal computational budget, ELROND maintains competitive performance (App. I). We additionally evaluate the effect of dataset size on the discovered subspace in App. E. Second, our concept-level analysis relies on the common assumption that the influence of a token is predominantly localized to the concept it represents. Because token embeddings are contextual, perturbing one token can nevertheless affect the visual realization of other tokens (App. H). Finally, some directions discovered by either SAE or PCA may not admit a clear interpretation (App. H).

6 Conclusions

We introduce ELROND, a method for recovering the tangent space of the generative manifold of a diffusion model, by collecting gradients in its conditioning space. We validate the recovery against a known ground truth and behaviourally on large-scale models, and show that the recovered structure enables broader exploration of a model's capabilities than external proxies and mitigates mode collapse in distilled models. Ultimately, we offer a geometry-grounded account of a model's generative variety, derived from the model itself rather than imposed upon it.

AI USE STATEMENT

We employed generative AI tools during this work, primarily to assist with developing experimental code and to support the analysis of results. Generative models are also part of our evaluation: we use Qwen2-VL as an automatic judge of the variation between image pairs (Sec. 4.2), and all studied diffusion models are themselves generative systems.

We did not use generative AI to originate the research question or core idea, develop the mathematical formulation, or propose research hypotheses. We employed an LLM to improve the clarity and readability of the manuscript and the title. All AI-assisted text and code have been manually verified by us. As authors, we take full responsibility for the final content of this work, including text, claims, code, and visualizations produced with the aid of generative AI.

ETHICS STATEMENT

This work studies how diffusion models organize the images they can generate for a given prompt. All experiments use publicly available models and a synthetic dataset of geometric shapes, and no new real-world data was collected. Our user study involved 40 volunteers who compared generated images anonymously; no personal or demographic information was recorded (App. M). Because ELROND surfaces variation the model has already learned, it may also expose biases present in the model, for instance along demographic attributes of concepts such as *person*. We see this as useful for auditing such biases, but note that the method does not correct them. As with any image-generation tool, the increased control could be misused, though it does not extend what the underlying model is able to produce.

REPRODUCIBILITY STATEMENT

We aim to make our results easy to reproduce. Section 3 describes the full method, and Sec. 4 lists the models, prompts, dataset sizes, and main hyperparameters. The synthetic dataset and toy model are specified in App. A.1 and App. A.2. We justify our choices with ablations of the loss function (App. L), diffusion timestep (App. F), SAE dictionary size (App. C), intervention strength (App. D), and number of gradients, together with stability across random seeds (App. E). The compute budget is reported in App. I, and the user study protocol in App. M. All models used are publicly available, and we will release our code upon publication.

AUTHOR CONTRIBUTIONS

If you'd like to, you may include a section for author contributions as is done in many journals. This is optional and at the discretion of the authors.

ACKNOWLEDGMENTS

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper.

REFERENCES

Stefan Andreas Baumann, Felix Krause, Michael Neumayr, Nick Stracke, Melvin Sevi, Vincent Tao Hu, and Björn Ommer. Continuous, subject-specific attribute control in t2i models by identifying semantic directions. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13231–13241, 2025.

BlackForestLabs. Flux.1 [dev] model card. `https://huggingface.co/black-forest-labs/FLUX.1-dev`, 2024.

Manuel Brack, Felix Friedrich, Katharina Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinário Passos. Ledits++: Limitless image editing using text-to-image models. *arXiv preprint arXiv: 2311.16711*, 2023.

Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. *arXiv preprint arXiv:2412.06410*, 2024.

Bartosz Cywiński and Kamil Deja. Saeuron: Interpretable concept unlearning in diffusion models with sparse autoencoders. *arXiv preprint arXiv:2501.18052*, 2025.

Yusuf Dalva and Pinar Yanardag. Noiseclr: A contrastive learning approach for unsupervised discovery of interpretable directions in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24209–24218, 2024.

Amil Dravid, Yossi Gandelsman, Kuan-Chieh Wang, Rameen Abdal, Gordon Wetzstein, Alexei Efros, and Kfir Aberman. Interpreting the weight space of customized diffusion models. *Advances in Neural Information Processing Systems*, 37:137334–137371, 2024.

Maximilian Dreyer, Lorenz Hufe, Jim Berend, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. From what to how: Attributing clip's latent components reveals unexpected semantic reliance. *arXiv preprint arXiv:2505.20229*, 2025.

Josh Engels, Eric Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear. In *International Conference on Learning Representations*, volume 2025, pp. 84591–84622, 2025.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023.

Rohit Gandikota, Joanna Materzyńska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. In *European Conference on Computer Vision*, pp. 172–188. Springer, 2024.

Rohit Gandikota, Zongze Wu, Richard Zhang, David Bau, Eli Shechtman, and Nick Kolkin. Sliderspace: Decomposing the visual capabilities of diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15994–16003, 2025.

Daniel Garibi, Shahar Yadin, Roni Paiss, Omer Tov, Shiran Zada, Ariel Ephrat, Tomer Michaeli, Inbar Mosseri, and Tali Dekel. Tokenverse: Versatile multi-concept personalization in token modulation space. *ACM Transactions On Graphics (TOG)*, 44(4):1–11, 2025.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv: 2208.01626*, 2022.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

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. *arXiv preprint arXiv:2503.07050*, 2025.

Ronen Kamenetsky, Sara Dorfman, Daniel Garibi, Roni Paiss, Or Patashnik, and Daniel Cohen-Or. Saedit: Token-level control for continuous image editing via sparse autoencoder. *arXiv preprint arXiv:2510.05081*, 2025.

Dahye Kim, Xavier Thomas, and Deepti Ghadiyaram. Revelio: Interpreting and leveraging semantic information in diffusion models. *arXiv preprint arXiv:2411.16725*, 2024.

Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. *arXiv preprint arXiv:2210.10960*, 2022.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.

Haozhe Liu, Wentian Zhang, Jinheng Xie, Francesco Faccio, Mengmeng Xu, Tao Xiang, Mike Zheng Shou, Juan-Manuel Perez-Rua, and Jürgen Schmidhuber. Faster diffusion via temporal attention decomposition. *arXiv preprint arXiv: 2404.02747*, 2024.

Nan Liu, Yilun Du, Shuang Li, Joshua B Tenenbaum, and Antonio Torralba. Unsupervised compositional concepts discovery with text-to-image generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2085–2095, 2023.

Yin Lu, Xuening Zhu, Tong He, and David Wipf. Sparse autoencoders, again? *arXiv preprint arXiv:2506.04859*, 2025.

Alexander Modell, Patrick Rubin-Delanchy, and Nick Whiteley. The origins of representation manifolds in large language models. *arXiv preprint arXiv:2505.18235*, 2025.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv: 2211.09794*, 2022.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pp. 4296–4304. AAAI Press, 2024. doi: 10.1609/AAAI.V38I5.28226. URL https://doi.org/10.1609/aaai.v38i5.28226.

Bruno A. Olshausen and David J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, 37:3311–3325, 1997. URL https://api.semanticscholar.org/CorpusID:14208692.

Mateusz Pach, Shyamgopal Karthik, Quentin Bouniot, Serge Belongie, and Zeynep Akata. Sparse autoencoders learn monosemantic features in vision-language models. *arXiv preprint arXiv:2504.02821*, 2025.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Pau Rodriguez, Arno Blaas, Michal Klein, Luca Zappella, Nicholas Apostoloff, Marco Cuturi, and Xavier Suau. Controlling language and diffusion models by transporting activations. *arXiv preprint arXiv:2410.23054*, 2024.

Pau Rodriguez, Michal Klein, Eleonora Gualdoni, Valentino Maiorca, Arno Blaas, Luca Zappella, Xavier Suau, et al. Lineas: End-to-end learning of activation steering with a distributional loss. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

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. *arXiv preprint arXiv:2410.22366*, 2024.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Türe. What the daam: Interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5644–5659, 2023.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Weixin Ye, Hongguang Zhu, Wei Wang, Yahui Liu, and Mengyu Wang. All-in-one slider for attribute manipulation in diffusion models. *arXiv preprint arXiv:2508.19195*, 2025.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *arXiv:2405.14867*, 2024.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3836–3847, 2023.

A SYNTHETIC VALIDATION DETAILS

In this section, we provide details on the synthetic dataset validation performed in Sec. 4.1

A.1 DATASET DETAILS

The synthetic dataset consists of $100{,}000$ RGB images of size 64×64, each containing a single geometric shape: a triangle, a square, or a circle. Every image is generated from a 9-dimensional conditioning vector, whose structure is given in Tab. 5. The first three coordinates one-hot encode the shape, and the remaining six control its colour, size, and position. Because we decide which of these attributes vary for each shape, the dimension of the corresponding tangent space is known exactly.

Table 5: Structure of the 9-dimensional conditioning vector.

Index	Coordinate	Range
0	$\mathrm{is_triangle}$	$\{0, 1\}$
1	$\mathrm{is_square}$	$\{0, 1\}$
2	$\mathrm{is_circle}$	$\{0, 1\}$
3	r	$[0, 1]$
4	g	$[0, 1]$
5	b	$[0, 1]$
6	size	$[0, 1]$
7	x_{pos}	$[0, 1]$
8	y_{pos}	$[0, 1]$

Attributes. The colour is given by three coordinates, one per channel, each mapped to an intensity of at most 220. To keep every shape clearly visible against the background, we scale all three channels down proportionally whenever their sum exceeds 580, and replace combinations whose sum falls below 30 with dark grey. The size coordinate maps to a half-size between 6 and 26 pixels, and the two position coordinates place the centre of the shape between pixels 16 and 48 along each axis. With these ranges, a shape may extend slightly past the edge of the canvas, but most of its surface always remains visible.

