

Compositional Concepts Erasing: Disentangling Object-Style via Domain Manifold Unlearning

Yang Xiao¹, Kaiyuan Deng², Gen Li³, Zinan Ling¹
 Zhiwen Fan⁴, Xiaolong Ma², Bo Hui^{1*}
¹University of Tulsa ²The University of Arizona
³Clemson University ⁴Texas A&M University

Abstract

Diffusion models pose significant risks regarding the generation of illicit content and copyright violations. This has catalyzed the development of machine unlearning to erase harmful concepts or copyrighted styles from a model’s generative repertoire without compromising its general performance. However, previous unlearning approaches on diffusion models have largely focused on either object-level or style-level removal and failed to address the challenge of object-style combination unlearning. At the same time, multiple unlearning requests often lead to catastrophic forgetting. To address these challenges, we propose **Domain Manifold Unlearning (DMU)**, which uses an EMA-updated feature proxy extracted from intermediate diffusion features to represent the retained target-object direction. As a lightweight estimating representation of an object, DMU enables selective unlearning of domain-specific components in a specific object while preserving other semantic representations, thereby significantly enhancing robustness in multiple combination unlearning scenarios. Extensive experiments show that DMU significantly mitigates catastrophic forgetting, outperforming state-of-the-art methods of batch unlearning and sequential unlearning.

1 Introduction

With the rapid advancement of artificial intelligence, the expanding capabilities of foundation models, such as diffusion models, have raised significant concerns regarding data privacy protection, copyright compliance, and safe deployment [1, 2]. For example, users can craft prompts that manipulate these models into producing unethical outputs, such as violent imagery, sexually explicit content, or protected copyrighted material. Driven by regulations, such as the “right to be forgotten”, laws against NSFW and copyright infringement by generative AI [3], machine unlearning has rapidly emerged as a critical pillar [4–12]. This paradigm enables models to selectively erase the influence of specific data or concepts without requiring complete retraining from scratch.

While previous unlearning approaches on diffusion models have largely focused on either object-level or style-level removal, a more challenging setting is to forget **object-style combinations** [9, 10] (e.g., a human in Van Gogh style). Such fine-grained unlearning provides more flexibility and reliability in real applications. For example, in content moderation systems, a model may need to forget specific object–style combinations, such as “weapons” depicted in cartoon styles, while still retaining the ability to recognize weapons in real-world images and benign cartoon content. Removing only the object or the style would lead to over-suppression, highlighting the need for fine-grained unlearning of object–style combinations. As another example, recent controversies around AI-generated images in the style of specific studios or artists (e.g., Studio Ghibli-like renderings) highlight a realistic need for fine-grained control over object–style combinations. In practice, a platform may permit generating images of a public figure or character in generic styles, while restricting the generation of

*Corresponding Author: bo-hui@utulsa.edu

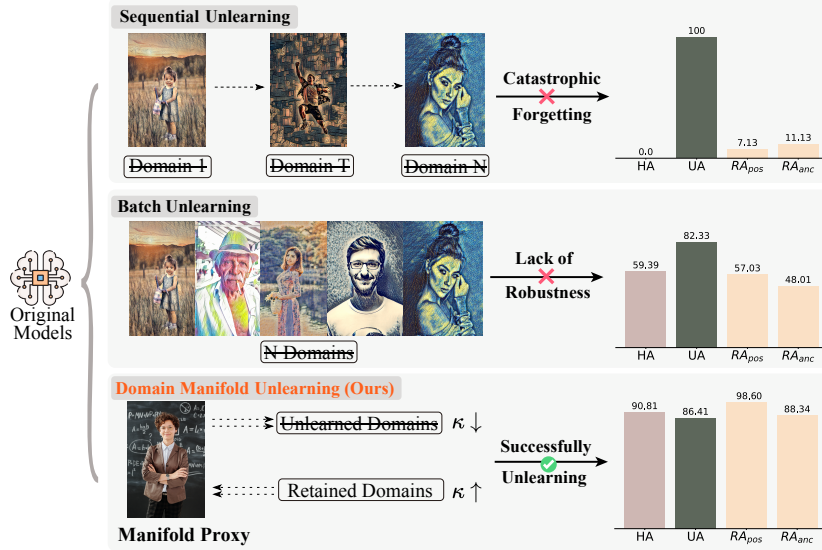


Figure 1: Motivation of our method. HA refers to the harmonic average of performance across the later three metrics: UA refers to 100 minus the accuracy in the negative set (Unlearned domains. RA_{pos} refers to accuracy in the positive set (Retained domains). RA_{anc} refers to accuracy in the anchor set (Other objects in all styles).

the same object in the distinctive style of a protected studio, franchise, or living artist. This requires selectively removing only the prohibited subject–style combination, rather than eliminating the entire object or the entire style. While pre-trained diffusion models exhibit strong generalization across diverse datasets, they often entangle semantic and stylistic representations within the latent space. As a result, existing object-level or style-level unlearning approaches are insufficient to satisfy such fine-grained and selective unlearning requirements.

Existing unlearning methods can be broadly categorized into batch unlearning (BU) and sequential unlearning (SU) [5, 13, 9, 14, 15, 4]. BU removes multiple concepts simultaneously in a single request, while SU processes removal requests incrementally over time. However, neither setting is well suited for fine-grained object–style combination unlearning. There is a formidable challenge of BU: because the unlearning gradients associated with different domain distributions may interfere with one another, a simple linear superposition often results in incomplete unlearning or an excessive degradation of the model’s general utility. Studies have demonstrated that independently designed unlearning operations can clash, causing the model to exhibit extreme instability when processing "hard-to-unlearn" samples that span multiple domains [14, 10]. SU [10, 11] addresses scenarios in which domain unlearning requests arrive in an asynchronous and continuous manner. In this setting, the model must incrementally remove the influence of specific data distributions, such as particular illustration styles, while strictly preserving its diagnostic and generative capabilities on the retained data. Our preliminary studies have consistently observed a significant decline in performance on retained data as the number of unlearning requests increases. As shown in Figure 1, the retaining accuracy of target object (in the remaining style) and other objects dropped catastrophically. This degradation cannot be attributed solely to simple changes in model weights [11], rather, it arises from gradient noise introduced during earlier unlearning operations, which is progressively amplified in subsequent tasks. In the deeply entangled latent space of foundation models, forcefully erasing multiple domains using conventional gradient descent often irreparably damages the underlying representation structure.

