CONCEPT BOTTLENECK DIFFUSION FOR STEERABLE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has brought the concept bottleneck architecture to the generative modeling space, adding a bottleneck inside of the typical model architecture to force interpretable reasoning. While this approach is flexible and can be applied to a variety of generative models, such architectural bottlenecks are better suited for single-step models like GANs than for diffusion models, which operate across a sequence of timesteps. This mismatch limits steerability and interpretability over the full generative process. In contrast, we place the bottleneck directly in the latent space used by a latent diffusion model, performing denoising along concept channels rather than in pixel space. By reshaping the latent space in which diffusion occurs, we learn a denoising process that is inherently aligned with the concepts we aim to control, enabled by our novel concept masking procedure. Our method, `CBDiffuse`, achieves improved steerability and control compared to prior work across CelebA-HQ and CUB.

1 INTRODUCTION

Generative models excel at producing realistic and high-quality images (Ho et al., 2020; Rombach et al., 2022; Podell et al., 2023). As these models improve, a key challenge is controlling generation to meet specific user requirements, enabling predictable interventions and more reliable outputs. Interpretability offers a natural mechanism for achieving this control, as it aligns internal model representations with human-understandable concepts, providing structured handles for intervention.

Concept Bottleneck Generative Models (CBGMs) (Ismail et al., 2023; Kulkarni et al., 2025b; Anonymous, 2026) present one avenue for leveraging interpretability to increase control in generative models. These works extend the classic Concept Bottleneck Model idea (Koh et al., 2020) to generative architectures such as GANs, VAEs, and diffusion models. By introducing intermediate concept representations, CBGMs provide a way to steer generation through interpretable, concept-level interventions.

However, these methods all operate within the model architecture itself, attempting to build a generative model bottleneck that is agnostic to the underlying generation process. While this design works well for GANs, which produce images in a single forward pass, it is less effective for diffusion models, where samples are produced through an iterative denoising trajectory. Concept bottlenecks in diffusion often influence generation only indirectly: the model still predicts pixel-space noise, so concept signals provide limited control over the output, acting as a conditioning signal rather than a structural constraint.

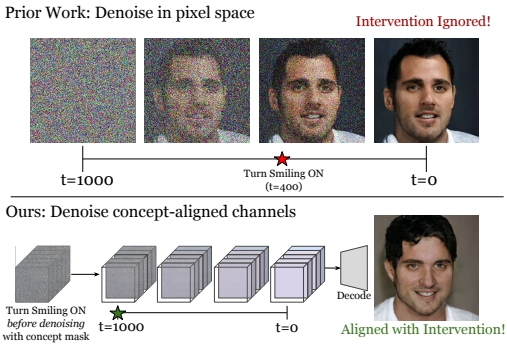

Figure 1: Prior work intervenes in pixel space partway through denoising, making it harder for the U-Net to propagate concept information and sometimes leaving the final image unchanged. Our method masks concept-aligned latent channels before denoising, ensuring the trajectory follows the intervention and the resulting image faithfully reflects the intended concepts.

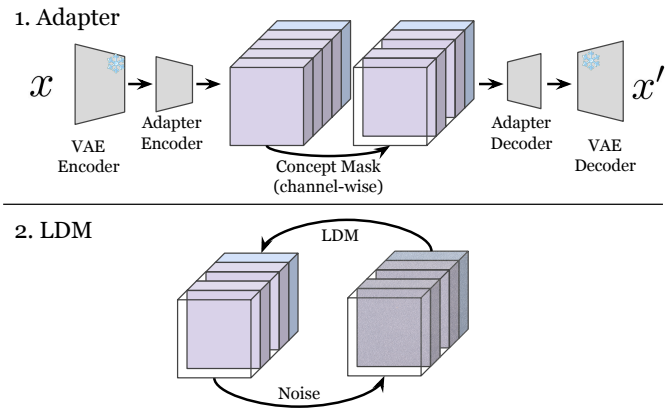

Figure 2: Overview of the `CBDiffuse` architecture. (1) A frozen VAE encoder produces a spatial latent that is mapped by a lightweight adapter into our concept-aligned latent space. Channels in this space are grouped into concept-specific blocks (in purple) and a residual block (in blue). Concept blocks encode discrete semantic states and are selectively masked based on the given concept values, while the residual block preserves non-concept information. (2) The latent diffusion model operates on the resulting masked representation, learning to denoise along active concept blocks and ignore masked channels.

We instead place the concept bottleneck directly in latent diffusion space and train a latent diffusion model to perform concept-aligned denoising in this interpretable space. To do so, we first train a lightweight adapter that maps from a pretrained, frozen VAE's latent space into our concept-structured latent space, and then perform diffusion entirely within this space. Figure 1 illustrates this contrast: prior diffusion-based bottlenecks provide only partial control over generation, whereas our approach enables substantially improved concept-aligned steering by design.

To enforce controllability, we apply channel masking during training based on ground-truth concept values. Each concept is assigned a dedicated subset of channels, and only the channels corresponding to the active concept labels are unmasked. This encourages the model to encode each concept in its assigned channels and ensures that denoising progresses in a concept-aligned manner.