From conditioning to distribution. An attribute coordinate does not set the rendered value directly. Instead, it specifies the distribution from which that value is drawn, mirroring how a text prompt specifies a distribution of images rather than a single one (Sec. 4.1). Given a coordinate value v, the rendered value is sampled uniformly from the window $[v - h(v),\ v + h(v)]$ with half-width

$$h(v) = \max\big(0,\ 0.4 - |v - 0.5|\big). \tag{1}$$

At $v = 0.5$ the window is widest and covers $[0.1, 0.9]$. As v moves toward either end of its range, the window moves with it and narrows linearly, until it closes at $v = 0.1$ and $v = 0.9$. From there on, the attribute is rendered exactly at v. A conditioning with attributes near the middle of their ranges therefore describes a broad distribution of images, while one with attributes near the extremes describes an almost deterministic one (Fig. 8).

Varying attributes. Each shape varies in a different subset of attributes, listed in Tab. 6: the square varies only in colour, the triangle in colour and size, and the circle in colour, size, and position. The remaining attributes are held fixed for that shape. We treat each shape as a separate concept, and the number of its varying attributes gives the ground-truth dimension of its tangent space.

Table 6: Varying attributes of each shape and the resulting ground-truth dimension of its tangent space.

Shape	Varying attributes	Ground-truth dimension
Square	r, g, b	3
Triangle	r, g, b, size	4
Circle	r, g, b, size, $x_{\text{pos}}, y_{\text{pos}}$	6

A.2 Model details

Architecture. The diffusion model is a UNet operating directly in pixel space on 64×64 RGB images. It has four resolution stages with 32, 64, 128, and 128 channels and two layers per block. The two highest resolutions (64×64 and 32×32) use standard convolutional blocks, while the lower resolutions (16×16, 8×8, and 4×4) use cross-attention blocks with a head dimension of 8. The conditioning vector is mapped into the model by the linear encoder E, which projects its 9 coordinates to a 64-dimensional embedding matching the cross-attention dimension of the UNet. This embedding space plays the role of the text embedding space of large-scale models, and it is the space in which we collect gradients.

Training. We use a linear noise schedule with $1{,}000$ timesteps and train the model to predict the added noise. Training runs for 200 epochs with a batch size of 64, using AdamW with a learning rate of 10^{-4}, 500 warmup steps, and gradient clipping at a norm of 1.0. We additionally maintain an exponential moving average of the model parameters with a decay of 0.9999, starting after 100 training steps. Samples from the trained model are shown in Fig. 8.

Gradient collection and dimension estimation. For each shape, we generate $5{,}000$ images and use them to collect $10{,}000$ gradients at each of 20 uniformly spaced timesteps. To estimate the dimension of the recovered subspace, we train VAEase with a hidden width and latent dimension of 64, equal to the dimension of the gradients. Training runs for $1{,}000$ epochs using Adam with a learning rate of 10^{-3} and a batch size of $1{,}024$, with 5% of the gradients held out for validation.

A.3 Gradients reflect attribute differences

We verify that individual gradients carry the information we expect. A gradient computed from a pair of images should reflect the attributes that actually differ between them, rather than arbitrary variation. We test this on the circle, the only shape for which all six attributes vary.

We run 100 independent trials. In each, we draw two circles, A and B, with independent attributes, and render both directly from their attribute values using the dataset generator rather than sampling them from the model. We noise the image of A to level t and use a single model call to estimate the clean image. We then compute the pixel MSE between this estimate and the image of B, and take its gradient with respect to the output of E. The model is always conditioned on the same neutral conditioning, with every continuous attribute set to 0.5, rather than on the attributes of A. This way, all gradients are computed at the same point in conditioning space.

For each attribute, we measure the alignment of the negated gradient with that attribute's ground-truth direction, the image under E of its coordinate direction, via their dot product. We average this alignment over 200 evenly spaced timesteps. Since the gradient is taken at the neutral conditioning, the expected change in each attribute is the difference between the value of B and 0.5. We therefore correlate the averaged alignment with this difference across the 100 trials.

As shown in Tab. 7, all six attributes correlate positively, confirming that the gradients track the attributes that differ between the images. The correlations are moderate rather than close to one. We attribute this mainly to averaging over the entire sampling trajectory: attributes are not equally determined at all noise levels, so timesteps at which an attribute is barely affected by the conditioning dilute its signal.

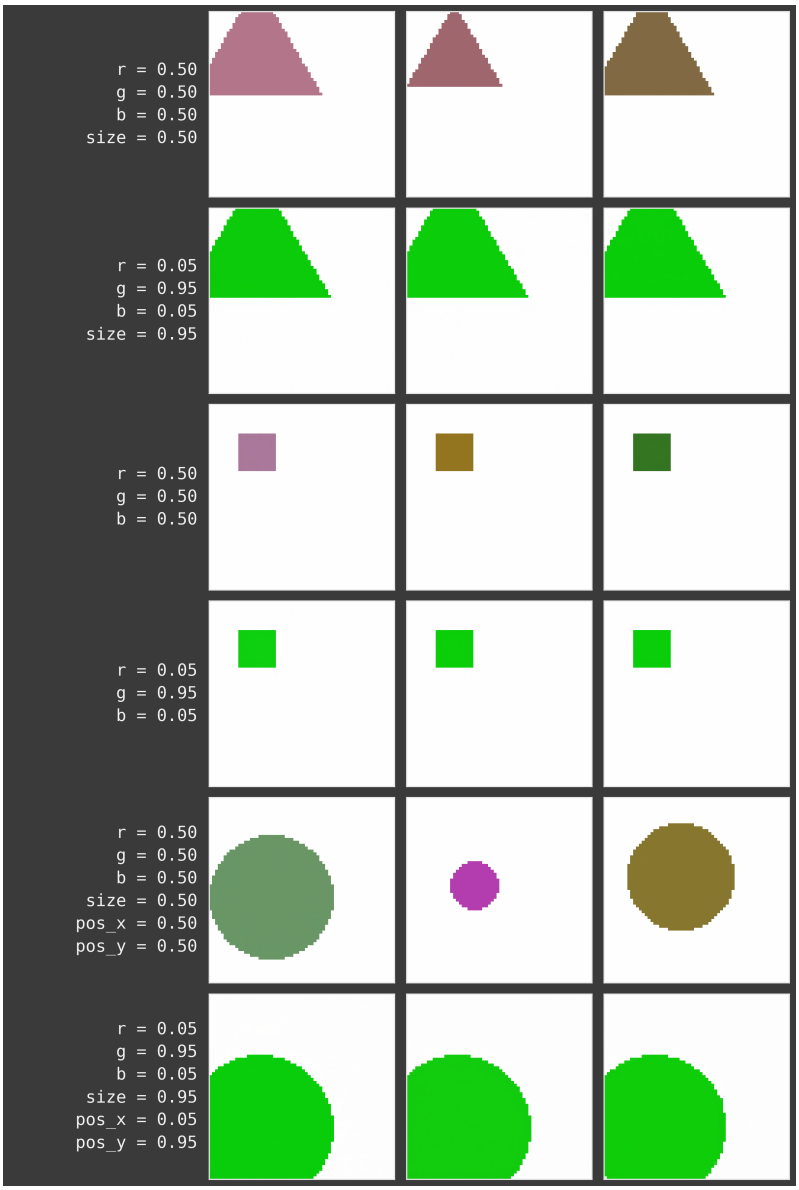

Figure 8: Samples from the trained diffusion model. Each row uses a single conditioning vector, shown on the left, and differs only in the noise seed. When all varying attributes are set to 0.5, the sampling window is at its widest and the samples differ visibly. When they are set near the ends of their ranges, the window is closed and the samples are nearly identical. The number of listed attributes matches the number of varying attributes of each shape: three for the square, four for the triangle, and six for the circle.

A.4 Dimensionality discrepancy for the circle

For the circle, VAEase estimates an 8-dimensional subspace, whereas the dataset defines only 6 free generative factors. We argue that this discrepancy reflects the trained model rather than a failure of ELROND. Our ground truth rests on the assumption, stated in Sec. 4.1, that the model has learned the data distribution faithfully. Under that assumption, the only conditioning directions that alter its output are the images under E of the varying attributes. ELROND, however, recovers $\mathcal{T}_c = \ker(d\Phi_c)^{\perp}$, which is defined by the model and not by the dataset. If the model has additionally learned to respond to conditioning directions that correspond to no generative factor, those directions belong to $\mathcal{T}_c$, and a correct recovery should include them. Nothing in training rules this out. The

Table 7: Correlation, across 100 trials, between the alignment of the gradient with each ground-truth direction and the difference between the attribute value of circle B and 0.5. Alignments are averaged over 200 evenly spaced timesteps before computing the correlation.

Attribute	Correlation
Red	0.509
Green	0.374
Blue	0.576
Size	0.678
x position	0.737
y position	0.719

model only ever observes conditionings from a 9-dimensional subspace of its 64-dimensional input space, and its response to other directions is left unconstrained.

