YOUR T2I MODEL'S EMBEDDINGS KNOW ABOUT MEMORIZATION AND ENABLE BROAD UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image (T2I) diffusion models are known to memorize training examples, reproducing near-verbatim copies at inference time and raising privacy and copyright concerns. This troubling phenomenon has given rise to a plethora of methods to mitigate memorization, either via prevention measures, specialized inference procedures, or post-hoc unlearning methods. In this work, we make a surprising fundamental observation about memorized examples in text-to-image diffusion models: a simple logistic regression classifier on top of embeddings can predict with very high accuracy which examples are memorized, suggesting the existence of shared representational structure among memorized examples, which is distinct from non-memorized examples. We also show evidence that the relationship between memorization and the structures we identify in representation space is causal. The existence of this representational structure, aside from being scientifically interesting in its own right, is practically important as it opens the door to a new problem setting we refer to as "broad unlearning" of all memorized examples from knowledge of only a subset of them; a realistic setting that prior work has overlooked. Motivated by our findings, we propose a new method called BLUR that is significantly more effective than prior work at reducing the memorization of both known and unknown (to the unlearning algorithm) memorized examples, while preserving the model's ability to generate high quality and diverse images.

1 INTRODUCTION

Text-to-image (T2I) diffusion models have achieved remarkable progress in generating high-quality, semantically rich images from natural language prompts (Rombach et al., 2022; Ramesh et al., 2022; Saharia et al., 2022). However, alongside these capabilities, a growing body of work has demonstrated that T2I diffusion models can reproduce near-verbatim copies of their training images given semantically related prompts (Carlini et al., 2021; 2023; Somepalli et al., 2023a) – a phenomenon known as *memorization*. This poses serious privacy and copyright risks, with direct implications under data protection legislation such as the GDPR's right to erasure (European Parliament & Council of the European Union, 2016) and the EU AI Act's requirements on training data transparency (European Parliament & Council of the European Union, 2024). As model scale and capability continue to grow, so too does the risk of memorization and its associated harms.

In response, a plethora of methods have been proposed for memorization mitigation in T2I models, including prevention measures that act during pretraining (Somepalli et al., 2023b; Wen et al., 2024; Ren et al., 2024), specialized inference procedures (Somepalli et al., 2023b; Wen et al., 2024; Ren et al., 2024; Han et al., 2025; Jeon et al., 2025; Zhao et al., 2026) or post-hoc "unlearning" methods (Sekhari et al., 2021; Kurmanji et al., 2023; Triantafillou et al., 2024) that attempt to "erase" the influence of the memorized data from the model weights (Alberti et al., 2025). These categories face distinct challenges. On the one hand, modifying only the inference procedure but not the weights leaves the model vulnerable to white-box attacks. On the other hand, altering the weights risks degrading the model's general utility and generation quality; a trade-off that is observed empirically in existing approaches. We argue that finding the most effective solution to this problem requires deepening our understanding of how memorization manifests in these models. How does the degree of memorization of an example affect the way that the model represents it internally?

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

A recent interesting line of work has discovered a diverse set of properties that are encoded in surprisingly simple ways in the model's internal representations. Azaria & Mitchell (2023) found that the concept of "truthfulness" is decodable from hidden representations, and Arditi et al. (2024) found that "refusal" is mediated by a single direction. Krasheninnikov et al. (2026) discovered that training-order recency is linearly encoded in model activations too. Our first contribution is a key discovery that builds on this line of work to show that, in text-to-image diffusion models, *memorization is linearly decodable from the model's embeddings*. This foundational discovery is interesting in its own right, but also practically important, as it leads us to rethink how we should approach the problem of memorization mitigation in a principled way. It also opens the door to a new problem setting that we refer to as "broad unlearning of memorization": can we leverage the shared representational structure of memorized examples in order to eliminate all memorization from the model, even if we only have knowledge of a subset of the memorized examples?

In this paper, we answer that question to the affirmative. Equipped with our knowledge of how memorization affects embeddings of examples, we design a novel method called Broad Local Unlearning of Representations (BLUR) that finetunes the pretrained model to explicitly align the embeddings of the known subset of memorized examples towards non-memorized examples, in order to reduce memorization. Specifically, our representation-alignment objective pulls the embedding of each identified memorized example towards the centroid of its k nearest (in embedding space) non-memorized neighbours. Because there is shared structure in embedding space between memorized examples, we show that it suffices to apply this alignment objective on a subset of memorized examples in order to eliminate memorization more broadly. Empirically, our method achieves stronger and more consistent memorization reduction on both the examples used for unlearning and "held out" (from unlearning) memorized examples, while maintaining competitive generation quality across different memorization settings.