To overcome these fundamental limitations, it is essential to move beyond treating multi-target domain unlearning as a simple linear optimization problem. Recognizing that domain characteristics and core semantic knowledge are intricately entangled within a low-dimensional manifold structure in the model’s parameter space, we propose **Domain Manifold Unlearning (DMU)**. Instead of directly applying conflicting or noisy gradients, our method explicitly models the underlying manifold of the target domains. By navigating and decoupling the domain-specific manifolds from the shared semantic representation, DMU effectively neutralizes the gradient conflicts inherent in Batch Unlearning and mitigates the noise amplification in Sequential Unlearning. This approach allows the model to

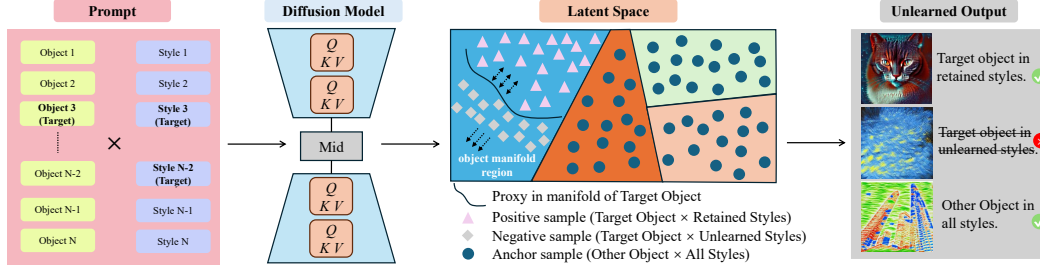


Figure 2: Method

precisely excise the targeted domain distributions while comprehensively preserving its core utility and generalization capabilities. We summarize our contributions as follows:

- Unlike prior methods that only focus on either object or style-level removal, we propose a framework to address object-style combination unlearning.
- Our approach significantly enhances robustness in multiple unlearning requests by directly operating on the domain manifold, providing a more precise way to isolate targeted data.
- Through extensive experiments, we demonstrate that DMU outperforms state-of-the-art methods of batch unlearning and sequential unlearning.

2 Method

2.1 Preliminaries

To achieve unlearning while preserving model performance, the unlearning problem is formulated as an optimization task to update parameters from their pretrained weights. In our setting, we divide the dataset into three components: 1) Positive Set: $\mathcal{D}_{t,r}$ refers to the data of target object on retained styles. 2) Negative Set: $\mathcal{D}_{t,u}$ refers to the data of target object on unlearned styles. 3) Anchor Set: $\mathcal{D}_r$ refers to the data of other objects on all styles:

$$\mathcal{L} = \lambda \mathcal{L}_u(\theta, \mathcal{D}_{t,u}, \mathcal{D}_{t,r}) + \mathcal{L}_r(\theta, \mathcal{D} \setminus (\mathcal{D}_{t,u})) \quad (1)$$

2.2 Non-parametric Object Proxy in the Manifold

To implement the unlearning objective $\mathcal{L}_u$ effectively, we introduce a non-parametric object proxy. As illustrated in Figure 2, given a batch of object–style pairs, our goal is to remove the target object under specific styles while preserving its representation under retained styles. In diffusion models, we extract the feature vector $\mathbf{h}(x)$ from intermediate representations of the U-Net (e.g., bottleneck layers) during the denoising process. We then use these features to approximate an object-specific proxy via dynamically updating.

A key challenge is that, in general, we cannot guarantee that each mini-batch contains sufficient positive samples (i.e., target object under retained styles, $x \in \mathcal{D}_{t,r}$). Directly using per-batch statistics would therefore lead to unstable or even undefined estimates of the proxy. To address this, we adopt an exponential moving average (EMA) to accumulate information across batches, ensuring a stable and consistent estimation.

To further avoid contamination of domains to be unlearned, the EMA is updated exclusively using positive samples. Let i denote the current optimization step. The proxy is updated as:

$$\boldsymbol{\mu}^{(i)} = m \cdot \boldsymbol{\mu}^{(i-1)} + (1 - m) \cdot \mathbb{E}_{x \in \mathcal{D}_{t,r}}[\mathbf{h}^{(i)}(x)], \quad \boldsymbol{\mu}^{(0)} = \mathbf{0} \quad (2)$$

In the special case where each mini-batch is guaranteed to contain sufficient positive samples, the proxy can be estimated directly from the current batch (i.e., setting momentum $m = 0$). In our ablation study, we empirically verify that this simplified variant achieves comparable performance under controlled sampling, suggesting that EMA mainly serves as a stabilizing mechanism when batch statistics are unreliable.

With random data shuffling and sufficient training iterations, it provides a stable approximation of the feature expectation. This manifold therefore acts as a semantic reference in the representation space. To formally define our objective, we first ℓ_2 -normalize the feature and the proxy:

$$\tilde{\mathbf{f}}(x) = \frac{\mathbf{h}(x)}{\|\mathbf{h}(x)\|} \in \mathbb{S}^{d-1}, \quad \tilde{\boldsymbol{\mu}} = \frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|} \in \mathbb{S}^{d-1} \quad (3)$$

where d refers to the dimension of the feature (In Stable Diffusion, it is 1280).

2.3 The view of von Mises-Fisher (vMF) distribution

Connection to the vMF distribution. We can interpret $\tilde{\mathbf{f}}(x)$ as directional data modeled by a vMF distribution on $\mathbb{S}^{d-1}$:

$$p(\tilde{\mathbf{f}} \mid \tilde{\boldsymbol{\mu}}, \kappa) = C_d(\kappa) \exp\left(\kappa \tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}\right), \quad (4)$$

Here, κ describes the concentration around the proxy $\tilde{\boldsymbol{\mu}}$. Therefore, maximizing the vMF likelihood is fundamentally equivalent to maximizing the cosine similarity $\tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}$.

Accordingly, our manifold loss can be viewed as a likelihood-based objective: positive samples $\mathcal{D}_{t,r}$ are encouraged to align with $\tilde{\boldsymbol{\mu}}$ (high likelihood under vMF with $\tilde{\boldsymbol{\mu}}$), while negative samples $\mathcal{D}_{t,u}$ are pushed away from $\tilde{\boldsymbol{\mu}}$ (equivalently, higher likelihood under a vMF with the $-\tilde{\boldsymbol{\mu}}$), thereby preserving retained semantics and disrupting the targeted object semantics under unlearned styles.

In practice, to formulate our empirical loss, we directly optimize the geometric direction, omitting the explicit dynamic scalar κ from the objective function. This leads to the following DMU loss:

$$\mathcal{L}_u(x) = \begin{cases} -\tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}(x), & x \in \mathcal{D}_{t,r} \\ \tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}(x), & x \in \mathcal{D}_{t,u} \end{cases} \quad (5)$$

Overall Objective While the DMU loss $\mathcal{L}_u$ is responsible for decoupling the domain-specific features of the target object, we must also ensure that the model does not suffer from catastrophic forgetting of other general concepts. To achieve this, we maintain the standard diffusion denoising loss, denoted as $\mathcal{L}_r$. The original denoising loss is applied exclusively to the positive set $\mathcal{D}_{t,r}$ and the anchor set $\mathcal{D}_r$ to preserve core generative semantics. The final overall optimization objective is a combination of the knowledge preservation loss and the domain unlearning manifold loss:

$$\mathcal{L} = \mathbb{E}_{x \in \mathcal{D}_{t,r} \cup \mathcal{D}_r} [\mathcal{L}_r(x)] + \lambda \mathbb{E}_{x \in \mathcal{D}_{t,r} \cup \mathcal{D}_{t,u}} [\mathcal{L}_u(x)] \quad (6)$$

where λ is a hyperparameter (Empirically, $0.05 \sim 1.00$ works for unlearning, See in Table 7).