The precision and recall reported in Tab. 1 already point in this direction. Both quantities are normalised traces of $P\hat{P}$, so they satisfy $\dim \hat{\mathcal{T}}_c \cdot \text{precision} = \dim \mathcal{T}_c \cdot \text{recall}$. The circle's recall of 0.968 shows that the ground-truth subspace is almost entirely contained in the recovered one. Its precision of 0.726 equals $0.968 \cdot 6/8$. The drop in precision is therefore fully accounted for by the two surplus dimensions, not by recovered directions that deviate from the ground truth. What remains to establish is whether these two dimensions are genuine structure of the model or estimation noise.

To test whether the model genuinely responds to these two dimensions, we first isolate them. Given matrices $V \in \mathbb{R}^{8\times 64}$ and $A \in \mathbb{R}^{6\times 64}$ whose orthonormal rows span the recovered and ground-truth subspaces, we compute

$$R = V - VA^\top A, \tag{2}$$

which removes from the recovered subspace everything contained in the ground truth. The two leading right singular vectors of R form an orthonormal basis of the part of the recovered subspace orthogonal to the ground truth. We refer to them as the additional directions.

We then intervene along each direction and measure how much the generated image changes. For each of 100 conditioning vectors of the circle, we shift the conditioning along a given direction with strength λ. We then compare the resulting image with the one generated from the unshifted conditioning under the same seed, using the mean squared pixel difference. As a reference, we repeat the procedure for two random directions of equal norm, orthogonal to the full 8-dimensional recovered subspace. According to our method, the model should be insensitive to such directions. If the additional directions were an artifact of the estimation, they should behave in the same way.

Table 8: Mean squared pixel difference between images generated from shifted and unshifted conditionings of the circle, for increasing intervention strength. Results are averaged over 100 conditioning vectors with matched seeds and over the two directions in each group. The last column gives the ratio between the additional and random directions.

Strength	Additional	Random	Ratio
0.1	0.0035	0.00058	6.1
0.2	0.0071	0.00094	7.6
0.3	0.0109	0.00149	7.3
0.4	0.0144	0.00184	7.8
0.5	0.0177	0.00224	7.9
0.6	0.0218	0.00249	8.8

As shown in Tab. 8, the additional directions change the generated image 6 to 9 times more than random directions of the same norm, at every strength we tested. The model therefore does respond to them, and by definition they belong to the tangent space of the trained model. The recovered 8-dimensional subspace is thus an accurate description of the model; the discrepancy with the ground truth stems from the model not having learned the dataset perfectly.

B LINEARITY IN SEMANTIC LATENT SPACES

In this section, we show that the organization of visual variations induced by an external vision model differs significantly from that of a diffusion model.

That these two organizations disagree can be shown directly. Following Baumann et al. (2025), we traverse the conditioning space of SDXL along a straight path between the embeddings of c_{start} = "Person" and c_{end} = "Old person", generating images conditioned on $c_t = (1-t)c_{start} + tc_{end}$ with the noise held fixed, and project the resulting sequence into the CLIP visual embedding space. As shown in Fig. 9, a displacement that is uniform for the diffusion model and perceptually smooth in the images is strongly curved under CLIP. Steps of equal size for one model are therefore of unequal size for the other, and axes that are orthogonal or evenly spaced under an external proxy carry no such guarantee with respect to the geometry of the diffusion model.

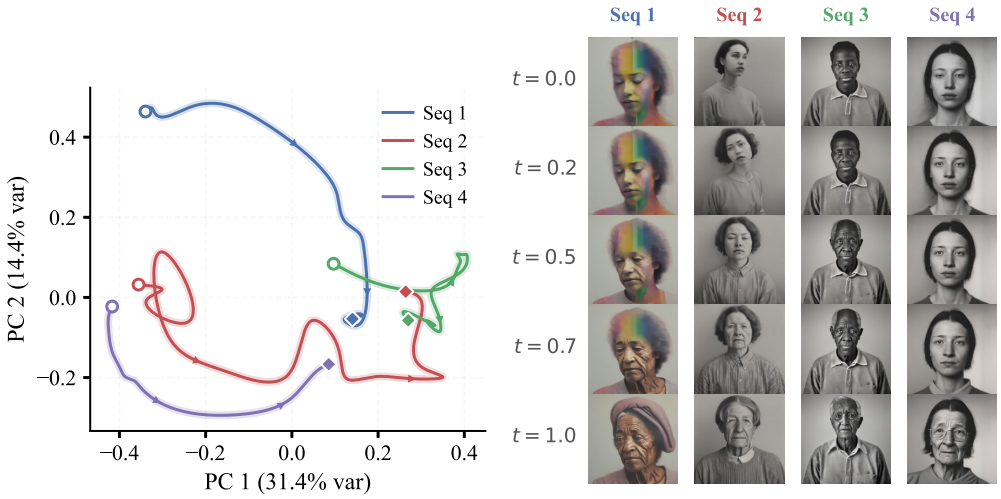

Figure 9: Semantic transitions in the diffusion model outputs resulting from linear interpolation (right) do not correspond to a linear direction within CLIP embedding space (left) (first 2 PCA dimensions).

To quantify this, we generated $N = 700$ trajectories between distinct semantic concepts. We projected these trajectories into the CLIP visual embedding space and computed four metrics to quantify their deviation from linearity:

1. **Straightness Index:** The ratio of the Euclidean distance between the start and end points to the cumulative length of the trajectory. A value of 1.0 indicates a perfectly straight line.
2. **Average Cosine Similarity:** The mean cosine similarity between consecutive step vectors along the path.
3. **Variance Ratio (PC1):** The proportion of variance explained by the first principal component of the trajectory points.
4. **RMSE (Linear Fit):** The root mean square error of the trajectory points relative to a linear interpolation between the start and end states.

The results, presented in Table 9, demonstrate that simple linear interpolation in the text conditioning space yields highly non-linear paths in the visual embedding space, necessitating the gradient-based approach proposed in our method.

C SAE DICTIONARY SIZE

The expansion factor of the SAE determines the size of the learned dictionary and therefore the number of directions the decomposition can express. To select it, we fix the concept (*Monster*) and

Table 9: **Quantitative Analysis of Trajectory Linearity.** Metrics are averaged over $N = 700$ trajectories. The "Ideal" column indicates values for a perfectly linear path. The observed negative cosine similarity and low straightness index confirm the highly non-linear nature of the latent evolution.

Metric	**Observed Value**	**Ideal (Linear)**
Avg. Cosine Similarity	-0.4696 ± 0.0590	1.0
Straightness Index	0.1696 ± 0.0180	1.0
Variance Ratio (PC1)	0.3353 ± 0.0472	1.0
RMSE (Linear Fit)	0.4271 ± 0.0254	0.0

the sparsity $k = 32$, train SAEs at expansion factors $\{1, 2, 4, 8, 16, 32\}$ on the same $50{,}000$ gradients, and count the latents that are selected in the top-k activation of at least one input.

Table 10: Active latents as a function of dictionary size, for a fixed sparsity $k = 32$ on the *Monster* concept.

Expansion factor	Dictionary size	Active latents	% active
1	2,048	1,497	73.1%
2	4,096	2,228	54.4%
4	8,192	3,151	38.5%
8	16,384	4,488	27.4%
16	32,768	6,523	19.9%
32	65,536	9,228	14.1%

As reported in table 10, the number of active latents grows far more slowly than the dictionary itself, increasing roughly sixfold over a thirty-twofold range of capacity, with the fraction of used latents falling from 73% to 14%. Additional capacity is therefore absorbed largely as unused dictionary entries rather than converted into new directions, and we adopt an expansion factor of 1 throughout as the most efficient point on this curve.

D Intervention Strength

Throughout the main experiments we steer by adding a linear combination of three randomly selected directions, scaled by an intervention strength $\lambda = 40$ for all concepts except *Person*, for which $\lambda = 60$ performed better empirically. To characterize this choice, we sweep λ from 1 to 100 on the *Monster* concept for both PCA and SAE directions, measuring intra-pool DreamSim as a measure of diversity and CLIP score against the prompt as a measure of alignment, with $2{,}000$ images per point.

Table 11: Diversity and prompt alignment as a function of intervention strength λ on the *Monster* concept.

λ	1	5	10	20	30	50	70	100
PCA, DreamSim ↑	0.318	0.333	0.363	0.445	0.529	0.635	0.690	0.747
PCA, CLIP score ↑	0.265	0.266	0.265	0.260	0.253	0.243	0.237	0.231
SAE, DreamSim ↑	0.321	0.325	0.349	0.406	0.462	0.572	0.648	0.720
SAE, CLIP score ↑	0.266	0.265	0.265	0.262	0.258	0.250	0.243	0.235

Both metrics vary smoothly and monotonically across the entire range for both decompositions, exhibiting no threshold or discontinuity at which the intervention begins to break down. Alignment is essentially unaffected below $\lambda = 10$, where diversity also grows slowly, and declines steadily thereafter as diversity increases. The values used in the main experiments sit past this inert region while alignment remains close to its maximum. We do not claim this operating point is optimal, only that it is a reasonable choice on a smooth trade-off admitting no sharper criterion.

E INFLUENCE OF THE DATASET SIZE ON THE RESULTING SPACE

In this section, we analyze the stability of the discovered semantic subspace relative to the number of gradients, N, used for decomposition. Our objective is to determine a sufficient dataset size that captures the concept's manifold without excessive computational cost.

To evaluate this, we compute the Principal Component Analysis (PCA) multiple times using varying subset sizes and compare the resulting subspaces against a reference space constructed from $50,000$ gradients.