We summarize our key contributions as follows:

1. We make the foundational discovery that memorization in a text-to-image diffusion model is linearly-decodable from the model's embeddings. We show that a linear classifier can tell apart memorized from non-memorized examples with almost perfect accuracy. We also present a causal intervention that shows that when a random set of examples is made to be more memorized, their representations begin to become more linearly separable from representations of a set of non-memorized examples.
2. Motivated by these findings, we propose the important and previously-overlooked problem setting of broad unlearning of memorization: reducing the memorization of all memorized examples when given access to only a subset of memorized examples; a task that is only possible due to the fact that memorized examples share representational structure distinct from that of non-memorized examples. This generalizable memorization removal task is a realistic one, as in practice we won't have access to every example that has been memorized, and yet we may require to eliminate memorization broadly.
3. We design a novel unlearning algorithm for broad unlearning of memorization, called BLUR. This algorithm is crafted specifically with representation alignment in mind, based on the findings from our analysis of how memorization manifests in representation space.
4. We show empirically that BLUR reduces the memorization of the memorized examples it intervenes on much more effectively compared to prior work. Further, it also reduces the memorization score of a held-out-from-unlearning set of memorized examples much more effectively compared to prior work. We show that its reduction of memorization is indeed also associated with a reduction in the linear separability between memorized and non-memorized examples. BLUR achieves this goal of broad unlearning of memorization without sacrificing the model's ability to generate high-quality images.

2 Preliminaries and problem formulation

2.1 Diffusion Models

Diffusion models are a class of generative models that learn to reverse a gradual noising process (Ho et al., 2020; Song et al., 2021). Given a data sample x_0, the forward process progressively

adds Gaussian noise over T timesteps, producing a noisy sample $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$ and $\bar{\alpha}_t$ is a predefined noise schedule. A neural network ϵ_θ is trained to predict the added noise at each timestep, minimizing:

$$\mathcal{L}_{\text{DM}} = \mathbb{E}_{x_0,\epsilon,t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right] \tag{1}$$

Latent diffusion models (LDMs) (Rombach et al., 2022) extend this framework by operating in the latent space of a pretrained variational autoencoder (VAE) rather than in pixel space, substantially reducing computational cost. An encoder E maps an image x to a latent representation $z = E(x)$, and the denoising process is performed on z, with the decoded output $\hat{x} = D(z_0)$ recovered via the VAE decoder D. Text-to-image generation is enabled by conditioning the denoising network on a text prompt c, encoded via a text encoder (e.g. CLIP (Radford et al., 2021)) and injected into the UNet through cross-attention layers. The conditioned objective becomes:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{z,\epsilon,t,c}\left[\|\epsilon - \epsilon_\theta(z_t, t, c)\|^2\right] \tag{2}$$

In this work, we focus on SD v1.4 and SD v2.0 as they are the most widely studied models in the memorization literature (Carlini et al., 2023; Somepalli et al., 2023a; Wen et al., 2024), which enable direct comparison with prior work.

2.2 Memorization in T2I Diffusion Models

Text-to-image diffusion models have been shown to memorize training examples: given a prompt from the training set, the model can reproduce a near-identical copy of the associated training image (Carlini et al., 2023; Somepalli et al., 2023a). Prior work has identified two distinct types of memorization (Webster, 2023; Ren et al., 2024). **Verbatim memorization** occurs when a prompt is associated with a single training image, and the model generates a near-exact reproduction of that image given the prompt. **Template memorization** occurs when a prompt is associated with multiple training images; the model then generates outputs that reproduce that structure but vary in non-semantic details. Zhao et al. (2026) first characterised the difference between these two memorization types in terms of their behaviour under unlearning, showing that existing methods perform inconsistently across the two types. We adopt this distinction throughout our experiments and evaluate our method on both verbatim and template memorization separately.

2.3 Unlearning

Unlearning is the problem of post-processing a pretrained model, trained on a dataset $\mathcal{D}_{\text{train}}$, to make it "forget" a subset of its training data, $\mathcal{F} \subset \mathcal{D}_{\text{train}}$ referred to as the "forget set", but without otherwise modifying its utility. This problem is typically formalized as obtaining an unlearned model that is "close" (in distribution) to the ideal unlearned model we would have obtained had we trained from scratch on $\mathcal{D}_{\text{train}} \setminus \mathcal{F}$ Neel et al. (2021); Sekhari et al. (2021); Triantafillou et al. (2024); a process that is considered too computationally expensive in practice. Unlearning is tightly connected with memorization, and the difficulty of unlearning an example depends on how memorized that example is by the model Zhao et al. (2024); Sepahvand et al. (2025).