This formulation ensures that the domain unlearning operations remain highly targeted without interfering with the shared semantic knowledge space.

Dynamic Evolution of vMF Concentration. This dynamic evolution is mathematically grounded in the properties of the vMF distribution. Specifically, the expected cosine similarity is given by:

$$\mathbb{E}[\tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}] = A_d(\kappa) = \frac{I_{d/2}(\kappa)}{I_{d/2-1}(\kappa)} \quad (7)$$

where $I_v(\cdot)$ denotes the modified Bessel function of the first kind at order v . Crucially, $A_d(\kappa)$ is a strictly monotonically increasing function of κ . Therefore, explicitly optimizing $\mathcal{L}_u$ strictly corresponds to manipulating the semantic concentration κ of the feature space:

- **Positive alignment** ($\kappa_r \uparrow$): Minimizing the loss for positive samples increases their cosine similarity, thereby implicitly increasing their concentration κ_r around $\tilde{\boldsymbol{\mu}}$. It ensures the core semantics of the target object remain robust.
- **Negative repulsion** ($\kappa_u \downarrow$): Pushing negative samples away from $\tilde{\boldsymbol{\mu}}$ continuously decreases their expected similarity, dropping their concentration κ_u . As $\kappa_u \rightarrow 0$, the feature distribution degenerates toward a uniform distribution on the hypersphere. This geometrically explains the process of “losing semantics”, where the target object features in unlearned styles are completely dispersed and structurally destroyed.

Table 1: The larger the κ , the more it aligns with the object’s semantics; the smaller the value, the further it diverges.

Steps	κ_{pos}	κ_{neg}
0	1815.85	1773.03
500	1843.38	637.38
1000	4576.56	0

Table 2: Results over different momentum.

Momentum	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
HA ($\uparrow$)	88.54	88.25	87.93	85.67	88.05	88.06	84.37	83.18	90.45

Simultaneously, the original denoising loss $\mathcal{L}_r$ acts as a crucial spatial regularizer. While the proxy inherently encourages positive samples to converge towards a single center (increasing κ_r), $\mathcal{L}_r$ counterbalances this by preserving the rich intra-class variance. It ensures that positive samples maintain their domain-specific stylistic differences rather than experiencing dimensional collapse, thereby preserving the original generative diversity.

Empirical estimation of concentration. To quantitatively analyze how concentrated the retained and unlearned features are around the shared proxy in the manifold, we further estimate the vMF concentration parameter from the extracted feature embeddings. Given normalized features $\tilde{\mathbf{f}}_j \in \mathbb{S}^{d-1}$ and a normalized shared proxy $\tilde{\boldsymbol{\mu}}$, we first compute the empirical mean cosine similarity

$$\bar{R} = \frac{1}{N} \sum_{j=1}^N \tilde{\boldsymbol{\mu}}^\top \tilde{\mathbf{f}}_j, \quad (8)$$

which serves as an empirical estimate of the mean resultant length under the vMF model.

Since $A_d(\kappa)$ is strictly increasing but does not admit a simple closed-form inverse (See in Equation 7), we adopt the standard high-dimensional approximation to estimate κ from $\bar{R}$:

$$\hat{\kappa} = \begin{cases} 0, & \bar{R} < 0, \\ \frac{\bar{R}(d - \bar{R}^2)}{1 - \bar{R}^2}, & \bar{R} \geq 0. \end{cases} \quad (9)$$

In our setting, the feature dimension is high ($d = 1280$), making this approximation accurate and numerically stable. Intuitively, a larger $\bar{R}$ corresponds to a larger $\hat{\kappa}$, indicating that features are more tightly concentrated around the proxy; conversely, a smaller $\bar{R}$ implies a weaker concentration and a more dispersed feature distribution.

3 Experiments

3.1 Experiment Setups

We implemented our method under two unlearning scenarios: 1) Sequential Unlearning (SU), where requests are processed chronologically, requiring the model to unlearn the first domain before proceeding to the next in a continuous stream; and 2) Batch Unlearning (BU), where multiple unlearning requests are consolidated and executed simultaneously in a single optimization step.

Goal: Our objective is to cause the generation of the target object to fail within specific domains, while maintaining the performance of other objects in all domains.

Metrics: Following UnlearnCanvas [10], we use the followings as metrics: **UA** is 100% minus the classification accuracy of target object in $\mathcal{D}_{t,u}$. **RA_{pos}** is classification accuracy of target object in $\mathcal{D}_{t,r}$. **RA_{anc}** is classification accuracy of other objects in $\mathcal{D}_r$. **HA** [9] is the harmonic average of **UA**, **RA_{pos}**, and **RA_{anc}**, measuring the balance between unlearning effectiveness and knowledge retention: $\mathbf{HA} = 3 \times (\frac{1}{\mathbf{UA}+1e-8} + \frac{1}{\mathbf{RA}_{pos}+1e-8} + \frac{1}{\mathbf{RA}_{anc}+1e-8})^{-1}$. **StyleAcc_R** is the classification accuracy of styles in $\mathcal{D} \setminus \mathcal{D}_{t,u}$. **CLIP_R** refers to the clip score of (image, prompt) pairs in $\mathcal{D} \setminus \mathcal{D}_{t,u}$.

Feature Extraction Details: To construct the proxy, we extract the intermediate feature $\mathbf{h}_t(x)$ from the U-Net’s bottleneck. Specifically, we hook the output of the first block within the Middle Block (middle_block[1]), as it captures the most compressed semantic and domain representations. Since styles are generally independent of spatial layouts, we apply Global Average Pooling over the spatial dimensions to obtain a 1D feature vector, followed by l_2 -normalization.

Datasets & Models: The experiments are primarily implemented in UnlearnCanvas [10] benchmark where it has 20 objects and 51 styles, each object-style pair contains 20 samples. Based on this benchmark, we extended our experiments to OfficeHome [16] (65 objects, 4 styles, and 15,500 samples) and PACS [17] (7 objects, 4 styles, and 9991 samples). The experiments are conducted on the SD1.5, SD2, and SDXL.

Table 3: Main Experiments. The default request is to unlearn 6 domains.