By structuring the latent space around semantic concepts and constraining denoising along concept-specific channels, our framework produces interpretable representations and enables predictable, controllable interventions. Our contributions:

- We propose a new framework for interpretable diffusion that introduces a concept bottleneck directly in the latent diffusion space, enabling concept-aligned denoising rather than pixel-space conditioning.
- We introduce a concept-aware channel masking mechanism that enforces semantic structure throughout the diffusion process, resulting in more reliable and consistent interventions.
- We demonstrate substantially improved steerability on CelebA-HQ and CUB compared to prior diffusion-based concept bottleneck methods.

2 CBDiffuse

The goal of interpretable generation is to enable control over the generative process using human-understandable concepts, producing images consistent with specified concept conditions. In this work we introduce `CBDiffuse`, an interpretable diffusion framework that performs denoising in a concept-aligned latent space before decoding back to pixel space (Figure 2). By organizing latent channels into semantically meaningful groups and enabling selective concept masking, our proposed architecture provides improved controllability compared to prior work.

In Section 2.1, we first describe the `CBDiffuse` architecture, which is built around concept channel masking. We then present our training algorithm (Section 2.2): including a minimal core version that relies on concept masking, followed by a full version that additionally incorporates techniques from prior work.

2.1 ARCHITECTURE

`CBDiffuse` operates by performing diffusion in an interpretable latent space rather than directly in pixel space, unlike the standard DDPM (Ho et al., 2020) architecture used in prior work (Ismail et al., 2023; Kulkarni et al., 2025b; Anonymous, 2026). To do this, we adopt a latent diffusion framework (Rombach et al., 2022), enabling the model to learn a denoising process that is explicitly aligned with human-understandable concepts.

As is typical for latent diffusion, we use a pretrained frozen VAE. We also introduce a lightweight adapter (Figure 2, Adapter). Our adapter is an autoencoder that maps VAE latents to our new interpretable latent space, with channels aligned to human-understandable concepts. Within this interpretable space, we train a latent diffusion model (LDM) to perform denoising directly on concept-aligned latents (Figure 2, LDM) instead of pixels.

Notation & Latent Structure. Let $x \in \mathcal{X}$ denote an input image, and let $c = \{c_1, \ldots, c_K\}$ denote the K human-interpretable concepts associated with x. Each concept c_k takes values in a discrete domain (e.g., binary or categorical), and concept labels may be provided directly or obtained from external predictors such as CLIP.

First, the frozen VAE encoder E_{vae} maps images to a spatial latent representation z, and the VAE decoder D_{vae} reconstructs the image from z: $z = E_{\text{vae}}(x), \quad \hat{x} = D_{\text{vae}}(z), \quad z \in \mathbb{R}^{H \times W \times C_{vae}}$ where H and W denote the height and width of the latent spatial map, respectively.

We then introduce a lightweight adapter to map the VAE latent space to an interpretable latent space. The adapter encoder E_{adapter} and decoder D_{adapter} transform the latent z into an interpretable latent s and back: $s = E_{\text{adapter}}(z), \quad \hat{z} = D_{\text{adapter}}(s), s \in \mathbb{R}^{H \times W \times C_{\text{adapter}}}$.

For binary concepts, the interpretable latent s is partitioned into $2K$ concept blocks and a residual block

$$s = \left[s^{1+}, s^{1-}, \ldots, s^{K+}, s^{K-}, s^{(\text{res})}\right],$$

where each concept c_k is represented by two groups of C_{concept} channels: s^{k+} encodes the presence of the concept, and s^{k-} encodes its absence. The residual block $s^{(\text{res})}$ with C_{res} channels preserves non-conceptual information. More generally, for a concept c_k with n_k discrete categories, we allocate one group of C_{concept} channels per category. The total number of adapter channels is therefore

$$C_{\text{adapter}} = \sum_{k=1}^{K} n_k \, C_{\text{concept}} + C_{\text{res}}.$$

Concept Channel Masking. To enforce concept alignment, we define a binary mask $M(c) \in \{0,1\}^{H \times W \times C_{\text{adapter}}}$ based on the concept assignment c for a given sample x.

For each concept, the mask activates only the channel group corresponding to the specified concept value, while all other concept groups are zeroed out. The masked latent is produced as follows $\tilde{s} = M(c) \odot s$, where $\odot$ denotes element-wise multiplication. Residual channels are always unmasked. During training this encourages each concept block to represent only its intended semantic attribute.

Finally, we parameterize the latent diffusion model as a denoising network ϵ_θ, which operates in the interpretable latent space and predicts added noise at each diffusion timestep. The model takes as input a masked latent, and the diffusion timestep $\epsilon_\theta : (\tilde{s}_t, t) \mapsto \hat{\epsilon}$.

Generation. To generate new samples, we first sample a concept assignment c and gaussian noise $s_T \sim \mathcal{N}(0, I)$ in the interpretable latent space. The concept mask $M(c)$ then produces the masked latent $\tilde{s}_T$, which is iteratively denoised by the latent diffusion model $\epsilon_\theta(\tilde{s}_t, t)$ to yield the concept-aligned latent $\hat{s}_0$. The final image is decoded via $\hat{x} = D_{\text{vae}}(D_{\text{adapter}}(\hat{s}_0))$.