Triantafillou et al. (2026) recently established a fundamental distinction between two problem formulations that have both been referred to as "machine unlearning" in the literature. *Untraining*, in their terminology, aims to reverse the effect of having trained on a specific forget set $\mathcal{F}$, i.e., removing only the influence of those particular examples, aiming to produce a model that is close to one retrained from scratch on $\mathcal{D}_{\text{train}} \setminus \mathcal{F}$. They contrast that problem with that of *Unlearning* which is about generalization: the goal is to use the forget set $\mathcal{F}$ in order to remove the effect of the entire sub-distribution of examples $\mathcal{M}$ from where that forget set was sampled, aiming to instead produce a model that is close to one retrained from scratch on $\mathcal{D}_{\text{train}} \setminus \mathcal{M}$.

2.4 Broad unlearning of memorization

Inspired by the above distinction, we propose a problem setting in T2I models that requires generalization: our goal is to post-process the trained model to make it "forget" all memorized examples $\mathcal{M}$ using only a subset of identified memorized examples $\mathcal{F} \in \mathcal{M}$. We refer to this as "broad unlearning" of memorization. This is a realistic problem setting, since in practice it is challenging to identify every memorized example. And it's practically important. To study this problem, we build

upon the same metrics used in previous memorization mitigation works but we modify the setup to split the set of examples that are memorized in standard T2I models into an intervention set $\mathcal{F}$ and a "held out" (from the perspective of the unlearning algorithm) set $\mathcal{H}$; so that $\mathcal{M} = \mathcal{F} \cup \mathcal{H}$. The goal of an unlearning algorithm here is to reduce the memorization of all examples in $\mathcal{M}$ given access to only $\mathcal{F}$ and the "retain set" $\mathcal{R}$ of non-memorized examples.

3 Related Work

Memorization mitigation and unlearning. Machine unlearning has been studied across a range of generative model settings. In LLMs, methods such as Negative Preference Optimization (NPO) (Zhang et al., 2024c), Direct Preference Optimization (DPO) (Rafailov et al., 2023), and Selective Gradient Ascent (SGA) (Bărbulescu & Triantafillou, 2024) have been proposed to selectively remove the influence of undesirable or memorized training data. In text-to-image diffusion models, the majority of unlearning research has focused on erasing a particular artistic style or broad visual category from the model (Gandikota et al., 2023; Zhang et al., 2024d; Cywiński & Deja, 2025; Zhang et al., 2024b; Chen et al., 2025). Unlearning memorized training examples is a distinct and comparatively underexplored problem. Existing approaches for memorization mitigation in T2I diffusion models fall into three categories depending on when they are applied: (i) *inference-time intervention*, which suppresses memorized outputs during the generation phase (Somepalli et al., 2023b; Wen et al., 2024; Ren et al., 2024; Han et al., 2025; Jeon et al., 2025; Zhao et al., 2026); (ii) *training-time prevention*, which modifies the training procedure to reduce memorization (Somepalli et al., 2023b; Wen et al., 2024; Ren et al., 2024); and (iii) *unlearning*, which removes the influence of specific memorized examples after training (Alberti et al., 2025). Despite their differences, all existing approaches assume access to the complete set of memorized examples.

Generalizable unlearning. While we are the first to study generalizable unlearning for memorization mitigation in T2I models, previous work has studied generalizable unlearning in other settings. In image classification, Goel et al. (2024); Schoepf et al. (2025) study the problem of unlearning of a set of corrupted or poisoned data points when only a subset of them is available, showing that algorithms that are designed for the case of unlearning all and only the examples given in the forget set fail catastrophically under partial discovery. Zhang et al. (2024a) empirically show that generalization of forgetting in LLMs varies substantially across tasks, and find that tightly clustered representations of forget examples are associated with more generalizable forgetting. In a related direction, Bărbulescu & Triantafillou (2024) propose an output-oriented unlearning goal in LLMs, i.e., unlearning outputs rather than just unlearning training data. They measure success by whether memorized sequences remain extractable at generation time, which is a perspective that aligns with the generalization goal of ensuring no memorized content can be reproduced, even from examples not explicitly targeted during unlearning. In image classification, Zhao et al. (2024) further show that representation distance from the class centroid is a key factor affecting unlearning behavior, reinforcing the role of representational structure in how unlearning propagates.

Evidence for memorization as a generalizable concept. Liu et al. (2026) recently showed that fine-tuning on a narrow subset of memorized content can reactivate verbatim recall far beyond the fine-tuning data, suggesting that interventions on a subset of memorized content can propagate to other memorized examples. This serves as interesting complementary evidence in support of the finding that the property of memorization is encoded in internal representations of models. Our work studies this directly in a causal way, in T2I models, and leverages this finding for developing a novel method that achieves generalizable unlearning of memorization.