Method	HA ($\uparrow$)	UA ($\uparrow$)	RA _{pos} ($\uparrow$)	RA _{anc} ($\uparrow$)	StyleAcc _R ($\uparrow$)	CLIP _R ($\uparrow$)
FT w/o $\mathcal{D}_{t,u}$ (BU)	47.98	30.24	53.33	93.40	91.27	29.10
GA (BU)	3.31	97.78	1.13	97.73	94.16	27.56
L1 (BU)	69.74	48.89	82.49	95.77	94.29	28.12
L2 (BU)	60.38	36.44	87.22	92.76	95.17	28.83
UCE (BU) [18]	59.39	82.33	57.03	48.01	94.69	27.25
SculpMemory (BU) [4]	53.63	52.09	39.69	86.62	96.23	27.79
ADU (BU) [9]	53.63	52.09	39.69	86.62	89.31	27.77
SSDA (BU) [19]	64.08	41.47	86.93	89.27	97.12	29.06
DMU (BU)	87.33	87.07	89.36	85.63	95.20	29.08
ESD (SU) [20]	22.11	100	11.91	23.97	93.96	22.81
EDiff (SU) [21]	22.99	99.17	14.76	19.00	24.85	22.68
UCE (SU) [18]	27.95	100	21.77	19.46	76.53	22.94
CUIG (SU) [11]	70.49	56.25	72.11	91.64	97.09	29.67
DMU (SU)	85.81	91.13	84.24	82.52	88.90	29.55

Table 4: Other Datasets.

Dataset	Method	HA ($\uparrow$)	UA ($\uparrow$)	RA _{pos} ($\uparrow$)	RA _{anc} ($\uparrow$)	StyleAcc _R ($\uparrow$)	CLIP _R ($\uparrow$)
OfficeHome	ADU	56.61	44.12	62.17	70.22	84.67	25.49
	SSDA	50.77	37.50	50.00	80.47	80.19	25.17
	DMU	81.35	82.35	76.47	85.78	93.26	26.41
PACS	ADU	58.32	59.80	41.18	95.83	94.43	24.39
	SSDA	64.93	39.22	94.12	99.17	96.73	26.57
	DMU	91.63	98.04	88.24	89.23	91.49	25.79

3.2 Results

3.2.1 Main Experiments

Table 3 presents the main experimental results under both Sequential Unlearning (SU) and Batch Unlearning (BU) settings on the UnlearnCanvas benchmark, where six domains are removed by default (see details in Appendix). **Overall**, DMU consistently achieves the **SOTA** HA across both settings. Compared to existing methods, it not only significantly improves UA, but also preserves higher retained accuracy on both positive samples (RA_{pos}) and anchor samples (RA_{anc}), while maintaining competitive style consistency and semantic alignment (StyleAcc_R and CLIP_R).

We use the HA to characterize the trade-off effects of unlearning; a low HA corresponds to: 1) unlearning failure, or 2) successful unlearning that lacks robustness. In the SU setting, prior methods like ESD, EDiff, and UCE achieve nearly perfect UA scores, but suffer from severe degradation in retained performance, with RA_{pos} and RA_{anc} dropping significantly. In contrast, CUIG improves retention performance but still yields relatively limited unlearning effectiveness. DMU effectively mitigates this trade-off, achieving a high HA and UA, while maintaining strong retained performance. These results indicate that DMU can remove target knowledge without excessively damaging the retained distribution. In the BU setting, a similar trend is observed. Existing methods either prioritize retention (e.g., SSDA achieves high RA_{pos} but low UA) or emphasize unlearning at the cost of performance degradation. DMU again achieves the best overall performance, with the highest HA and UA, while maintaining competitive retained accuracy and generation quality. Notably, DMU outperforms all baselines in jointly optimizing unlearning effectiveness and retention, demonstrating its robustness under BU setting.

Generalization across models and datasets. Beyond the UnlearnCanvas benchmark, we further evaluate DMU across different model architectures and data distributions. Specifically, we extend our experiments to larger diffusion backbones (SDXL and SD2) on UnlearnCanvas (See in Table 5), and to domain generalization benchmarks (OfficeHome and PACS) based on SD1.5 (See in Table 4). Across all these settings, DMU consistently maintains strong unlearning effectiveness and retention

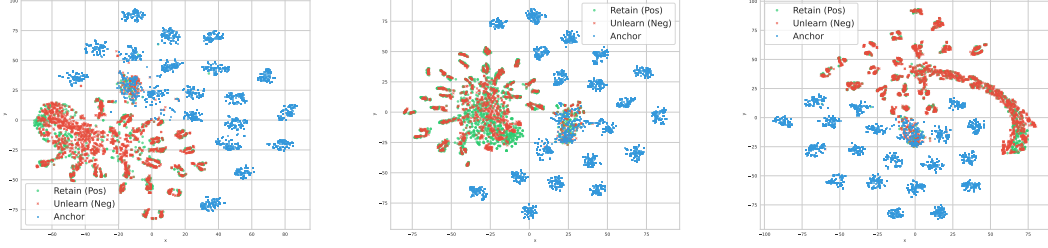


Figure 3: Manifold Visualization of SSDA, SculptingMemory, and ADU, in 1000 steps.

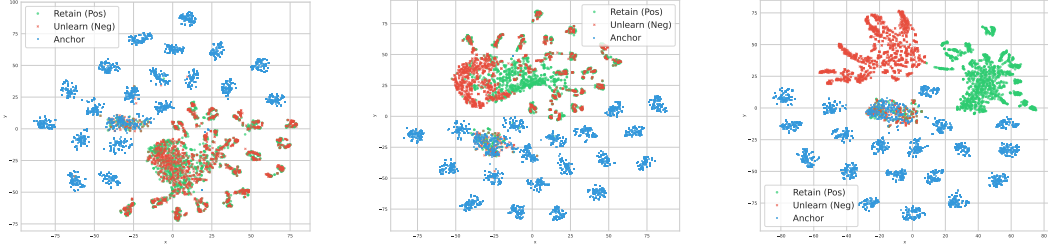


Figure 4: Manifold Visualization of DMU in 0, 500, and 1000 steps.

performance, demonstrating that our method is not tied to a specific model scale or dataset, but generalizes well across architectures and domains.

3.2.2 Visualization

Using the unlearned weights, we extracted features for 100 batches from the bottleneck layers of U-Net. To highlight the differences between the positive set and the negative set, we then randomly removed a portion of the anchor samples so that the anchor samples constituted 50% of the total samples, while the positive and negative samples each accounted for 25%. Figure 3 shows that three compared methods did not completely decouple the positive set and the negative set. This also explains why the results for **UA** and **RA_{pos}** are not robust. Figure 4 shows the decoupling process. At the beginning of the unlearning process (step 0), the positive and negative samples are tightly clustered and overlapping in the latent space, indicating that the model retains a strong influence from the negative data. As the process reaches step 500, a noticeable divergence begins to emerge, with the two sets showing initial signs of separation. By the completion of unlearning at step 1000, the positive and negative samples are entirely decoupled and form distinct clusters. This progression demonstrates that the model has successfully eliminated the influence of the negative samples, achieving effective domain unlearning. Table 1 shows the increasing κ_{pos} and decreasing κ_{neg} , which indicates progressive semantic concentration of positive samples and effective dispersion of negative samples.