Table 1: Steerability comparisons for diffusion-based methods on CelebA-HQ (8 concepts). Results for CBDiffuse are computed on 5k samples. Results for CBDiffuse are reported on 256×256 images. CBGM and CB-AE results are taken from their reported results since they do not share publicly available diffusion code, and were evaluated at 64×64 instead of 256×256 for CUB.

Steerability (%)	CelebA-HQ	CUB
CBGM	–	14.8
CB-AE	23.5	25.6
CBDiffuse (Core)	**67.6**	**41.5**
CB-AE + OptInt	50.5	36.9
CC + OptInt	56.7	44.5
CBDiffuse (Full) + OptInt	**82.0**	**44.6**

Intervention. To perform concept intervention, we replace the original concept assignment c_1 with a new assignment c_2 while keeping the same noise latent s_T. We construct the corresponding mask and obtain the masked latent $\tilde{s}'_T = s_T \odot M(c_2)$, which is denoised by the latent diffusion model $\epsilon_\theta(\tilde{s}'_t, t)$ to yield $\hat{s}'_0$. The intervened image is then decoded as $\hat{x}' = D_{\text{vae}}(D_{\text{adapter}}(\hat{s}'_0))$.

2.2 Training

Training proceeds in two stages. First, we pretrain the adapter to shape the latent space into a concept-aligned representation, ensuring that semantic concepts are disentangled prior to diffusion training. Second, we jointly train the adapter and the latent diffusion model (LDM) to guide the denoising process along these concept-aligned channels.

We first present our *core* algorithm, which relies solely on concept masking and an orthogonality loss. We then describe a *full* variant that incorporates additional auxiliary losses inspired by prior work, including concept classification losses, intervention losses, and test-time optimization-based interventions. We briefly describe each below, see Appendix C for full details.

CBDiffuse (Core). We first train the core variant of our model to isolate the effect of the proposed architecture on downstream steerability. We train `CBDiffuse` (Core) with the following objective:

$$\mathcal{L}_{\text{core}} = \mathcal{L}_{\text{recons}} + \mathcal{L}_{\text{LDM}} + \lambda_{\text{orth}}\mathcal{L}_{\text{orth}}$$

CBDiffuse (Full). While our core architecture is already effective at learning steerable concept representations, it can be further enhanced with auxiliary losses introduced in prior work. Importantly, our main contribution lies in the architectural design and masking procedure of `CBDiffuse`, which remains orthogonal to these additions. For CBDiffuse (Full), we train with the following objective:

$$\mathcal{L}_{\text{full}} = \mathcal{L}_{\text{core}} + \lambda_{\text{concept}}\mathcal{L}_{\text{concept}} + \lambda_{\text{intervention}}\mathcal{L}_{\text{intervention}}.$$

3 Results

We first describe general details about our experimental setup in Section 3.1, and then aim to answer the following questions: **Q1:** How controllable is `CBDiffuse` compared to prior work on interpretable generative modeling (Section 3.2)? **Q2:** How do `CBDiffuse` (Full) losses impact the downstream steerability and image quality (Section 3.3)?

3.1 Experimental Setup

Architecture Details. We use a pretrained E2E-VAVAE-HF model from Leng et al. (2025) with a latent space of size $16 \times 16 \times 32$. The adapter consists of four residual convolutional blocks in both the encoder and decoder. As the latent diffusion backbone, we use SiT-B/1 (Ma et al., 2024). For concept classification, we train lightweight linear heads for each concept. Additional architectural and training details are provided in Appendix A.

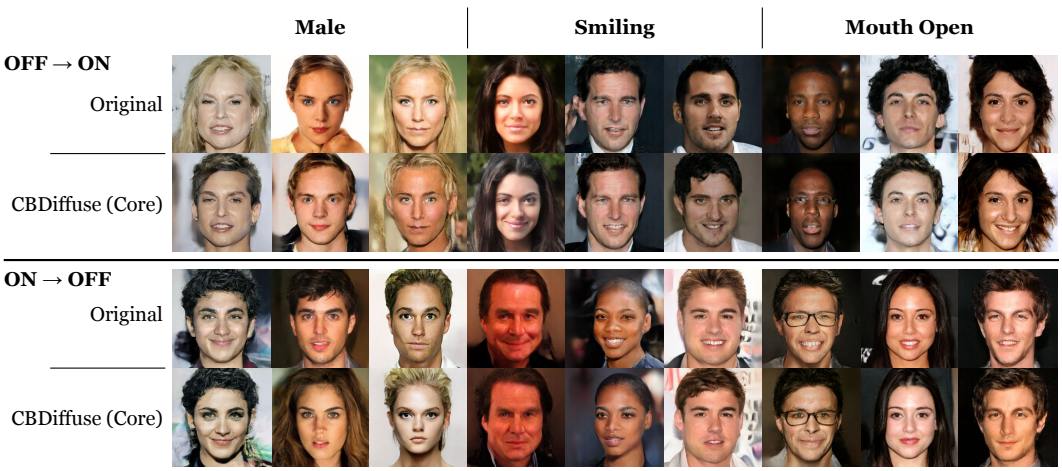

Figure 3: Concept intervention results for `Male`, `Smiling`, and `Mouth Open`. The top half shows images where the concept is flipped from OFF to ON, while the bottom half shows flips from ON to OFF. For each concept, the first row displays the original images and the second row shows the corresponding images generated by `CBDiffuse` (Core) after intervention.