4 How is memorization encoded in a T2I model?

We begin by examining how the concept of memorization is represented within the UNet of SD v1.4 and SD v2. This is a fundamental analysis that forms the basis of the method we present in the next section. We seek to answer: Does the embedding space carry information about which examples are memorized? And is there a causal relationship between the memorization status of different training examples and the way that they are arranged in representation space?

Verbatim ; SD v1.4	Template ; SD v1.4	Verbatim ; SD v2.0	Template ; SD v2.0
99.89%	100.00%	96.60%	99.60%

Table 1: **Memorization is linearly decodable from embeddings**. We show the accuracy of a linear classifier to separate memorized from non-memorized examples, across memorization types (verbatim and template) and architectures (SD v1.4 and SD v2.0). See Section 4.1 for details.

4.1 Memorization is Linearly Decodable from Representations

We begin by investigating whether there is any embedding-space separability between memorized and non-memorized examples. We train a logistic regression classifier for the associated supervised binary classification problem. We refer to the memorized examples identified by Webster (2023) as the memorized set $\mathcal{M}$ and the rest of the dataset as the non-memorized set $\mathcal{R}$. We first extract the pooled hidden representations from the U-Net mid-block of each example. We then train a logistic-regression classifier using these representations. We balance the two classes (memorized vs non-memorized examples), repeat the experiment over five random samplings of the subset of $\mathcal{R}$ used to train the classifier (since $\mathcal{R}$ is larger than $\mathcal{M}$, we take a subset of it each time, to balance the two classes), and use cross-validation for model selection and evaluation.

Table 1 reports the resulting classification accuracies across model architectures and memorization types. For SD v1.4 and SD v2, memorized and non-memorized examples can be distinguished with accuracy above 96% for both verbatim and template memorization. This shows that whether an example is memorized is almost perfectly linearly decodable from the model's internal representations. We additionally provide PCA visualizations in Figure 2 in Appendix A.2.3, which qualitatively show a similar separation between memorized and non-memorized examples.

4.2 Evidence of a Causal Link between Memorization and Representations

Next, we examine whether there is a causal relationship between a set of examples being memorized and their linear separability from non-memorized examples. Alternatively, it could have been the case that the examples that are memorized happened to also be semantically related, and that semantic relationship could also have explained the relatedness of their embeddings.



Figure 1: Applying an intervention to induce memorization on a set of examples $\mathcal{I}$ increases both its mean memorization score (SSCD) and its linear separability from the set of non-memorized examples $\mathcal{R} \setminus \mathcal{I}$; the unlearning method we develop in Section 5, when using $\mathcal{I}$ as its forget set, is able to decrease both.

To investigate this, we perform two types of causal interventions. First, we construct a causal intervention whereby we artificially induce memorization of a random subset of examples (that were initially not memorized) and then inspect how the increase of those examples' memorization score affects their embedding relationships with the set of non-memorized examples. Specifically, we construct an intervention set $\mathcal{I} \in \mathcal{R}$ containing 200 examples that are initially not memorized in SD v1.4. We then fine-tune SD v1.4 on duplicated copies of these examples together with the rest of $\mathcal{R}$. This procedure was previously proposed in prior work for obtaining an "artificially-memorized" set of examples Somepalli et al. (2023b); Wen et al. (2024); Ren et al. (2024). In our experiment, this intervention substantially increases the memorization score of the examples in $\mathcal{I}$, with mean SSCD score rising from 0.077 to 0.654. We then extract representations of these examples before and after memorization induction, and train the same logistic-regression classifier to distinguish the sets $\mathcal{I}$ and $\mathcal{R} \setminus \mathcal{I}$. Here, $\mathcal{I}$ have become memorized via our intervention, whereas $\mathcal{R} \setminus \mathcal{I}$ remain non-memorized, so the ability of the classifier to tell these two sets apart after the intervention, when it was not able to initially tell them

apart, suggests that inducing memorization on $\mathcal{I}$ is the cause for the linear separability of these two sets. Indeed, as shown in Figure 1, increasing memorization of the intervention set is accompanied by a large increase in linear separability: balanced classification accuracy rises from 65.5% before induction to 93.3% after induction. This provides causal evidence that inducing memorization of the examples in $\mathcal{I}$ alters their internal representations in a way that makes them substantially more linearly separable from non-memorized examples.

We also perform a different causal intervention using steering vectors (Turner et al., 2023) in the Appendix A.2.1, showing evidence that we can create a single direction in embeddings along which we can increase or decrease the memorization score to some degree.