3.3 Ablation Studies

Ablation of Loss As shown in Table 6, the full DMU achieves the best overall balance with the highest HA. Removing *positive alignment* still maintains relatively high UA, but causes **RA_{pos}** to drop sharply, showing severe degradation on the preserved positive set. In contrast, removing *negative*

Table 5: Performance of SDXL and SD2 in UnlearnCanvas Dataset.

Method	HA ($\uparrow$)	UA ($\uparrow$)	RA _{pos} ($\uparrow$)	RA _{anc} ($\uparrow$)	StyleAcc _R ($\uparrow$)	CLIP _R ($\uparrow$)
SDXL	6.79	2.38	92.7	89.68	97.84	28.55
+ DMU	95.75	100.00	98.10	89.78	97.78	28.59
SD2	7.32	2.57	95.56	97.11	96.25	29.16
+ DMU	96.60	95.71	96.19	97.94	94.63	29.67

Table 6: Performance of Loss Ablation Studies

Method	HA ($\uparrow$)	UA ($\uparrow$)	RA _{pos} ($\uparrow$)	RA _{anc} ($\uparrow$)	StyleAcc _R ($\uparrow$)	CLIP _R ($\uparrow$)
Full DMU (*)	87.33	87.07	89.36	85.63	95.20	29.08
* Remove Pos. alignment	58.77	72.92	38.15	89.89	93.61	28.76
* Remove Neg. repulsion	57.81	35.42	77.46	92.98	96.73	29.19
* Remove Denoising preservation (Pos. & Anchor set)	84.39	89.58	79.48	84.73	83.80	28.79
* Remove Pos. alignment & Denoising preservation (Pos. set)	48.07	79.17	26.01	88.24	81.30	28.70

repulsion leads to a collapse of UA, indicating failed unlearning despite high retention performance. Removing *denoising preservation* preserves strong UA, but slightly degrades both RA_{pos}, RA_{anc}, and StyleAcc_R, suggesting its role in stabilizing retained semantics. Finally, jointly removing *positive alignment* and positive-set *denoising preservation* causes RA_{pos} to further collapse, demonstrating that both objectives are essential for preventing catastrophic forgetting on the preserved target object.

Momentum In this part, we show the impact of the momentum parameter. The momentum of a manifold refers to its tendency to maintain a value in each update of each batch. A momentum value of 0 means the proxy only relies on the current batch features, while a value of 1 results in a static proxy (e.g., remaining at initialization), both of which fail to capture the desired dynamic update behavior of the proxy and are therefore excluded from our analysis. HA exhibits oscillatory behavior when momentum is between 0.1 and 0.9. The overall performance remains robust, with no significant decline observed (See in Table 2).

More Unlearn Pairs In this part, we test the robustness of DMU by increasing the number of unlearning requests. Under the same training timestep (timestep = 1000), we observe that as the number of requests increases, UA gradually decreases, indicating that the unlearning task becomes more challenging with more domains to be removed (See in Figure. 5). As the requests increase, the positive set becomes smaller while the anchor set remains unchanged; thus, aside from UA and HA, the metrics show little change. To further investigate this effect, we increase the training timestep to 2000 for the cases with 25 and 30 requests. We find that UA recovers to approximately 85%, demonstrating that the performance degradation is not due to a limitation of the method itself, but rather insufficient optimization steps. This suggests that as the number of unlearning requests grows, DMU requires more training iterations to effectively decouple unlearned domains’ information.

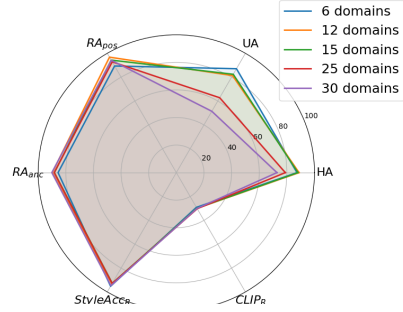


Figure 5: Exp. of Unlearning requests.

4 Discussion and Limitation

4.1 Extension to multi-domain unlearning.

In our current experiments, we perform unlearning on a single target object with multiple styles, and report averaged results across different objects. Each experiment maintains an object-specific proxy, which is updated independently using only the corresponding positive samples and does not involve gradient-based optimization. This design naturally enables a more general multi-domain setting, where multiple objects and multiple styles can be jointly unlearned within a unified framework. Specifically, one can construct multiple positive and negative sets corresponding to different object–style combinations, allowing for flexible configurations such as jointly unlearning N objects across M styles, or even heterogeneous settings where each object is associated with a different subset of styles. Due to space limitations, we leave a comprehensive exploration of such multi-domain unlearning scenarios for future work. We believe this direction is particularly promising for scaling unlearning methods to more realistic and complex data distributions.

4.2 Auditing and Concept Identifiability

Our evaluation measures forgetting and retention under the tested object–style requests; it does not establish resistance to latent-space recovery or concealment of erased concepts. Latent-space auditing [22] and erasure-footprint identification [23] provide complementary tests that remain to be investigated for DMU.

4.3 Feature Extraction in Different Layers

Due to page constraints and the substantial time cost of our experiments, we do not explore the impact of using different layers for feature extraction in the DMU process. In practice, many

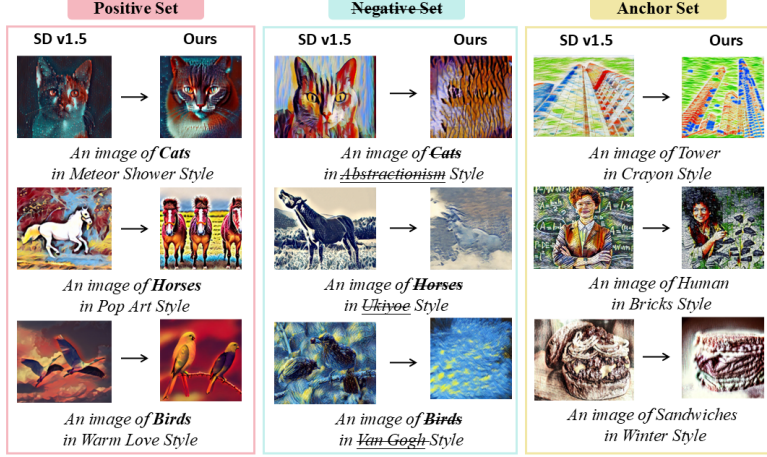


Figure 6: Case figure shows that our method selectively erases the domains (coupling of object-style pairs) while preserving retained domains and maintaining performance on other objects.

diffusion-based works commonly extract features from intermediate (middle) layers [24–26], as these representations tend to balance low-level details and high-level semantics. Early layers primarily capture local textures and noise patterns, while deeper layers encode more abstract, task-specific concepts; intermediate layers therefore provide a more stable and informative representation for downstream objectives such as unlearning.