Datasets. We evaluate our approach on CelebA-HQ (Lee et al., 2020) using eight binary concepts (Smiling, Male, Heavy Makeup, Mouth Slightly Open, Attractive, Wearing Lipstick, High Cheekbones, Arched Eyebrows), and on CUB (Wah et al., 2011) using ten visual attributes (Small size (5–9 inches), Perching-like shape, Solid breast pattern, Black bill color, Bill length shorter than head, Black wing color, Solid belly pattern, All-purpose bill shape, Black upperparts color, White underparts color). For both datasets, all images are resized to 256×256 to match the resolution of the pretrained VAE.

Training. For both models, we first pretrain the adapter for 40K steps, followed by joint training of the full model for 400K steps. For `CBDiffuse` (Full), additional loss terms are incorporated during joint training. Further training details and loss weights are provided in Appendix A.

Concept Evaluation. We automate concept evaluation using ViT-L-16 binary classifiers trained to detect each concept in an image. For CelebA-HQ, we use the same classifiers as Kulkarni et al. (2025b) to ensure fair comparison. Since concept classifiers for CUB are not publicly available, we train our own using the original authors' code.

Baselines. We compare against two primary baselines: CBGM Ismail et al. (2023) and CB-AE Kulkarni et al. (2025b). Since neither provides code for replicating their diffusion model results, we report their published numbers directly.

3.2 Steerability Results

To evaluate how controllable `CBDiffuse` is compared to prior work, we measure the model's *steerability*. Following prior work, we generate samples in both directions: first with a concept OFF and then flipping it ON, and second with a concept ON and then flipping it OFF. For each flip, an external concept classifier determines whether the generated image reflects the intended change. Steerability is reported as the percentage of samples where the concept successfully flips, averaged over all concepts and directions.

Results are shown in Table 1. The core model (`CBDiffuse` Core) improves over CB-AE by 44.1% on CelebA-HQ and 15.9% on CUB, demonstrating that the masked reconstruction and denoising losses alone provide strong control. The full model (`CBDiffuse` Full) further increases steerability by 14.4% on CelebA-HQ and 3.1% on CUB, showing that these gains are complementary to prior approaches. Overall, these results emphasize that learning an interpretable latent space is crucial for precise control in diffusion models.

Table 2: Stepwise ablation of components on CelebA-HQ. Concept classifiers are required to perform interventions or apply OptInt. Steerability steadily improves as components are added. The intervention loss slightly increases FID, reflecting a trade-off between image quality and controllability. For reference, a standard LDM with the same hyperparameters achieves an FID of 24.9.

$L_{concept}$	$L_{intervene}$	OptInt	Conc. Acc. (%)	Steerability (%)	FID (↓)
×	×	×	-	67.6	29.0
✓	×	×	74.5	66.3	**24.6**
✓	×	✓	74.5	74.6	24.6
✓	✓	×	**80.9**	78.4	33.9
✓	✓	✓	**80.9**	**82.0**	33.9

Figure 3 illustrate the visual effects of our approach, presenting interventions on the concepts *Male*, *Smiling*, and *Mouth Open* using `CBDiffuse` (Core), demonstrating consistent and precise control over multiple attributes.

3.3 CBDiffuse Losses

In this section, we perform a stepwise ablation of the losses that differentiate `CBDiffuse` (Full) from `CBDiffuse` (Core): the concept loss ($L_{concept}$), the intervention loss ($L_{intervene}$), and optimized interventions (OptInt). We evaluate these models on three metrics: concept accuracy, steerability, and generation quality. Concept accuracy is measured by randomly sampling a concept assignment, generating an image conditioned on the corresponding mask, and assessing whether the generated image exhibits the intended concept; results are averaged across all concepts. Generation quality is evaluated using the Fréchet Inception Distance (FID) (Heusel et al., 2017), computed over 10K images generated by sampling concept masks according to their empirical distribution in the training data.

Our results are shown in Table 2. For reference, the baseline SiT model trained with the same pretrained VAE achieves an FID of 24.9. Incorporating the concept loss ($L_{concept}$) slightly improves FID compared to other variants, likely due to the additional supervision it provides, closely matching the FID of the baseline latent diffusion model. While the concept loss alone does not improve steerability, it enables the use of optimized interventions (OptInt), which further increase steerability beyond the core model. In contrast, adding the intervention loss ($L_{intervene}$) slightly degrades FID, likely because it emphasizes learning a concept-aligned latent space at the expense of image quality. However, it substantially boosts steerability compared to using only the concept loss, by +12.1% without OptInt and +7.4% with OptInt.

Optimized interventions (OptInt) consistently improve steerability, yielding an increase of 8.3% when applied without the intervention loss and 3.6% when used with the full model. The smaller absolute gain for the full model is expected, as higher baseline steerability leaves less room for improvement. This trend aligns with observations from CB-AE (Kulkarni et al., 2025b), where OptInt improved steerability by 27%, starting from a relatively low baseline of 23.5%.