Together, our results so far suggest that memorization is linearly decodable from internal representations, both in the case of naturally-occurring memorization (Section 4.1) and in the case of artificially-induced memorization (Section 4.2). And we have found evidence of a causal relationship between memorization and representational separability. A natural next question is whether we can design unlearning algorithms that leverage this relationship we have identified between memorization and representation-space relationships, in order to more effectively reduce memorization, across a broad set of memorized examples.

5 BLUR: Broad Local Unlearning of Representations

5.1 Motivating BLUR

In the previous section, we observed that memorization is linearly decodable from the model's embeddings. We showed evidence of a causal relationship between memorization of a set of examples and representational separability of that set of examples from a set of non-memorized examples. We also showed that embedding-space steering can, to some degree, control the degree of memorization of a set of examples by moving along a single "memorization direction". That experiment is useful for providing evidence of a causal pattern, but it does not alone constitute an optimal generalizable unlearning method, since that single direction does not fully explain memorization.

In this section, we propose a fine-tuning procedure for effective broad unlearning of memorization. Our objective for this unlearning phase is drawn from our findings in Section 4: given the finding that memorized examples are separable from non-memorized ones in representation space, we propose to explicitly fine-tune to align the representations of the forget set with those of the retain set, so as to reduce the memorization score of the forget set examples. This fine-tuning phase allows a more flexible nonlinear transformation compared to the pure steering that we investigated in Section 4.

Further, since it's unlikely that there exists a single "memorization direction" that optimally reduces memorization of all examples without hurting generation quality, we design our objective to operate in local neighbourhoods: it moves the representations of the examples in the forget set closer to the representations of their deemed-to-be-most-similar non-memorized examples from $\mathcal{R}$. We refer to this approach as Broad Local Unlearning of Representations (BLUR).

Because of our observations in Section 4, showing that there is shared representational structure between memorized examples that is separable from non-memorized examples, we hypothesize that BLUR can achieve broad unlearning: it can effectively reduce the degree of memorization of all examples, even when intervening on only a subset of them.

5.2 Introducing BLUR

Let $h_\theta(x, c)$ denote the pooled U-Net mid-block representation of an example (x, c) extracted from the diffusion model ϵ_θ. We use θ_0 to denote the original pretrained model parameters and θ to denote the parameters being optimized during unlearning. Using the pretrained model, we first construct a fixed retain representation bank $\mathcal{B}_\mathcal{R} = h_{\theta_0}(x, c) : (x, c) \in \mathcal{R}$. For each forget example $(x, c) \in \mathcal{F}$, we retrieve its k nearest retain examples by comparing its current representation $h_\theta(x, c)$ against $\mathcal{B}_\mathcal{R}$ using Euclidean distance. Let $\mathcal{N}_k(x, c; \theta) \subset \mathcal{R}$ denote these neighbors. Their mean representation defines the local (to a forget example x) retain anchor:

$$\mu_k(x, c; \theta) = \frac{1}{k} \sum_{(x', c') \in \mathcal{N}_k(x, c; \theta)} h_{\theta_0}(x', c'). \tag{3}$$

We then align each identified memorized example toward its local retain anchor:

$$\mathcal{L}_{\text{align}}(\theta) = \frac{1}{|\mathcal{F}|} \sum_{(x,c)\in\mathcal{F}} \|h_\theta(x,c) - \mu_k(x,c;\theta)\|_2^2. \tag{4}$$

To preserve generation utility, we additionally apply the standard diffusion denoising objective from Eq.(2) to examples from the retain set $\mathcal{R}$, which we denote by $\mathcal{L}_{\text{retain}}(\theta)$. Our final unlearning objective is

$$\theta^\star = \arg\min_\theta \left[\mathcal{L}_{\text{retain}}(\theta) + \lambda \mathcal{L}_{\text{align}}(\theta)\right], \tag{5}$$

where λ controls the strength of representation alignment. In this way, the objective uses only the identified forget set $\mathcal{F}$ to target a representation-space structure shared more broadly across memorized examples, which enables the unlearning effect to generalize to unidentified memorized examples in $\mathcal{H}$.

6 Experiments

6.1 Experimental Setup

Models and datasets. Following prior work, we use SD v1.4 and v2.0, pretrained on the LAION dataset (Schuhmann et al., 2022) as our base models, and use LAION Aesthetics 6.5+, a curated high-quality subset of LAION as the retain set. For the memorized set $\mathcal{M}$, we adopt the prompts identified by Webster (2023). As defined in Section 2.2, we consider both *verbatim* and *template* memorization. We further filter these candidate examples using an SSCD threshold of > 0.1, yielding 145 examples exhibiting verbatim memorization and 258 examples exhibiting template memorization. For SD v1.4, we evaluate both memorization types separately, as mitigation methods can behave differently across them (Zhao et al., 2026). For SD v2.0, we focus on template memorization, since prior work finds verbatim memorization to be substantially rarer in later Stable Diffusion models (Webster, 2023).