5 Related Works

5.1 Multiple Concept Erasing

Early concept erasure methods in diffusion models mainly focus on removing a single concept through static fine-tuning or prompt-level editing [18, 20, 27–29]. However, real-world unlearning requests often involve multiple concepts that may be semantically correlated or partially overlapping, making direct extension of single-concept methods prone to interference and utility degradation. ESD [20] fine-tunes diffusion model weights with negative guidance. UCE [18] offers a unified closed-form framework for multi-concept forgetting. MACE [28] employs LoRA modules alongside closed-form cross-attention refinement to erase a large number of concepts while maintaining the model’s generative performance for other concepts. SepME [29] achieves independent, non-interfering removal of multiple concepts by generating concept-irrelevant representations and decoupling weight updates for each concept. Sculpting Memory [4] leverages dynamic masks and concept-aware optimization for stable multi-concept erasing. Forget-It-All (FIA) [13] performs training-free multi-concept unlearning by identifying concept-sensitive neurons and fusing their masks while preserving neurons that support general generation. CUIG [11] proposes a continual concept unlearning framework for diffusion models that mitigates utility degradation by stabilizing parameter updates and reducing cross-concept interference during sequential removal. Extending multi-concept erasure to video generation, FADE [30] combines joint closed-form key/value editing with frame-aware low-rank adapters and prompt-dependent routing to address residual concept leakage in individual frames. These approaches mark progress toward scalable and unified multi-concept erasing, but are still insufficient for large-scale forgetting, especially when erasing (object, style) group.

5.2 Auditing and Identifying Erased Concepts

Beyond erasure effectiveness on standard prompts, recent work examines what unlearned models still reveal or encode. AVCE [22] uses latent-space auditing to expose recoverable erased concepts and incorporates auditing-aware updates to strengthen erasure. Tracer [23] addresses a complementary question: it identifies which concepts were erased and estimates their number through spectral analysis of weight footprints and limited forward probes. These works motivate evaluating compositional unlearning beyond output-based forgetting and retention metrics.

5.3 Domain Unlearning

The essence of domain unlearning lies in eliminating the statistical footprints of a specific domain from the parameter distribution of a pre-trained model [9, 10]. ADU [9] uses Maximum Mean Discrepancy (MMD) to explicitly disentangle domain distributions and adaptively captures instance-

specific domain information. SSDA [19] minimizes the classification loss for retained images and maximizes uncertainty loss for unlearned images, then uses Gaussian-guided latent alignment (GLA) strategy that uses a prior distribution to align domain-agnostic information and retained domain information. However, these methods only consider the explicit decoupling of the unlearned and retained domains, they fail to account for whether the retained domain still persists within the manifold of general knowledge. Consequently, the unlearning performance often lacks robustness.

6 Conclusion

Main Findings: In this paper, we present Domain Manifold Unlearning (DMU), a method designed to address the challenges of robustness and catastrophic forgetting in multiple domains unlearning for diffusion models. By explicitly modeling data manifold using a non-parametric proxy updated via EMA, DMU enables the precise decoupling of domain-specific features from shared semantic space. Through the lens of the von Mises-Fisher distribution, we provide a mathematical foundation for the unlearning process, showing that optimizing the manifold loss corresponds to manipulating semantic concentration and dispersion in the latent space. By combining the DMU loss with a standard denoising loss, our method ensures that "forgotten" domains are structurally destroyed while the model's core utility and intra-class diversity on retained data remain intact. Empirical evaluations across different benchmarks and models demonstrate that DMU is a versatile solution for large-scale and continuous unlearning requests.

Future work: We will explore the extension of this proxy to the limitation we discussed before and other generative modalities, such as Large Language Models, to further enhance the "right to be forgotten" in the era of foundation models.

References

- [1] A. F. Cooper, C. A. Choquette-Choo, M. Bogen, M. Jagielski, K. Filippova, K. Z. Liu, A. Chouldechova, J. Hayes, Y. Huang, N. Mureshghallah *et al.*, "Machine unlearning doesn't do what you think: Lessons for generative ai policy, research, and practice," *arXiv preprint arXiv:2412.06966*, 2024.
- [2] N. Li, C. Zhou, Y. Gao, H. Chen, Z. Zhang, B. Kuang, and A. Fu, "Machine unlearning: Taxonomy, metrics, applications, challenges, and prospects," *IEEE Transactions on Neural Networks and Learning Systems*, 2025.
- [3] P. Regulation, "Regulation (eu) 2016/679 of the european parliament and of the council," *Regulation (eu)*, vol. 679, no. 2016, pp. 10–13, 2016.
- [4] G. Li, Y. Xiao, J. Ji, K. Deng, B. Hui, L. Guo, and X. Ma, "Sculpting memory: Multi-concept forgetting in diffusion models via dynamic mask and concept-aware optimization," *arXiv preprint arXiv:2504.09039*, 2025.
- [5] K. Deng, G. Li, Y. Xiao, B. Hui, and X. Ma, "Forget many, forget right: Scalable and precise concept unlearning in diffusion models," *arXiv preprint arXiv:2601.06162*, 2026.
- [6] Y. Xiao, R. Ye, B. Liu, X. Ma, and B. Hui, "Efficient knowledge graph unlearning with zeroth-order information," in *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, 2025, pp. 3509–3519.
- [7] Y. Xiao, R. Ye, and B. Hui, "Knowledge graph unlearning with schema," in *Proceedings of the 31st International Conference on Computational Linguistics*, 2025, pp. 3541–3546.
- [8] Y. Xiao, G. Li, J. Ji, R. Ye, X. Ma, and B. Hui, "The right to be forgotten in pruning: Unveil machine unlearning on sparse models," in *NeurIPS 2025 Workshop on Regulatable ML*.
- [9] K. Kawamura, Y. Goto, R. Yanagi, H. Kataoka, and G. Irie, "Approximate domain unlearning for vision-language models," *arXiv preprint arXiv:2510.08132*, 2025.
- [10] Y. Zhang, C. Fan, Y. Zhang, Y. Yao, J. Jia, J. Liu, G. Zhang, G. Liu, R. R. Kompella, X. Liu *et al.*, "Unlearncanvas: Stylized image dataset for enhanced machine unlearning evaluation in diffusion models," *arXiv preprint arXiv:2402.11846*, 2024.
- [11] J. Lee, Z. Mai, J. Yoo, C. Fan, C. Zhang, and W.-L. Chao, "Continual unlearning for text-to-image diffusion models: A regularization perspective," *arXiv preprint arXiv:2511.07970*, 2025.
- [12] K. Thakral, T. Glaser, T. Hassner, M. Vatsa, and R. Singh, "Continual unlearning for foundational text-to-image models without generalization erosion," *arXiv preprint arXiv:2503.13769*, 2025.