4 Conclusions

We introduced `CBDiffuse`, an interpretable diffusion framework that places the bottleneck directly in the latent space instead of in the diffusion model architecture. By training an interpretable adapter and enforcing concept-specific channel masking throughout the denoising process, our approach aligns semantic structure with diffusion dynamics. This design enables concept-aligned denoising and yields substantially improved steerability compared to prior concept bottleneck generative models.

Empirically, we demonstrated strong gains in steerability on CelebA-HQ and CUB, with ablations showing that channel masking alone provides effective control while remaining complementary to existing intervention losses and optimization-based techniques. While this approach may trade off some fidelity in exact pixel-level edits, it allows for more reliable and precise interventions at the semantic level, highlighting the benefits of embedding interpretable structure directly into the generative process.

REFERENCES

Anonymous. A probabilistic hard concept bottleneck for steerable generative models. 2026. Accepted submission.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, et al. Concept embedding models: Beyond the accuracy-explainability trade-off. *Advances in neural information processing systems*, 35:21400–21413, 2022.

Rohit Gandikota, Joanna Materzynska, 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.

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.

Ada Gorgun, Fawaz Sammani, Nikos Deligiannis, Bernt Schiele, and Jonas Fischer. Temporal concept dynamics in diffusion models via prompt-conditioned interventions. *arXiv preprint arXiv:2512.08486*, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Aya Abdelsalam Ismail, Julius Adebayo, Hector Corrada Bravo, Stephen Ra, and Kyunghyun Cho. Concept bottleneck generative models. In *The Twelfth International Conference on Learning Representations*, 2023.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pp. 5338–5348. PMLR, 2020.

Akshay Kulkarni, Tsui-Wei Weng, Vivek Narayanaswamy, Shusen Liu, Wesam A Sakla, and Kowshik Thopalli. Interpretable and steerable concept bottleneck sparse autoencoders. *arXiv preprint arXiv:2512.10805*, 2025a.

Akshay Kulkarni, Ge Yan, Chung-En Sun, Tuomas Oikarinen, and Tsui-Wei Weng. Interpretable generative models through post-hoc concept bottlenecks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8162–8171, 2025b.

Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5549–5558, 2020.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers. *arXiv preprint arXiv:2504.10483*, 2025.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

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.

Kanchana Ranasinghe, Muzammal Naseer, Munawar Hayat, Salman Khan, and Fahad Shahbaz Khan. Orthogonal projection loss. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12333–12343, 2021.

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.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

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 Experimental Details

A.1 Architecture

VAE. We use a pretrained E2E-VAVAE-HF that maps input images of size $256 \times 256 \times 3$ to a spatial latent representation

$$z \in \mathbb{R}^{H \times W \times C_{\text{vae}}}, \quad H = W = 16, \quad C_{\text{vae}} = 32,$$

and reconstructs images as $\hat{x} = D_{\text{vae}}(z)$.

Latent Adapter. The adapter maps the VAE latent space to an interpretable, concept-aligned space:

$$f : \mathbb{R}^{16 \times 16 \times 32} \to \mathbb{R}^{16 \times 16 \times C_{\text{concept}}},$$

where

$C_{\text{concept}} = (\text{channels per concept}) \times (\text{num concepts}) \times (\text{categories per concept}) + (\text{residual channels}).$

Table 3: Latent space and concept channel specifications for CelebA-HQ and CUB.

Dataset	VAE Latent ($H \times W \times C_{\text{vae}}$)	Adapter Latent ($H \times W \times C_{\text{concept}}$)	Concept Channels
CelebA-HQ	$16 \times 16 \times 32$	$16 \times 16 \times 72$	8 concepts, 4 channels per concept, 8 residual channels
CUB	$16 \times 16 \times 32$	$16 \times 16 \times 88$	10 concepts, 4 channels per concept, 8 residual channels

The adapter consists of an encoder and decoder with 4 residual convolution blocks each, 256 hidden channels, 3×3 convolutions, GroupNorm, and ReLU activations, totaling 9.76M parameters. The `CBDiffuse` latent diffusion model uses a SiT-B/1 backbone with 138M parameters.

A.2 Training and Losses

Adapter pretraining is performed for 40K steps, followed by joint training with the diffusion model for 400K steps. Batch size is 64. AdamW optimizers with learning rates of 1×10^{-4} are used for both adapter and diffusion model.

- **CBDiffuse (Core):** masked reconstruction and denoising losses.
- **CBDiffuse (Full):** concept loss ($\lambda_{\text{concept}} = 1$), intervention loss ($\lambda_{\text{intervene}} = 0.1$), orthogonality regularization ($\lambda_{\text{orthog}} = 0.1$), optionally combined with optimization-based intervention (OptInt).

The cyclical intervention loss is applied every 25 iterations to reduce computational cost. OptInt performs 50 optimization steps with step size $\epsilon = 0.1$ during inference to refine latent concept activations.

A.3 Evaluation and Hardware

For each concept assignment, 500 samples are generated. Concept accuracy is measured using external ViT-L-16 classifiers. Steerability is computed by flipping concepts ON/OFF and verifying that generated images reflect the intended change. FID is computed over 10K images sampled according to the empirical concept distribution.