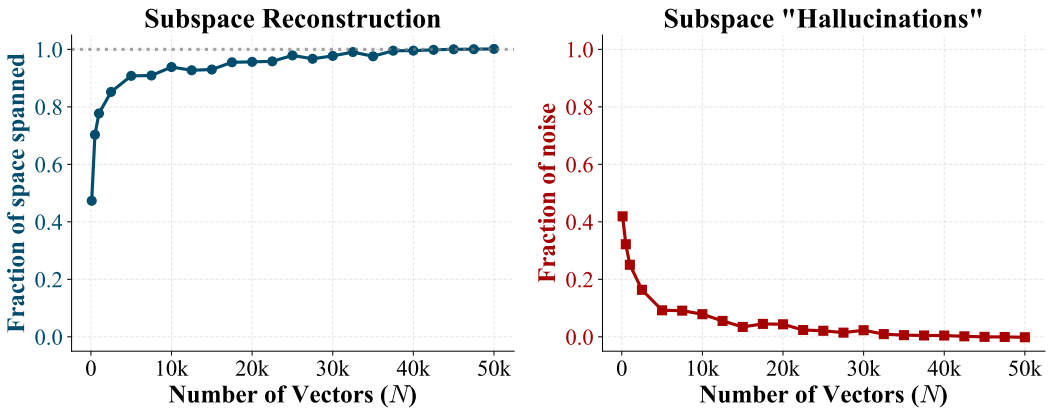

Figure 10: Influence of number of gradients on resulting subspace

As illustrated in fig. 10, the discovered subspace stabilizes rapidly as the number of vectors increases. In fig. 10 (left), we measure the fraction of subspace spanned by the smaller subset compared to the reference space. The curve shows a steep initial ascent, indicating that even a small number of gradients can capture the primary semantic directions. On the other hand in fig. 10 (right) we quantify the fraction of noise, defined as directions present in the smaller subset but absent in the reference, to ensure the discovered axes are semantically valid. This noise fraction decreases significantly as N approaches $30,000$.

By comparing the trade-off between reconstruction accuracy and the reduction of "hallucinated" directions, we identify $30,000$ gradients as the optimal sweet spot for our experiments. This volume provides a robust representation of the concept manifold while maintaining computational efficiency.

Additionally, to verify that the discovered semantic subspace is invariant to the initial noise sampling, we evaluate the stability of ELROND across different random seeds. We sample two independent datasets of $5,000$ images initialized with distinct random noise, calculate $25,000$ semantic gradients for each set, and extract their respective subspaces via PCA. Averaged over 10 such independent pairs, the fraction of the subspace spanned by one set relative to the other remains highly robust (0.930 ± 0.01), accompanied by a negligible fraction of noise (0.069 ± 0.01). These results confirm that ELROND successfully captures the intrinsic semantic manifold of a given concept, fundamentally independent of the specific noise initialization used during the dataset collection phase.

F DIFFUSION TIMESTEP SELECTION

One of the hyperparameters affecting the way our method works is the selection of the diffusion timestep t at which semantic gradients are collected. In this section, we analyze how the noise level during backpropagation influences the richness and magnitude of the discovered semantic directions. Figure 11 illustrates the relationship between the diffusion timestep and the magnitude of the resulting gradient backpropagated to the specific token. We observe that the gradient magnitude generally increases with t, peaking at the highest noise levels. As shown in Table 12, we quantitatively evaluated the semantic alignment of these gradients using DreamSim, DINO, and CLIP similarity metrics.

Our analysis indicates that gradients collected at high noise levels ($t \approx T$) produce the strongest CLIP and DINO similarity scores. This is in line with recent work by Liu et al. (2024), which suggests that during the initial stages of the diffusion process, the model is most reliant on text conditioning to establish the global structure of the image. While the DreamSim distance remains relatively consistent across various timesteps, there is a marginal improvement as t increases, which points toward more precise semantic transitions. Based on these observations, we identified the early denoising stages as the optimal spot for capturing the most semantically rich and steerable directions within the embedding space

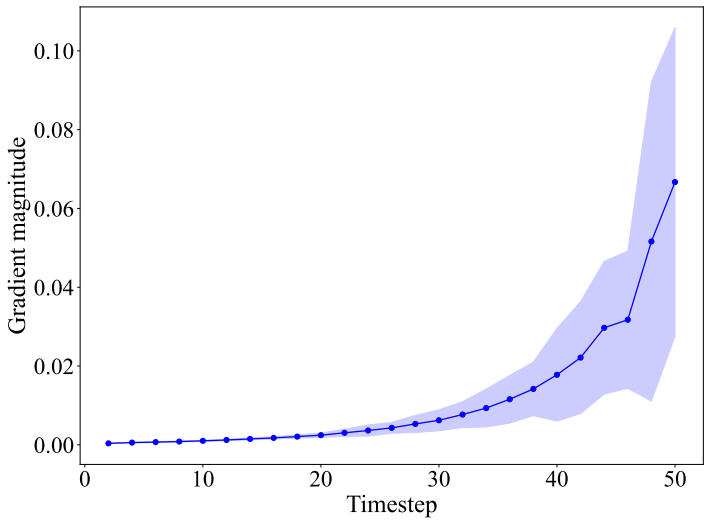

Figure 11: **Gradient Magnitude across Diffusion Timesteps.** The evolution of semantic signal strength relative to the denoising schedule. The increase in gradient magnitude at higher timesteps suggests that the model's receptivity to text-based conditioning is most significant during the early stages of generation, where the high noise floor requires the conditioning signal to define the core semantic layout.

Timestep	DreamSim (↓)	DINO (↑)	CLIP (↑)
36	0.415	0.485	0.781
40	0.415	0.488	0.782
43	0.415	0.486	0.783
46	0.407	0.491	0.785
50	**0.404**	**0.502**	**0.791**

Table 12: Average (across 500 pairs) DreamSim distance, DINO similarity, and CLIP similarity between image changed by a gradient from the corresponding timestep and the image towards which the gradient is pointing. Arrows indicate the direction of better performance.

G GENERALIZATION TO TRANSFORMER-BASED MODELS

To evaluate the architectural agnosticism of ELROND, we extended our experiments to Flux Schnell, a state-of-the-art transformer-based diffusion model. As illustrated in Figure 12, our method successfully identifies semantically rich and steerable directions within the conditioning space of the transformer architecture. The discovered directions for the tested concepts show high levels of semantic coherence, including different styles of cars, textures/breeds of dogs, or demographic and environmental attributes. Interestingly, our method was also able to discover composition-related features like the pose of a dog (lying) or a person (from behind).

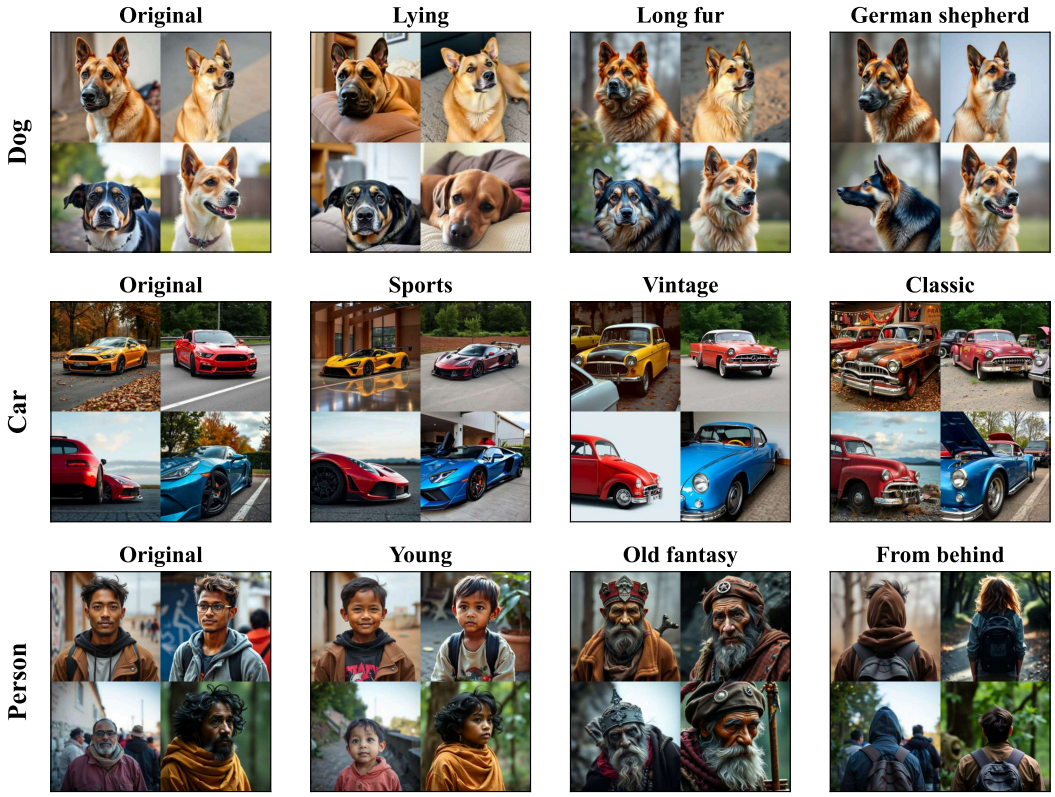

Figure 12: **Generalization to Transformer-based Architectures.** ELROND effectively extracts semantically coherent latent directions from Flux Schnell, a transformer-based diffusion model. The discovered directions for Dog, Car, and Person show high levels of interpretability.