- [13] K. Deng, B. Hui, G. Li, J. Ji, M. Qin, G. Yuan, and X. Ma, “Forget-it-all: Multi-concept machine unlearning via concept-aware neuron masking,” *arXiv preprint arXiv:2601.06163*, 2026.
- [14] Y. Liu, L. Xu, X. Yuan, C. Wang, and B. Li, “The right to be forgotten in federated learning: An efficient realization with rapid retraining,” in *IEEE INFOCOM 2022-IEEE conference on computer communications*. IEEE, 2022, pp. 1749–1758.
- [15] A. K. Tarun, V. S. Chundawat, M. Mandal, and M. Kankanhalli, “Fast yet effective machine unlearning,” *IEEE transactions on neural networks and learning systems*, vol. 35, no. 9, pp. 13 046–13 055, 2023.
- [16] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, “Deep hashing network for unsupervised domain adaptation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5018–5027.
- [17] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, “Deeper, broader and artier domain generalization,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5542–5550.
- [18] R. Gandikota, H. Orgad, Y. Belinkov, J. Materzyńska, and D. Bau, “Unified concept editing in diffusion models,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2024, pp. 5111–5120.
- [19] H. Basak and Z. Yin, “Forget more to learn more: Domain-specific feature unlearning for semi-supervised and unsupervised domain adaptation,” in *European Conference on Computer Vision*. Springer, 2024, pp. 130–148.
- [20] R. Gandikota, J. Materzynska, J. Fiotto-Kaufman, and D. Bau, “Erasing concepts from diffusion models,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 2426–2436.
- [21] J. Wu, T. Le, M. Hayat, and M. Harandi, “Erasing undesirable influence in diffusion models,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 28 263–28 273.
- [22] K. Deng, Y. Li, G. Li, Y. Xiao, G. Yuan, X. Yuan, B. Hui, and X. Ma, “No concept escapes the audit: Auditing-aware unlearning for verifiable concept erasure in diffusion models,” *arXiv preprint arXiv:2610.05401*, 2026.
- [23] K. Deng, Y. Li, Y. Xiao, B. Hui, G. Yuan, and X. Ma, “Your unlearning gives you away: Identifying erased concepts in diffusion models,” *arXiv preprint arXiv:2610.05601*, 2026.
- [24] T. Lan, Y. Zheng, and J. Yin, “Diffusion-based cross-modal feature extraction for multi-label classification,” *arXiv preprint arXiv:2509.15553*, 2025.
- [25] S. Chang, P. Wang, J. Tang, F. Wang, and Y. Yang, “Sparsedit: Token sparsification for efficient diffusion transformer,” *arXiv preprint arXiv:2412.06028*, 2024.
- [26] H. Lee, W. Jang, J. Yang, T. Kim, S. Kim, S. Jung, and S. Kim, “V-warper: Appearance-consistent video diffusion personalization via value warping,” *arXiv preprint arXiv:2512.12375*, 2025.
- [27] C. Fan, J. Liu, Y. Zhang, E. Wong, D. Wei, and S. Liu, “Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation,” *arXiv preprint arXiv:2310.12508*, 2023.
- [28] S. Lu, Z. Wang, L. Li, Y. Liu, and A. W.-K. Kong, “Mace: Mass concept erasure in diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6430–6440.
- [29] M. Zhao, L. Zhang, T. Zheng, Y. Kong, and B. Yin, “Separable multi-concept erasure from diffusion models,” *arXiv preprint arXiv:2402.05947*, 2024.
- [30] Y. Li, K. Deng, C. Feng, Z. Tang, and L. Yuan, “Fade: Frame-aware diffusion-transformer-based multi-concept erasure for video unlearning,” *arXiv preprint arXiv:2610.03980*, 2026.
- [31] L. Tunstall, E. Beeching, N. Lambert, N. Rajani, K. Rasul, Y. Belkada, S. Huang, L. von Werra, C. Fourrier, N. Habib, N. Sarrazin, O. Sanseviero, A. M. Rush, and T. Wolf, “Zephyr: Direct distillation of lm alignment,” 2023.
- [32] S. Pal, C. Wang, J. Diffenderfer, B. Kailkhura, and S. Liu, “Llm unlearning reveals a stronger-than-expected coreset effect in current benchmarks,” *arXiv preprint arXiv:2504.10185*, 2025.
- [33] N. Li, A. Pan, A. Gopal, S. Yue, D. Berrios, A. Gatti, J. D. Li, A.-K. Dombrowski, S. Goel, L. Phan *et al.*, “The wmdp benchmark: Measuring and reducing malicious use with unlearning,” *arXiv preprint arXiv:2403.03218*, 2024.

A Detailed Experiment Setups

Provided from UnlearnCanvas benchmarks [10], we use the pretrained object and style classifier to get **UA**, **RA_{pos}**, **RA_{anc}**, and **StyleAcc_R** on UnlearnCanvas dataset. Following their plan, we adopted a ViT-L/16 model pretrained on ImageNet and finetune it on OfficeHome [16] and PACS [17]. Datasets are split into the train set and test set with a ratio of 9: 1. After hyper-parameter tuning, the classifiers are trained with the Adam optimizer at a learning rate of 0.01 for 10 epochs. We use openai/clip-vit-base-patch32 to compute the **CLIP_R** between generated images and their corresponding prompts.

A.1 GPU, Training and Test Latency

We run all experiments in 4 L40S GPUs.

BU: Each experiment requires approximately 210 minutes. The entire image generation process typically takes approximately 1 hour to complete.

SU: Each experiment requires approximately Number of styles $\times$ 210 minutes. The entire image generation process typically takes approximately 1 hour to complete.

A.2 How many objects and how many styles?

UnlearnCanvas Here we show the specific objects and styles we use in this dataset.

6 Objects: Birds, Trees, Cats, Horses, Jellyfish, Human.

6 Styles: Abstractionism, Byzantine, Cartoon, Cold Warm, Ukiyoe, Van Gogh.

12 Styles: Abstractionism Blossom Season, Byzantine, Cartoon, Cold Warm, Expressionism, Ukiyoe, Van Gogh, Meta Physics, Sponge Dabbed, Warm Smear, Winter.

15 Styles: Abstractionism, Artist Sketch, Blossom Season, Byzantine, Cartoon, Cold Warm, Dapple, Expressionism, Greenfield, Ukiyoe, Van Gogh, Meta Physics, Sponge Dabbed, Warm Smear, Winter.

25 Styles: Abstractionism, Artist Sketch, Blossom Season, Bricks, Byzantine, Cartoon, Cold Warm, Color Fantasy, Comic Etch, Crayon, Cubism, Dadaism, Dapple, Defoliation, Early Autumn, Expressionism, Fauvism, French, Greenfield, Ukiyoe, Van Gogh, Meta Physics, Sponge Dabbed, Warm Smear, Winter.