All experiments were conducted on a node with 112 CPU cores, 1TB RAM, and 2×NVIDIA RTX 6000 Ada GPUs.

B Related Work

B.1 Concept Bottleneck Generative Models (CBGMs)

Our work builds on recent advances in interpretable generative modeling using the concept bottleneck architecture (Koh et al., 2020). Ismail et al. (2023) first introduced a concept embedding layer

(Espinosa Zarlenga et al., 2022) into generative models, enabling interventions to guide image generation.

Kulkarni et al. (2025b) extended this by proposing a post-hoc CBGM, which learns a mapping from a frozen pre-trained generative model to interpretable concept representations. They also introduced a concept controller strategy that allows for steerability without modifying the underlying latent. Kulkarni et al. (2025a) further explored combining concept bottlenecks with sparse autoencoders to balance supervised and unsupervised concept learning in large vision-language models. Most recently, concurrent work (Anonymous, 2026) added "hard" concept constraints to reduce concept leakage and improve steerability.

Our Approach. Prior CBGM work focuses on inserting bottlenecks within the generative model architecture itself. While effective for GANs or single-step models, this design provides limited interpretability and control in diffusion models. Interventions applied mid-denoising in pixel space are difficult for the U-Net to propagate, often leaving the final image largely unchanged. `CBDiffuse` addresses this by mapping VAE latents to a concept-aligned latent space and performing denoising directly along concept channels. By structuring the diffusion process around these channels, interventions are faithfully propagated throughout the denoising trajectory, enabling more predictable concept-level control.

B.2 Semantic Control in Diffusion Models

Controlling semantic information in diffusion models has become a major area of research. Early approaches include *classifier guidance* (Dhariwal & Nichol, 2021) and *classifier-free guidance* (Ho & Salimans, 2022), which steer the generation process by adjusting the conditioning signal during denoising. While effective for coarse control, these methods operate indirectly and do not enforce a structured latent representation for specific concepts, limiting the precision and predictability of interventions. Text-conditioned diffusion models and ControlNet (Rombach et al., 2022; Zhang et al., 2023) extend this idea by learning mappings from text prompts or structured conditions to image outputs. These approaches achieve impressive results at scale, particularly for high-level attributes like scene composition or object presence. However, control remains largely post-hoc: the model still denoises in pixel or latent space without an explicit semantic structure, and concepts can become locked at specific timesteps along the denoising trajectory (Gorgun et al., 2025). Other work explores mechanistic interpretability techniques for more targeted semantic control. Gandikota et al. (2023) propose erasing specific visual concepts directly from a pretrained diffusion model, effectively removing objects, styles, or attributes from the model itself rather than merely filtering outputs at inference. Building on this idea, Gandikota et al. (2024) introduce Unified Concept Editing (UCE), which enables simultaneous editing of multiple concepts, such as debiasing or content manipulation, by intervening in model parameters in a closed-form, scalable manner.

Our Approach. In contrast, `CBDiffuse` focuses on training semantic control into the diffusion model from the start, rather than adapting an already pretrained model. By mapping a pretrained VAE's latent space into concept-aligned channels and applying concept-aware masking during training, we guide the denoising trajectory along desired concepts. This enables concept-aligned generation and precise interventions at smaller scale, making it more accessible for specialized domains or experiments where large pretrained models are impractical.

C Loss Definitions

Masked Reconstruction Loss ($\mathcal{L}_{\text{recons}}$). To encourage the adapter to learn an interpretable latent space, we apply a masked reconstruction loss. By reconstructing latents from only the channels corresponding to active concepts in each image, the adapter is incentivized to isolate information about each concept into its designated channels.

Formally, given an input image $x \sim \mathcal{X}$ with concept assignment c, we first encode x using the frozen VAE: $z = E_{\text{vae}}(x)$. We then encode the latent with the adapter encoder $s = E_{adapter}(z)$, apply the concept-specific mask $M(c)$ to the resulting latent, and pass the masked latent through the adapter decoder to reconstruct the original latent. The loss is the mean squared error (MSE) between the original and reconstructed latents:

$$\mathcal{L}_{\text{recons}} = \mathbb{E}_{x\sim\mathcal{X}}\Big[\big\|D_{\text{adapter}}\big(s \odot M(c)\big) - z\big\|_2^2\Big].$$

Masked Denoising Loss ($\mathcal{L}_{\text{LDM}}$). To align the generation process with the interpretable latent space learned by the adapter, we train the latent diffusion model (LDM) using a masked denoising loss. By restricting denoising to the unmasked concept channels, the LDM is guided to produce images consistent with the specified concept mask.