H Analysis of Failure Cases and Limitations

As discussed in the limitations section, the decomposition of the gradient dataset $\mathcal{G}$ can occasionally yield directions that lack clear human interpretability. As visualized in Figure 13, these failure modes typically fall into two categories:

- Perceptually Inert Directions: Modulations that, despite a non-zero norm, fail to produce a significant semantic shift in the generated output.
- Unstructured Artifacts: Interventions that trigger "hallucinated" textures or chaotic pixel-level variations rather than coherent attribute transformations.

Moreover, while our method facilitates token-level steering, the contextual dependency of text embeddings can result in partial semantic leaking. Figure 14 illustrates instances where independent subject steering is compromised, leading to unintended variations in non-target subjects when the magnitude of the steering vector is high.

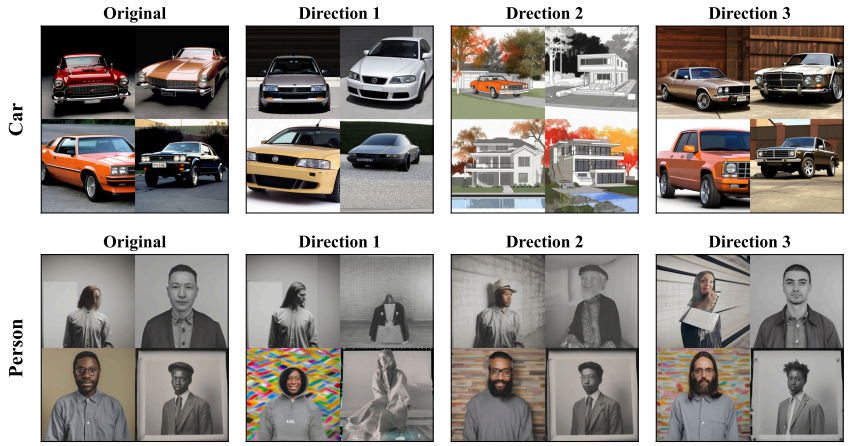

Figure 13: Failure cases for sdxl-dmd.

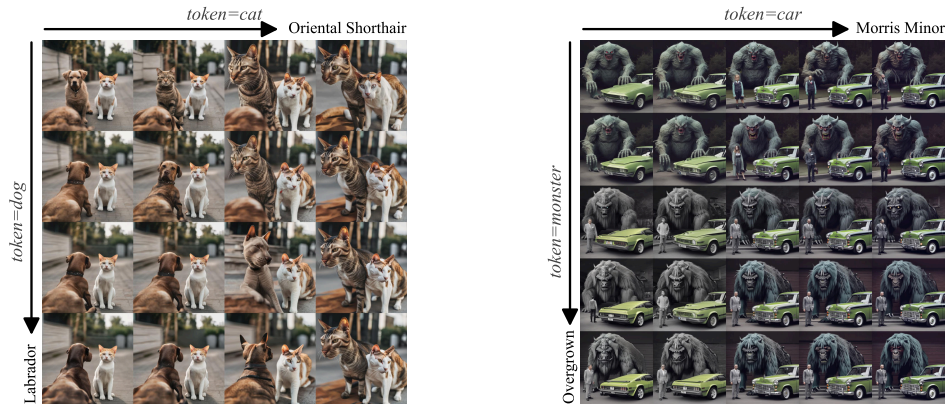

Figure 14: Applying latent directions discovered by ELROND does not always allow for a token-level composability, enabling independent steering of distinct concepts.

I Equal Computation Evaluation

To provide a fair assessment of efficiency, we benchmarked ELROND against SliderSpace under an equal computational budget (utilizing 4 NVIDIA RTX 5000 24GB GPUs, for around 75 minutes). For this setup, we drastically reduced the number of sampled gradients for ELROND to approximately

$2,500$. As presented in Table 13, even under this constrained setup, ELROND maintains highly competitive performance, outperforming SliderSpace in terms of intra-class diversity (DreamSim) on most concepts, while maintaining comparable or better semantic consistency (CLIP Distance).

Furthermore, it is crucial to note a fundamental difference in how these methods scale: SliderSpace's computational cost increases linearly with each additional direction discovered, whereas ELROND maps the entire semantic subspace and extracts all principal directions simultaneously from a single set of gradients.

Table 13: Comparison of ELROND and SliderSpace under an equal computational budget. ELROND maintains competitive diversity and text alignment even when gradient sampling is drastically reduced.

Concept	**Method**	**DreamSim** (↑)	**CLIP** (↑)
Car	ELROND	0.4814	0.2571
	SliderSpace	**0.5273**	0.2511
Cat	ELROND	**0.3986**	0.2693
	SliderSpace	0.2926	0.2756
Dog	ELROND	**0.5151**	0.2648
	SliderSpace	0.5118	0.2687
Monster	ELROND	**0.5936**	0.2471
	SliderSpace	0.5257	0.2563
Person	ELROND	**0.6167**	0.2410
	SliderSpace	0.5631	0.2498

J Discovered subspaces and SAE features analysis

In this section, we provide a quantitative analysis of the semantic subspaces discovered by ELROND. We compare the complexity and richness of the representations learned by the SDXL-DMD model across two decomposition methods: Principal Component Analysis (PCA) and Sparse Autoencoders (SAE).

Table 14: Number of principal components found and non-dead features in the trained SAEs. All SAE have exactly 2048 features total.

Concept	**PCA**	**SAE**
Car	95	1880
Dog	65	1601
Cat	77	1635
Person	68	1854
Monster	74	1425

K Generalization to Alternative Distillation Frameworks

To verify whether our method generalizes across different distillation frameworks, we evaluated ELROND's performance on SDXL-Turbo. For this experiment, we utilized the PCA variant of our decomposition method and transferred the semantic directions discovered directly from the base SDXL teacher model.

As demonstrated in Table 15, ELROND consistently yields substantial improvements in both overall generation quality (lower FID) and intra-class diversity (higher DreamSim distance). Simultaneously, it maintains comparable or slightly improved text-to-image alignment (CLIP Distance) relative to the unmodulated base model. These findings strongly align with our primary results on SDXL-DMD, confirming that ELROND effectively mitigates mode collapse and reintroduces lost semantic diversity across diverse distillation architectures.

Table 15: **FID evaluation of ELROND on SDXL-Turbo.** Lower is better. Applying ELROND consistently reduces FID across all evaluated concepts, indicating improved agreement with the target image distribution.

Concept	SDXL-Turbo	ELROND
Car	67.36	**55.50**
Cat	70.63	**65.19**
Dog	85.78	**60.58**
Monster	276.73	**245.58**
Person	186.76	**118.43**

L Loss Functions for Semantic Gradient Calculation

A critical component of our method is the calculation of the semantic gradient $g_{i,j}$ between a source image $x^{(i)}$ and a target image $x^{(j)}$. This gradient represents the direction in the embedding space that shifts the generation from the source state toward the target state. The choice of the loss function $\mathcal{L}_{sem}$ used to compute this gradient significantly influences the quality and steerability of the resulting semantic directions. In this section, we evaluate four different loss objective formulations to determine the optimal approach.

To assess the efficacy of different loss functions, we constructed a dataset of 200 image pairs sharing the same semantic concept (e.g., pairs of different "cats" or "cars"). For each pair (x_{src}, x_{tgt}), we computed the gradient $\nabla_{e_{tok}} \mathcal{L}(x_{src}, x_{tgt})$ using a specific loss function. As different loss functions naturally produce gradients of different scales, we normalize those gradients. We then initiated the reverse diffusion process using the fixed noise of the source image x_{src}, but intervened in the text embedding space by injecting the calculated gradient.

The resulting generated image x_{gen} represents the model's attempt to transform x_{src} into x_{tgt} using the semantic information encoded in the gradient. We evaluated the alignment between x_{gen} and the target image x_{tgt} using three metrics: DreamSim distance Fu et al. (2023) (perceptual similarity), CLIP similarity (semantic alignment), and DINO similarity (structural/object alignment). We compared the following four loss formulations:

1. **Mean Squared Error (MSE):** The standard reconstruction loss calculated directly between the predicted denoised latent $\hat{z}_0$ of the source and the latent representation of the target.
$$\mathcal{L}_{\text{MSE}} = \|\hat{z}_0^{(src)} - z_0^{(tgt)}\|_2^2 \tag{3}$$
2. **Aligned MSE:** A variant of MSE designed to mitigate spatial misalignment between the source and target objects. We utilized DAAM Tang et al. (2023) to generate attention masks M_{src} and M_{tgt} corresponding to the concept token. We then spatially aligned the features of the target to the source based on the centroids of these masks before computing the MSE. This loss aims to penalize semantic differences while being invariant to object position.
3. **Chamfer Distance:** Treating the feature maps of the source and target latents as sets of feature vectors, we computed the Chamfer distance. This permutation-invariant metric measures the similarity between the two sets of features without enforcing strict spatial correspondence.
4. **Maximum Mean Discrepancy (MMD):** Similar to Chamfer distance, we treated the feature maps as distributions and minimized the MMD kernel distance between the source and target distributions.

The quantitative results are summarized in Table 16. We observe that the standard **MSE** loss consistently outperforms other formulations across all three metrics. It achieves the lowest DreamSim distance (0.4114) and the highest CLIP (0.8037) and DINO (0.5073) similarities.