Baselines. We compare against three groups of baselines: (i) **No mitigation**, where we directly use the pretrained model SD v1.4/v2.0. (ii) **Fine-tune**, which fine-tunes the model only on the retain set and serves as a simple baseline. (iii) recent state-of-the-art unlearning methods, including **SISS**, which defines example-level unlearning objective based on Subtracted Importance Sampled Scores (Alberti et al., 2025); **NegGrad+**, which performs gradient ascent on the forget set and gradient descent on the retain set (Kurmanji et al., 2023); **RTA**, which appends random tokens to memorized prompts during unlearning (Somepalli et al., 2023b); and **Wen et al.**, which dynamically identifies likely memorized examples during fine-tuning and excludes them from the current mini-batch (Wen et al., 2024). In our setting, we adapt it as a post-hoc unlearning baseline by applying the same filtering mechanism during the unlearning stage.

Metrics. We evaluate each method along three axes: memorization level, measured by SSCD (Pizzi et al., 2022) (lower is better); text-image alignment, measured by CLIP score (Radford et al., 2021) (higher is better); and generation quality, measured by FID (Heusel et al., 2017) (lower is better).

6.2 Evaluation protocol

Splitting the set of memorized examples, to study generalization of unlearning. To evaluate whether unlearning generalizes to memorized examples *not seen* during unlearning, we partition the memorized set $\mathcal{M}$ into equally sized forget and held-out sets, $\mathcal{F}$ and $\mathcal{H}$, respectively. During unlearning, the method has access only to $\mathcal{F}$, while $\mathcal{H}$ is held out entirely. Our primary evaluation uses a **Low→High** memorization split. Specifically, we rank memorized examples by their pre-unlearning SSCD scores and assign the lower-memorization half to $\mathcal{F}$ and the higher-memorization half to $\mathcal{H}$. This is the most challenging of the settings we consider, as the unlearning algorithm can intervene only on relatively weakly memorized examples and must generalize to held-out examples that are substantially more strongly memorized. We additionally evaluate High→Low as a complementary control, with details in Appendix A.1.2 and full results in Appendix A.3.

We report overall SSCD and CLIP on the full memorized set $\mathcal{M} = \mathcal{F} \cup \mathcal{H}$, together with SSCD evaluated separately on $\mathcal{F}$ and $\mathcal{H}$, so that performance on the held-out subset $\mathcal{H}$ directly measures cross-example generalization. We additionally report the intervention–held-out gap $\Delta_{\text{gen}} =$

Table 2: Broad unlearning results for different memorization types and architectures. For each method, we report the configuration with the lowest overall SSCD subject to overall CLIP ≥ 0.276. Overall metrics are evaluated on $\mathcal{F} \cup \mathcal{H}$, while $\text{SSCD}_{\mathcal{F}}$ and $\text{SSCD}_{\mathcal{H}}$ are evaluated separately on the set of memorized examples used by the unlearning algorithm as the forget set ($\mathcal{F}$) and the held-out set of memorized examples ($\mathcal{H}$). Best and second-best results among mitigation methods are shown in **bold** and underlined, respectively. Low $\rightarrow$ High memorization. **BLUR achieves by far the largest reduction in $\text{SSCD}_{\mathcal{F}}$ while also generalizing far better than previous methods.**

(a) SD v1.4 - Verbatim memorization

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{SSCD}_{\mathcal{H}}$ ↓	Δ_{gen} ↓
No mitigation	0.581 ± 0.052	0.338 ± 0.008	163.946	0.292 ± 0.042	0.866 ± 0.022	0.574
BLUR (ours)	$\mathbf{0.177 \pm 0.033}$	0.290 ± 0.011	**117.232**	$\mathbf{0.085 \pm 0.017}$	$\mathbf{0.267 \pm 0.056}$	**0.182**
Fine-tune	0.429 ± 0.046	0.330 ± 0.009	143.502	0.217 ± 0.031	0.638 ± 0.053	0.421
SISS	0.325 ± 0.036	0.318 ± 0.003	151.376	0.178 ± 0.025	0.469 ± 0.048	0.291
NegGrad+	0.349 ± 0.041	0.320 ± 0.009	134.732	0.186 ± 0.027	0.509 ± 0.058	0.323
RTA	0.423 ± 0.047	0.324 ± 0.009	142.957	0.210 ± 0.034	0.634 ± 0.055	0.424
Wen et al.	0.299 ± 0.041	0.315 ± 0.008	128.814	0.166 ± 0.026	0.429 ± 0.065	0.263