At each diffusion timestep t, Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ is added to the adapter latent s_0 to obtain $s_t = s_0 + \epsilon$. The masked latent $\tilde{s}_t = s_t \odot M(c)$ is then fed to the LDM, which predicts the added noise $\hat{\epsilon} = \epsilon_\theta(\tilde{s}_t, t, c)$. The masked denoising loss is computed only over the unmasked channels, as

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{x,\epsilon,t}\Big[\big\|(\epsilon - \epsilon_\theta(\tilde{s}_t, t)) \odot M(c)\big\|_2^2\Big].$$

Orthogonality Loss ($\mathcal{L}_{\text{orth}}$). Known concepts rarely capture all semantic variation, so we include residual channels to model unknown concepts. To prevent the adapter from encoding known concepts into these residual channels, we introduce an orthogonality loss (Ranasinghe et al., 2021; Ismail et al., 2023) between each concept block $s^{(k)}$ and the residual block $s^{(\text{res})}$.

Formally, we compute the absolute cosine similarity between concept blocks and the residual block:

$$\mathcal{L}_{\text{orth}} = \mathbb{E}_{s\sim\mathcal{S}}\Big[\sum_{k=1}^{K}\Big|\frac{\langle s^{(k)}, s^{(\text{res})}\rangle}{\|s^{(k)}\|_2\,\|s^{(\text{res})}\|_2}\Big|\Big].$$

Concept Classifiers + Concept Loss ($\mathcal{L}_{\text{concept}}$). To ensure each concept block reliably encodes its intended concept, we introduce a binary concept classifier for each block. This encourages the adapter to disentangle concepts and aligns latent representations with their semantic meaning.

For a binary concept k, we summarize the block's features for both the active and inactive instances (by averaging over spatial dimensions) and concatenate them into a single vector:

$$\bar{s}^k = [\,\bar{s}^{k+}; \bar{s}^{k-}\,] \in \mathbb{R}^{2C_k}.$$

The classifier predicts which instance is active:

$$c_{\text{pred}}^{(k)} = q_k(\bar{s}^k) \in [0,1], \quad q_k : \mathbb{R}^{2C_k} \to [0,1].$$

We train the concept classifiers with a concept loss, the binary cross-entropy between the predicted label and the ground truth, summed over all concept blocks:

$$\mathcal{L}_{\text{concept}} = \mathbb{E}_{x\sim\mathcal{X}}\left[\sum_{k=1}^{K}\text{BCE}\big(c_{\text{pred}}^{(k)}(x), y_{\text{GT}}^{(k)}(x)\big)\right].$$

This formulation can be easily extended to multi-class concepts by letting q_k output a probability distribution over the concept categories.

Cyclical Intervention Loss ($\mathcal{L}_{\text{intervention}}$). To enforce that interventions on concept assignments produce the intended effect, we use a cyclical intervention loss (Kulkarni et al., 2025b). This

loss encourages the model to produce latent representations that respond predictably to manual interventions, reinforcing steerability.

Concretely, given a latent s, we first remask it according to the intended intervention labels, denoise it through the LDM, and then decode and re-encode it via the adapter. Each concept block of the resulting latent is then passed through its classifier q_k to obtain predicted labels $c_{\text{pred}}^{(k)}$.

The intervention loss is the cross-entropy between these predicted labels and the intended intervention labels, summed over all concepts and averaged over the dataset:

$$\mathcal{L}_{\text{intervention}} = \mathbb{E}_{x\sim\mathcal{X}}\left[\sum_{k=1}^{K} \text{CE}\big(c_{\text{pred}}^{(k)}(x), c_{\text{intv}}^{(k)}(x)\big)\right].$$

Optimization-based Intervention. Finally, we can also perform optimization-based intervention at inference time, following Wong et al. (2020); Kulkarni et al. (2025b). This allows us to directly manipulate concept activations in the latent space to achieve the desired intervention effect in the generated image.

Given a generated latent $\hat{s}_0$ with concept assignment c, we optimize the latent to match a target concept assignment c^* while staying close to the original latent:

$$\tilde{s}_0^* = \tilde{s}_0 + \arg\max_{\delta\in\Delta}\Big[-\mathcal{L}_{\text{concept}}\big(E_{\text{adapter}}(D_{\text{adapter}}(\tilde{s}_0+\delta)), c^*\big)\Big],$$

where $\Delta = \{\delta : \|\delta\|_\infty \le \epsilon\}$ constrains the perturbation magnitude, and $\mathcal{L}_{\text{concept}}$ is the cross-entropy loss computed via the concept classifiers.

The final manipulated image is then obtained by decoding the optimized latent:

$$\hat{x}^* = D_{\text{vae}}(D_{\text{adapter}}(\tilde{s}_0^*)).$$

D Additional Results

We provide additional example interventions for our `CBDiffuse` (Full) model. While image quality may be slightly lower, concept steerability is improved.