While *Aligned MSE* performs comparably, the overhead of computing attention maps via DAAM does not yield a performance benefit over standard MSE. The distribution-based losses (*Chamfer* and *MMD*) performed significantly worse, suggesting that preserving the specific spatial structure of the

latent code is crucial for extracting accurate semantic directions in the diffusion embedding space. Based on these findings, we employ standard MSE for all gradient calculations in our main method.

Table 16: Comparison of alignment metrics across different loss functions. The generated images steered by gradients derived from MSE loss exhibit the highest similarity to the target images across perceptual (DreamSim), semantic (CLIP), and structural (DINO) metrics.

Loss	DreamSim Dist. ↓	CLIP Sim. ↑	DINO Sim. ↑
MSE (Ours)	**0.4114**	**0.8037**	**0.5073**
Aligned MSE	0.4123	0.8021	0.5047
Chamfer Distance	0.4439	0.7837	0.4359
MMD	0.4533	0.7707	0.3928

M User study

To thoroughly evaluate the generative capabilities decomposed by our framework, we conducted a comprehensive user study. To that end, we developed a web-based questionnaire to facilitate side-by-side image comparisons. At the beginning of the study, participants were given the following instructions:

1. You will see 25 pairs of generated image collages side by side.
2. For each pair, both collages represent the same concept.
3. For each criterion, pick the option (A or B) that you think is better.
4. Focus solely on the properties of the main concept in the images (e.g. Dog/Car) and try to ignore style, background or other details.
5. The three criteria are: Usefulness (quality/fidelity), Diversity (variety of outputs), and Creativity (originality/surprise).

Figure 15 provides a screenshot of the evaluation interface presented to the users.

Protocol. Each evaluator is presented with 25 randomly sampled pairs of collages. Every pair shows two image grids generated for the same concept: one produced by ELROND (PCA), and one produced by a randomly selected competing method drawn uniformly from $\{\text{SDXL}, \text{SDXL–DMD}, \text{SliderSpace}\}$. The five concepts considered are *car*, *cat*, *dog*, *monster*, and *person*, and for each (method, concept) cell we pre-render three independent collages from which the interface samples uniformly. The side on which ELROND is shown is randomised per pair so that evaluators cannot rely on positional cues, and method identities are hidden behind neutral "Option A" / "Option B" labels. For each pair the evaluator answers three forced-choice questions, one per criterion:

- **Usefulness** – which option has higher quality and better fidelity to the concept;
- **Diversity** – which option exhibits a wider variety of distinct outputs;
- **Creativity** – which option is more original or surprising.

Participants. We collected responses from 40 evaluators, yielding a total of 764 submitted comparisons (mean 19.1, median 25 pairs per evaluator; 28 out of 40 evaluators completed the full set of 25 pairs, the remainder dropped out partway through). No demographic information was collected and participation was voluntary.

N Composability and Subject-Specific Control

A limitation of methods operating on the whole model, such as SliderSpace, is the difficulty of isolating interventions to specific subjects within a scene: because SliderSpace learns global low-rank adapters, applying a direction often affects the entire image. Our method instead operates

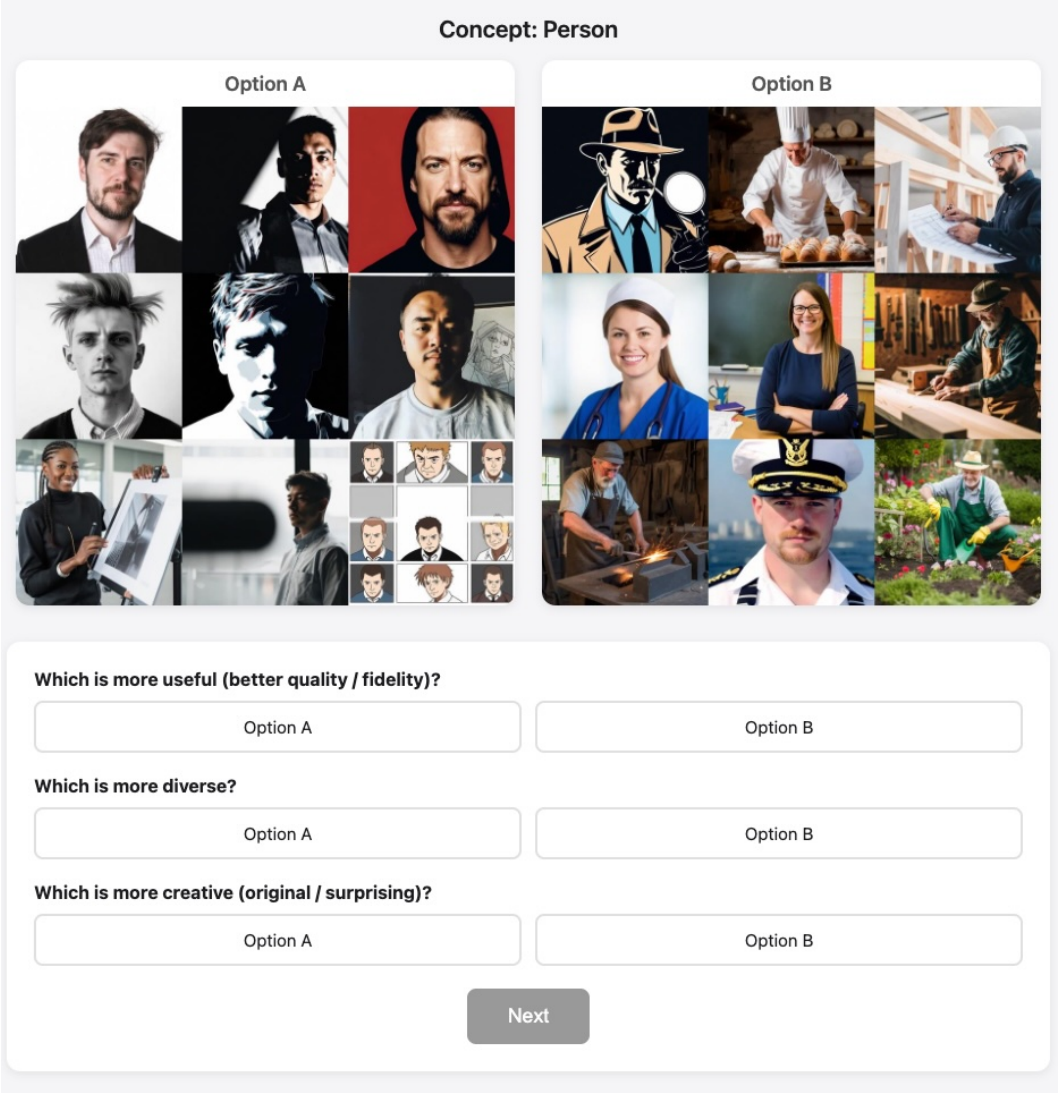

Figure 15: A screenshot of the user study questionnaire, illustrating the side-by-side image collage comparison and the specific questions targeted at usefulness, diversity, and creativity.

on the embeddings of individual tokens, enabling token-level composability, similarly to what was observed by Garibi et al. (2025). Using directions discovered for *Monster* and *Car* in SDXL, we construct prompts containing both subjects and apply distinct directions to the corresponding tokens simultaneously. Following Gandikota et al. (2025), we skip the steering for the first 5 of 50 denoising steps, as the structural layout is determined early. Fig. 16 shows that our method disentangles the control of spatially distinct subjects.

O Additional metrics and visual comparison

In this section, we expand our quantitative evaluation by analyzing the trade-off between the quality and diversity of the generated images using Precision and Recall metrics (Kynkäänniemi et al., 2019). Here, Precision measures the fidelity of the generated samples (i.e., how much they reside within the reference manifold of real data), while Recall quantifies the diversity coverage (i.e., how much of the reference manifold is captured by the generator).