30 Styles: Abstractionism, Artist Sketch, Blossom Season, Bricks, Byzantine, Cartoon, Cold Warm, Color Fantasy, Comic Etch, Crayon, Cubism, Dadaism, Dapple, Defoliation, Early Autumn, Expressionism, Fauvism, French, Greenfield, Glowing Sunset, Gorgeous Love, Impressionism, Ink Art, Joy, Ukiyoe, Van Gogh, Meta Physics, Sponge Dabbed, Warm Smear, Winter.

OfficeHome Here we show the specific objects and styles we use in this dataset.

6 Objects: Bike, Calculator, Monitor, Refrigerator, Telephone, TV.

2 Styles: Art, Clipart.

PACS Here we show the specific objects and styles we use in this dataset.

7 Objects: All Objects in PACS.

3 Styles: art painting, cartoon, sketch.

A.3 Image Generation

Random Seeds. For the positive and negative sets, we use seven random seeds: 42, 144, 79, 27, 599, 123, and 577. For the anchor set, we use five random seeds: 188, 288, 388, 488, and 588.

Generation Protocol. For each domain (i.e., each object–style pair) and each seed, we generate 4 images using the relevant seed + (0-3), because we have 4 GPUs. The final test dataset is obtained by aggregating samples across all seeds.

Table 7: Ablation Studies on Different Lambdas

λ	HA ($\uparrow$)	UA ($\uparrow$)	RA _{pos} ($\uparrow$)	RA _{anc} ($\uparrow$)	StyleAcc _R ($\uparrow$)	CLIP _R ($\uparrow$)
0.00	47.42	25.79	72.42	93.59	92.27	29.15
0.05	86.26	89.58	80.64	89.16	95.51	29.32
0.15	90.45	94.59	84.29	93.17	94.99	29.16
0.20	82.60	79.17	83.82	85.04	94.14	29.33
0.25	90.58	95.83	89.93	86.48	93.08	29.32
0.30	87.15	89.58	87.28	84.73	93.92	29.33
0.50	82.46	77.08	86.71	84.21	93.31	29.28
0.75	88.08	97.92	84.68	83.08	91.03	29.26
1.00	86.90	95.83	82.37	83.70	90.57	29.09

Table 8: Application in LLM Unlearning

Model	WMDP-Bio ($\downarrow$)	WMDP-Cyber ($\downarrow$)	MMLU ($\uparrow$)
ZEPHYR-7B	63.7	44.0	58.1
+ NPO	53.7	34.6	50.4
+ GA	51.8	30.4	45.9
+ RMU	27.3	26.7	56.4
+ RMU-LAT	31.7	29.6	57.5
+ DMU	29.3	27.3	56.6

Weight of DMU (Lambda). We conduct an ablation study on the loss weighting parameter λ using the UnlearnCanvas dataset, focusing on the Birds class with the Stable Diffusion v1.5 backbone. When $\lambda = 0.0$, the DMU objective is removed, reducing the training to standard fine-tuning on the retain set (i.e., positive and anchor samples only). As shown in Table 7, this setting leads to significantly degraded unlearning performance, particularly in terms of UA and HA, indicating that the DMU mechanism is essential for effective forgetting. In contrast, across a wide range of λ values from 0.05 to 1.00, the model consistently achieves strong unlearning results with high UA and HA scores, while maintaining stable retention performance (RA_{pos} and RA_{anc}) and generation quality (StyleAcc_R and CLIP_R). These results demonstrate that the proposed method is robust to the choice of λ , and that the unlearning effectiveness is not sensitive to precise tuning of this hyperparameter. For the experiments presented in the main text, we initially selected 0.15 as the weight via random sampling; given the significant results shown, a default weight of 0.15 was subsequently applied to all DMU experiments within the main text. However, this part indicated that varying the weight within the range of 0.05 to 1.00 yielded no substantial differences.

B Case Study

Here we show some cases in Figure 7:

C Application in LLM Unlearning.

We further extend DMU to the large language model setting based on Zephyr-7B [31]. For the retain set, we first construct a set of reference points by randomly partitioning the tokens in the first batch into 500 groups and computing the mean hidden state at layer l for each group. These reference points are initialized via a short warmup stage of w optimization steps. During unlearning, for each forget token, we identify its k nearest reference points in the layer- l representation space, compute a softmax over their distances, and use the resulting weighted centroid as the target representation. The forget loss is then defined similarly to RMU [32], except that the target is no longer a random vector but this adaptive centroid. Meanwhile, the reference points are continuously updated using an exponential moving average (EMA) based on retain tokens to ensure stability. As shown in Table 8, DMU achieves strong unlearning performance on both WMDP-Bio [33] and WMDP-Cyber [33]

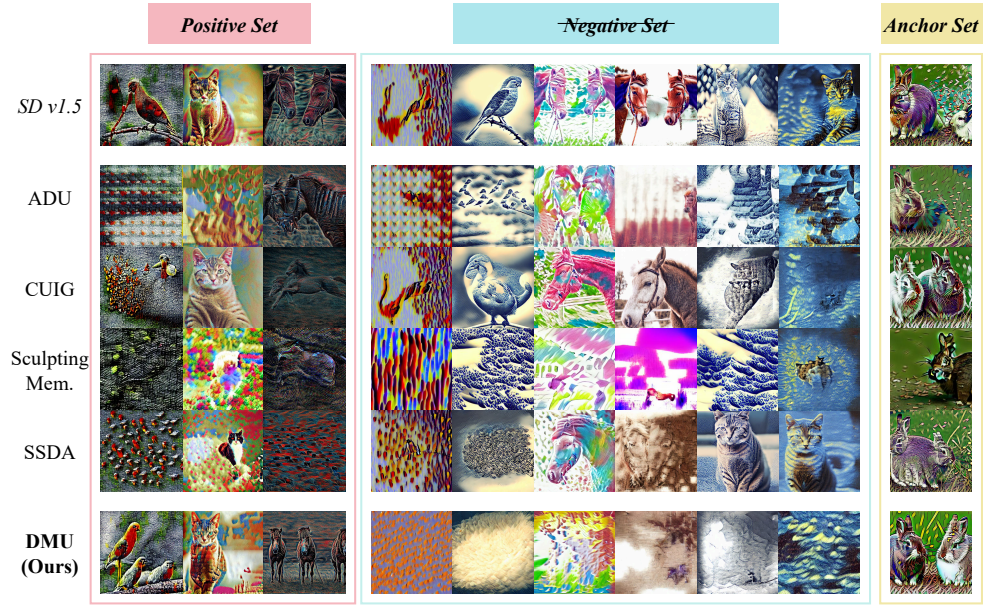


Figure 7: Case

while maintaining competitive general capability on MMLU. Compared with prior methods such as NPO, GA, and RMU variants, our approach provides a better trade-off between effective forgetting and knowledge retention, demonstrating its applicability beyond diffusion models.