(b) SD v1.4 - Template memorization

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{SSCD}_{\mathcal{H}}$ ↓	Δ_{gen} ↓
No mitigation	0.671 ± 0.021	0.317 ± 0.004	223.766	0.529 ± 0.024	0.813 ± 0.009	0.284
BLUR (ours)	$\mathbf{0.248 \pm 0.019}$	0.278 ± 0.006	**156.054**	$\mathbf{0.195 \pm 0.020}$	$\mathbf{0.301 \pm 0.026}$	**0.106**
Fine-tune	0.513 ± 0.024	0.309 ± 0.005	194.960	0.396 ± 0.027	0.630 ± 0.028	0.234
SISS	0.312 ± 0.012	0.278 ± 0.005	198.666	0.256 ± 0.014	0.368 ± 0.017	0.112
NegGrad+	0.521 ± 0.024	0.307 ± 0.005	199.630	0.397 ± 0.027	0.646 ± 0.026	0.249
RTA	0.543 ± 0.024	0.287 ± 0.006	205.745	0.440 ± 0.032	0.645 ± 0.024	0.205
Wen et al.	0.356 ± 0.022	0.286 ± 0.005	183.856	0.303 ± 0.026	0.409 ± 0.035	**0.106**

(c) SD v2.0 - Template memorization

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{SSCD}_{\mathcal{H}}$ ↓	Δ_{gen} ↓
No mitigation	0.545 ± 0.029	0.307 ± 0.005	218.950	0.366 ± 0.036	0.722 ± 0.011	0.356
BLUR (ours)	$\mathbf{0.121 \pm 0.006}$	0.279 ± 0.009	**139.552**	$\mathbf{0.119 \pm 0.010}$	$\mathbf{0.123 \pm 0.007}$	**0.004**
Fine-tune	0.499 ± 0.028	0.311 ± 0.005	210.774	0.328 ± 0.035	0.668 ± 0.016	0.340
SISS	0.296 ± 0.014	0.290 ± 0.005	207.295	0.213 ± 0.015	0.379 ± 0.011	0.166
NegGrad+	0.266 ± 0.017	0.285 ± 0.006	180.318	0.166 ± 0.018	0.366 ± 0.015	0.200
RTA	0.381 ± 0.022	0.286 ± 0.005	199.433	0.314 ± 0.033	0.447 ± 0.023	0.133
Wen et al.	0.234 ± 0.017	0.275 ± 0.006	300.831	0.224 ± 0.026	0.244 ± 0.021	0.020

$|\text{SSCD}_{\mathcal{H}} - \text{SSCD}_{\mathcal{F}}|$. A smaller Δ_{gen} indicates that the memorization reduction achieved on the intervention set transfers more successfully to the held-out set, so lower is better. Since FID is a distributional metric sensitive to sample size, we report it on the full memorized set $\mathcal{M}$.

Reporting protocol. Since memorization reduction can be trivially improved by severely degrading the generated outputs, we use CLIP as a coarse guardrail against substantial loss of generation quality. Following Zhao et al. (2026), we compute the average CLIP score on the retain set $\mathcal{R}$ using the original pretrained model as a reference, obtaining 0.307, which represents the CLIP level expected of a non-memorized example under the model. We allow up to a 10% degradation of the reference CLIP score, yielding a minimum acceptable CLIP score of 0.276, and we then report the best achievable results in terms of SSCD, CLIP, and FID within this range. Importantly, we note that small CLIP differences within this acceptable range should not be interpreted as meaningful differences in generation quality; we further illustrate this with qualitative examples in Appendix A.5. As a complementary evaluation, we reverse this selection criterion in Appendix A.4: we tune each baseline to approach the SSCD achieved by our method without imposing the CLIP constraint, and report the resulting CLIP and FID. This directly measures the utility cost required by competing methods to attain a comparable level of memorization reduction.

6.3 Results

Tables 2 report results for verbatim and template memorization in the challenging Low $\rightarrow$ High setting, where the intervention set $\mathcal{F}$ contains the less strongly memorized examples and the held-out set $\mathcal{H}$ contains the more strongly memorized examples. Full results, including per-subset CLIP scores, are reported in Appendix A.3.

Table 3: Attempting to tune other methods to match BLUR's low SSCD score leads to a significant degradation to their CLIP score (template memorization on SD v1.4). "—" indicates that no evaluated configuration reaches BLUR's SSCD. Full results are reported in Appendix A.4.