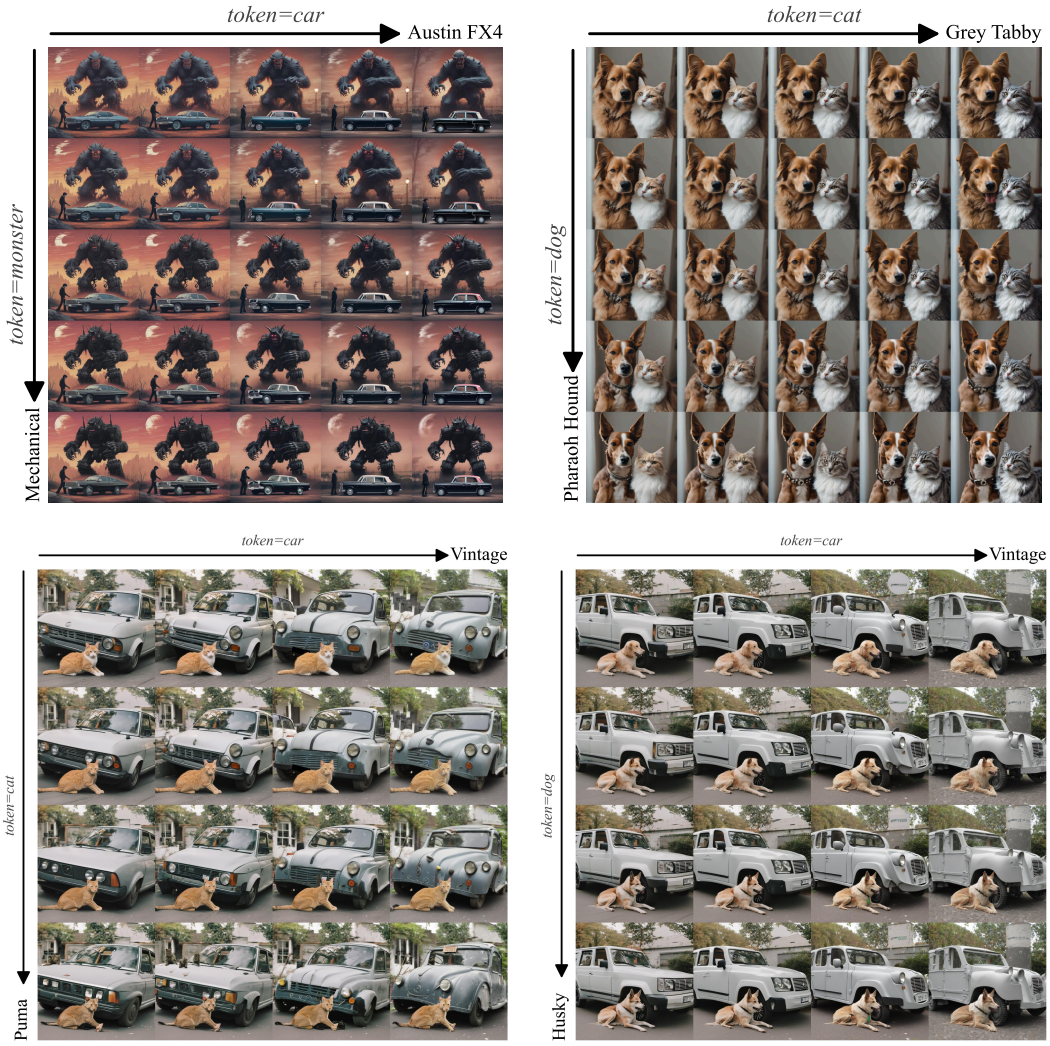

Figure 16: Latent directions discovered by ELROND allow for subject-specific control.

Table 17: **Precision** (↑), measuring how faithful the generated samples are to the reference distribution.

Concept	ELROND SAE (SDXL)	ELROND SAE (SDXL-DMD)	ELROND PCA (SDXL-DMD)	SDXL-DMD	SliderSpace
Car	0.1492	0.1478	0.1331	0.1622	**0.1673**
Cat	0.1410	0.1331	0.0979	**0.2306**	0.1349
Dog	0.1154	0.1105	0.0847	0.1431	**0.1901**
Monster	**0.0710**	0.0682	0.0585	0.0559	0.0649
Person	0.0874	0.0808	0.0808	**0.1169**	0.1019

Quantifying Mode Collapse and Restoration The severity of mode collapse in the distilled SDXL-DMD model is explicitly captured by its low Recall scores in Table 18. For instance, the base SDXL-DMD achieves a Recall of only 0.2059 for the ”Dog” concept, indicating that it generates a very narrow subset of the possible visual variations found in the teacher model. ELROND successfully mitigates this collapse, drastically increasing Recall across all concepts (e.g., improving ”Dog” Recall to 0.7218 with PCA), thereby verifying that our method effectively restores the intrinsic diversity of the generative process.

Table 18: **Recall** (↑), measuring how much of the reference distribution the generated samples cover.

Concept	ELROND SAE (SDXL)	ELROND SAE (SDXL-DMD)	ELROND PCA (SDXL-DMD)	SDXL-DMD	SliderSpace
Car	**0.5179**	0.4782	0.4852	0.1961	0.4672
Cat	0.5240	0.4856	**0.6469**	0.1243	0.4318
Dog	0.6616	0.6123	**0.7218**	0.2059	0.5761
Monster	0.4313	0.4793	**0.5432**	0.1139	0.4463
Person	0.5318	**0.5598**	0.5013	0.1296	0.3936

Comparison of Decomposition Methods: SAE vs. PCA When comparing the two decomposition strategies, a distinct trade-off between geometric coverage and semantic validity emerges. The Principal Component Analysis (PCA) variant consistently yields the highest Recall scores, such as 0.7218 for "Dog" compared to 0.6123 for the Sparse Autoencoder (SAE). This outcome is expected, as PCA maximizes variance and spans the largest possible linear subspace, thereby ensuring broadly diverse outputs. Conversely, the SAE variant consistently achieves higher Precision than PCA – for example, 0.1105 versus 0.0847 for "Dog". This suggests that while PCA covers a larger volume of the latent space, it may inadvertently include "noisier" or less semantically coherent directions. In contrast, the SAE adapts to the non-uniform density of the semantic gradients. By minimizing expected reconstruction error, the SAE naturally allocates a higher density of feature vectors to regions of the manifold where the probability density of gradients is high. Consequently, sampling from the SAE dictionary effectively approximates the true underlying density of the concept, resulting in generated samples that are more faithful to the reference manifold and achieve consistently higher Precision scores (e.g., 0.1105 versus 0.0847 for "Dog").

Comparison with SliderSpace While SliderSpace achieves high Precision scores (e.g., 0.1901 for "Dog"), likely due to its reliance on stable, external CLIP-guided directions, it lags behind our approach in Recall (e.g., 0.5761 vs. 0.7218). This confirms our hypothesis that deriving directions from the model's own gradients is better suited for uncovering the full, latent diversity of the model than when constrained by external proxies.

For direct visual comparison in the following plots, we present random samples from the original models (SDXL and SDXL-DMD) and version augmented with directions from different decomposition techniques.



Figure 17: Uncurated samples generated by the teacher model (SDXL) for the concept ”monster”

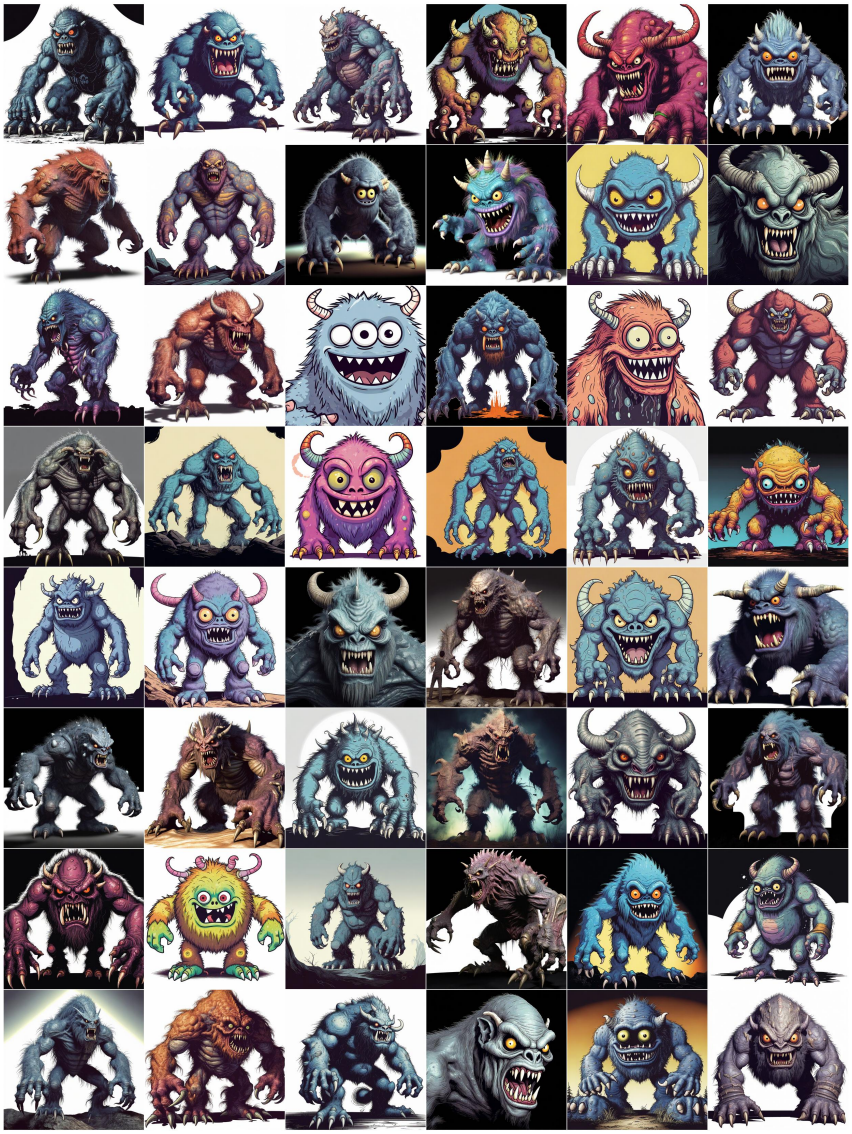

Figure 18: Uncurated samples generated by the distilled model (SDXL-DMD) for the concept ”monster”, illustrating the baseline performance.