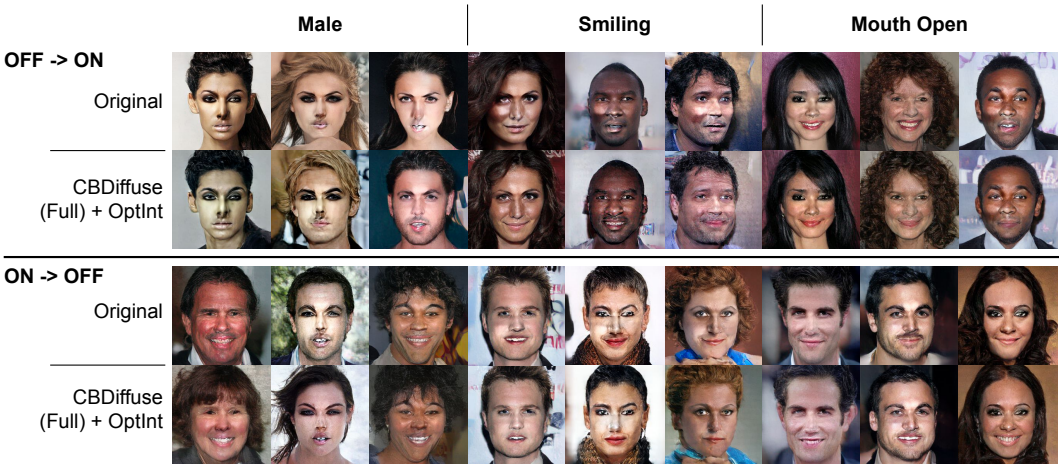

Figure 4: Concept intervention results for `Male`, `Smiling`, and `Mouth Open`. The top half shows images where the concept is flipped from OFF to ON, while the bottom half shows flips from ON to OFF. For each concept, the first row displays the original images and the second row shows the corresponding images generated by `CBDiffuse` (Full) after intervention.

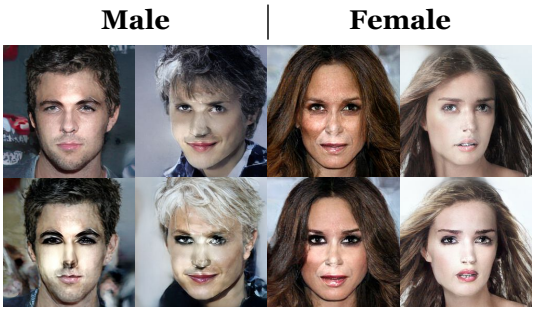

Figure 5: Generation of Heavy Makeup (OFF $\rightarrow$ ON) conditioned on gender using `CBDiffuse` (Full). While the model can generate Heavy Makeup on men despite limited training data, interventions are more challenging for males than for females.

D.1 Concept Combination Biases

Table 4: Per-concept activation steerability (%) conditioned on gender for `CBDiffuse` (Core). Makeup-related concepts exhibit substantially lower steerability for men, while gender neutral concepts have balanced steerability across gender, reflecting dataset imbalance.

Concept (OFF $\rightarrow$ ON)	Male	Female
Heavy Makeup	13.5	55.8
Lipstick	47.7	92.0
Mouth Open	89.6	90.3
Smiling	71.3	65.2

Real-world datasets exhibit strong biases in the joint distribution of concepts, with some combinations appearing far less frequently than others. For instance, in CelebA-HQ, images of men wearing heavy

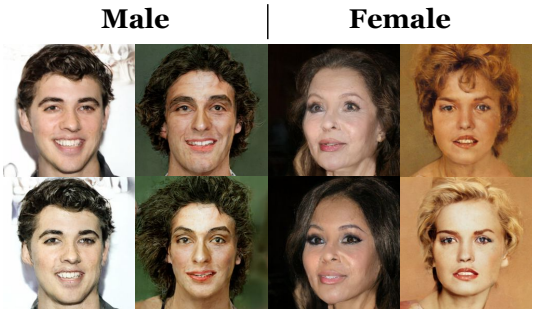

Figure 6: Generation of Heavy Makeup (OFF → ON) conditioned on gender using `CBDiffuse` (Core). While the model can generate Heavy Makeup on men despite limited training data, interventions are more challenging for males than for females.

makeup or lipstick are substantially rarer than the corresponding combinations for women, reflecting underlying cultural and dataset biases. This raises an important question: *how well can generative models handle controlled generation for underrepresented concept combinations?*

In Figure 6, we demonstrate that `CBDiffuse` can generate uncommon concept combinations, such as heavy makeup and lipstick on men, despite limited training data coverage. However, these rare combinations remain more challenging to generate effectively. To quantify this, Table 4 reports per-concept steerability conditioned on gender, focusing on two gender-biased and two gender-neutral concepts. We observe a significant drop in steerability for makeup-related concepts when conditioned on *male* compared to *female*, while gender-neutral concepts maintain balanced controllability across genders.

This behavior highlights a fundamental challenge in compositional generative modeling: although the number of possible concept combinations grows exponentially with the number of concepts, the effective support of the training data covers only a small fraction of this space. Consequently, interventions that push the model toward low-density regions of the joint concept distribution are inherently more difficult. Prior work has noted that enforcing concept changes in diffusion models is particularly challenging due to the iterative nature of the denoising process, which can resist out-of-distribution edits as noise is gradually removed (Anonymous, 2026).

In contrast to approaches that intervene directly in pixel space or late in the generation process, `CBDiffuse` operates in a concept-aligned latent space, allowing concept information to propagate more consistently throughout the denoising trajectory. While this does not eliminate dataset biases, it enables more faithful and stable interventions for rare concept combinations compared to baseline methods.

Overall, these results suggest that, although `CBDiffuse` is still constrained by the underlying concept distribution of the training data, its architecture mitigates some of the challenges associated with rare combinations by explicitly structuring and preserving concept information throughout the diffusion process.