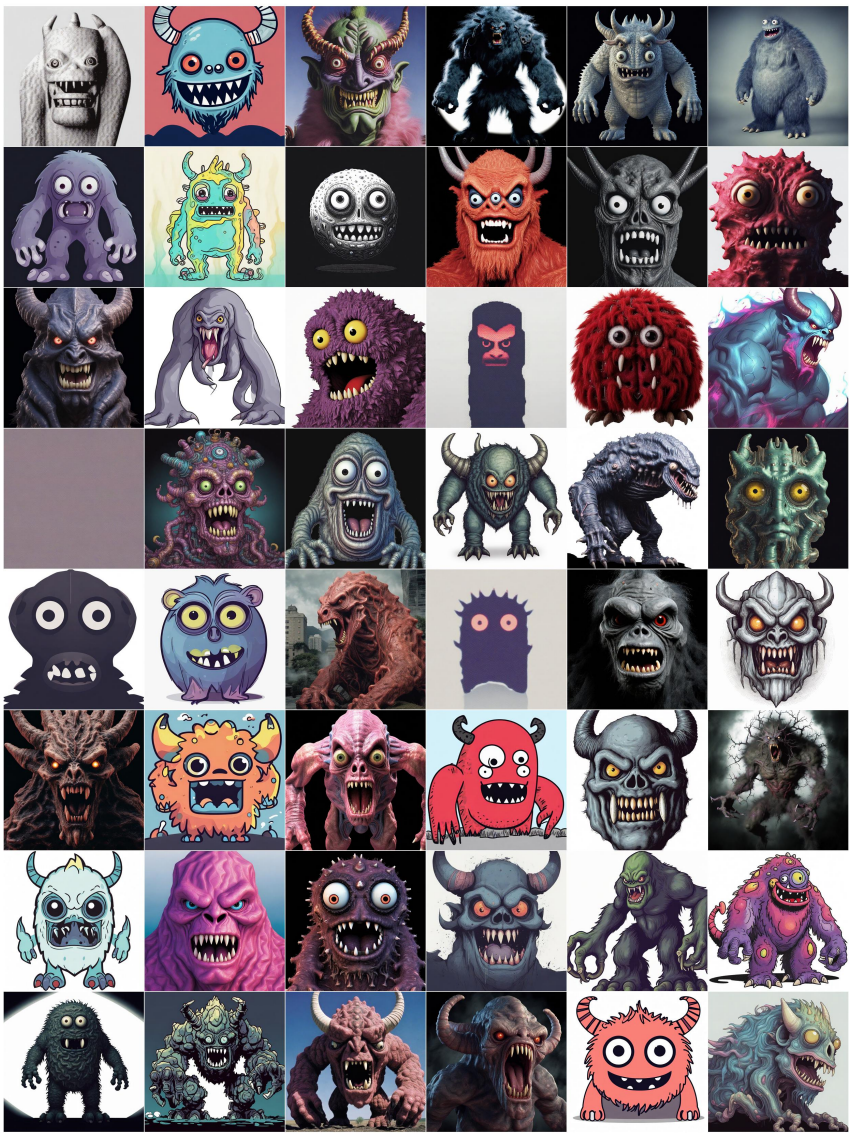

Figure 19: Samples generated by SDXL-DMD when modulated using the **SliderSpace** method.



Figure 20: Samples generated by SDXL-DMD modulated with latent directions discovered by **ELROND (PCA)** within the distilled model itself.

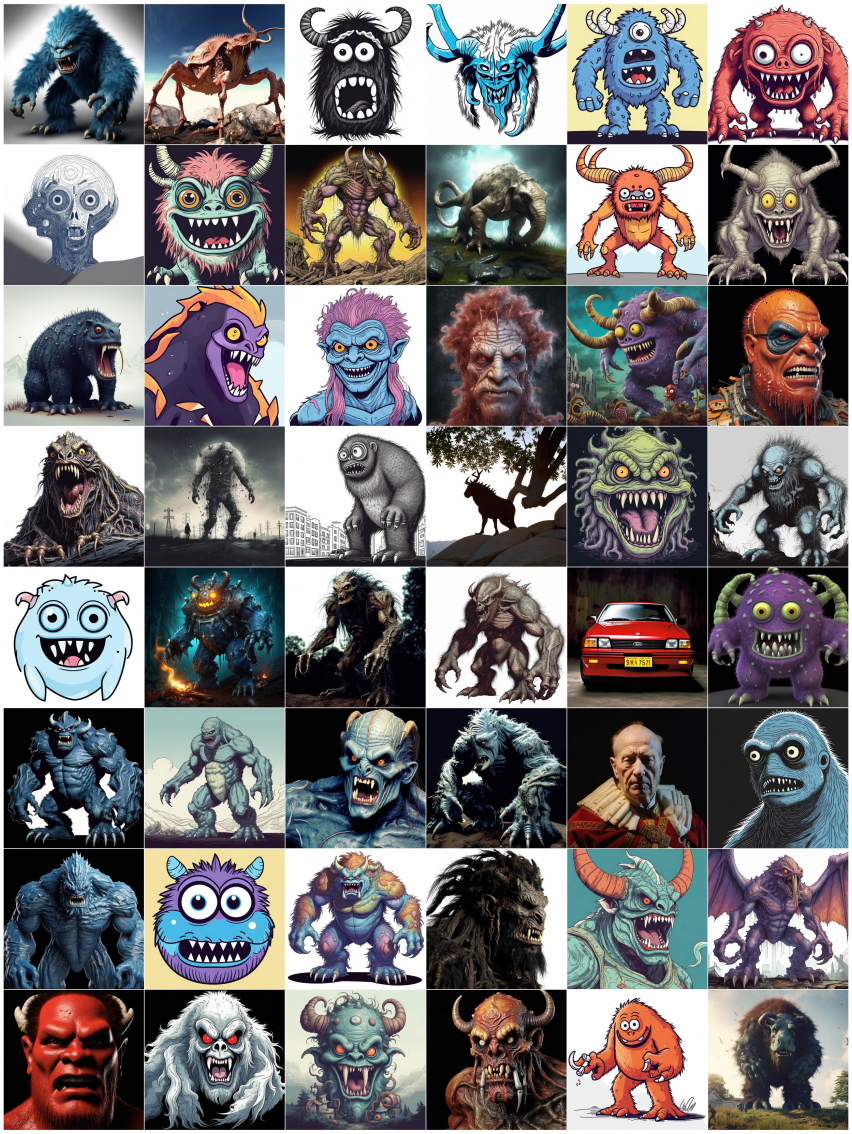

Figure 21: Samples generated by SDXL-DMD modulated with latent directions discovered by **ELROND (SAE)** within the distilled model itself.

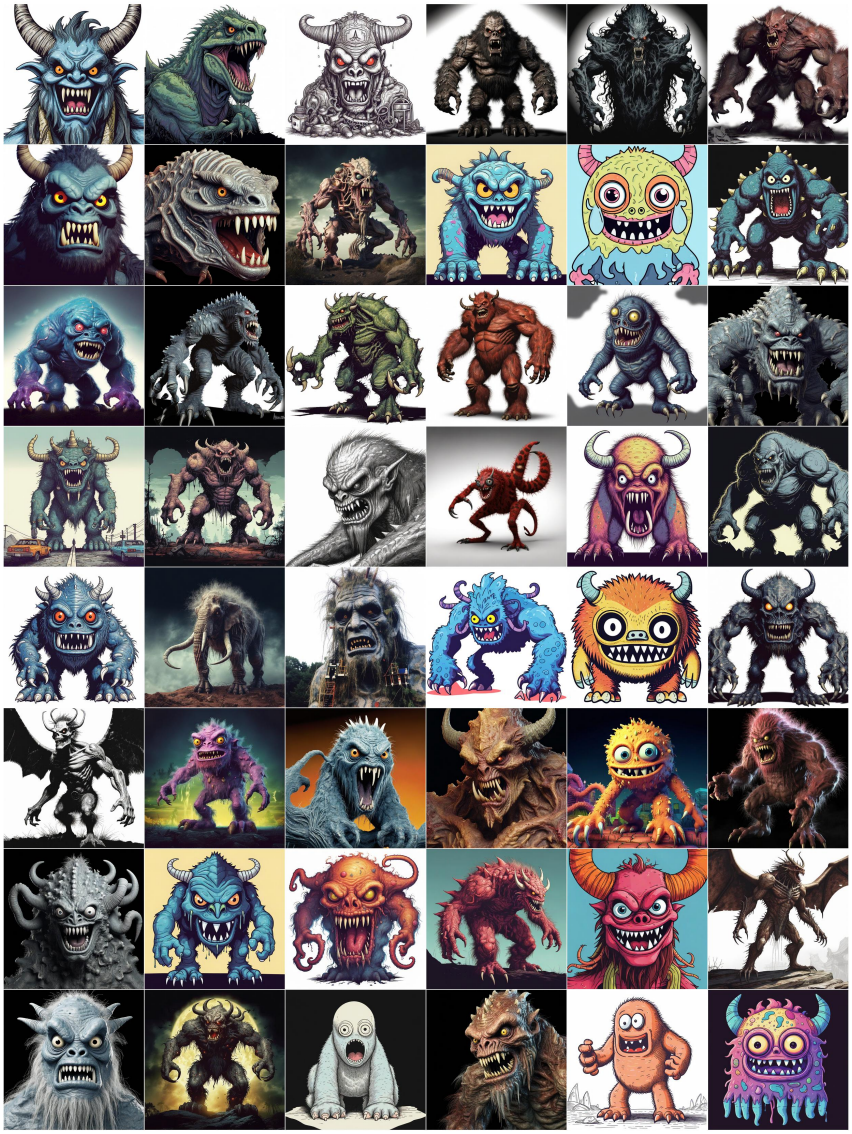

Figure 22: Samples generated by SDXL-DMD modulated with latent directions discovered by **ELROND (SAE)** within the *teacher* model (SDXL).