Method	**SSCD** $\downarrow$	**CLIP** $\uparrow$	**FID** $\downarrow$
BLUR (ours)	0.248	0.278	156.054
Fine-tune	—	—	—
NegGrad+	0.075	0.134	313.3
SISS	0.070	0.138	315.2
RTA	—	—	—
Wen et al.	0.231	0.265	197.578

For verbatim memorization in Table 2a, our method reduces overall SSCD from 0.581 to 0.177, compared with 0.299 for the strongest baseline. More importantly, despite intervening only on the less strongly memorized subset, it reduces $\mathrm{SSCD}_{\mathcal{H}}$ on the held-out high-memorization examples from 0.866 to 0.267. The corresponding generalization gap is 0.182, substantially smaller than those of most competing methods, indicating that the reduction achieved on $\mathcal{F}$ transfers effectively to the harder held-out examples.

A similar pattern holds for template memorization in Table 2b. Our method reduces overall SSCD from 0.671 to 0.248, while reducing memorization on the held-out subset from $\mathrm{SSCD}_{\mathcal{H}} = 0.813$ to 0.301. Its generalization gap of 0.106 shows that the intervention extends substantially beyond the examples directly used during unlearning.

The same behavior transfers to SD v2.0 in Table 2c. Our method reduces overall SSCD from 0.545 to 0.121, substantially below the strongest baseline at 0.234, while reducing held-out memorization from $\mathrm{SSCD}_{\mathcal{H}} = 0.722$ to 0.123, with a generalization gap of only 0.004. It also achieves the lowest FID (139.6), suggesting that this stronger memorization suppression does not come from broadly degrading generation quality.

Table 3 further compares methods at approximately matched SSCD levels to directly examine the utility cost required to attain comparable memorization reduction. Most baselines do not reach BLUR's SSCD within the evaluated range; NegGrad+ and SISS can be pushed below it only with severe utility degradation, while Wen et al. can achieve a comparable SSCD but only at a large cost to FID. Full matched-SSCD results are reported in Appendix A.4. Results for the easier High$\rightarrow$Low direction are reported in Appendix A.3, where they further illustrate how generalization difficulty depends on which memorized examples are available for intervention.

Finally, we observe from Figure 1 that BLUR's success in reducing memorization is indeed associated with a reduction of linear separability of the intervention set from non-memorized examples, confirming its representational alignment mechanism is working as intended.

Taken together, these results show that BLUR achieves substantially broader unlearning of memorization while preserving generation quality. It generalizes from identified memorized examples to held-out examples that are unseen during intervention, including more strongly memorized examples, and does so with a markedly more favorable memorization-utility trade-off than existing baselines.

7 Conclusion

We made the foundational discovery that memorization is linearly decodable from internal representations of T2I models. The representational structure we discovered has opened the door to studying the important and previously-overlooked problem setting of broad unlearning of memorization. We proposed BLUR, a novel method for broad unlearning of memorization, motivated by our analysis of how memorization manifests in representation space. Our key insight is that viewing the problem as representation alignment of the distributions of the retain and forget set examples yields a method that is able to both lower the SSCD of the forget set of memorized examples much more than prior work, while also generalizing much better than prior methods at reducing SSCD of other unseen memorized examples too, all while maintaining the model's ability to generate high-quality images.

AI USE STATEMENT

We used generative AI tools to assist with implementation and code development, and proofread and polish the manuscript. All AI-assisted content was reviewed and verified by the authors. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies machine unlearning as a means of mitigating memorization in text-to-image diffusion models. Such methods may help reduce risks associated with unintended reproduction of training data, including privacy and copyright concerns. However, our method should not be interpreted as providing a formal guarantee of data removal, privacy, or compliance with legal notions of deletion. Our experiments use existing pretrained models and research datasets and do not involve human subjects or the collection of new personal data. We report our methodology and evaluation transparently to facilitate further research on the capabilities and limitations of machine unlearning.

REPRODUCIBILITY STATEMENT

We provide the details necessary to reproduce our experiments throughout the main paper and appendix. The experimental setup, including the construction of intervention and held-out forget sets, models, datasets, baselines, and evaluation metrics, is described in Section 6. Additional implementation details, hyperparameters, and experimental settings are provided in Appendix A.1. We report results across multiple random seeds where applicable and include confidence intervals for our main evaluations. We also provide anonymized code at: `https://anonymous.4open.science/status/generalizable-unlearning-2012`.

REFERENCES

Silas Alberti, Kenan Hasanaliyev, Manav Shah, and Stefano Ermon. Data unlearning in diffusion models. *arXiv preprint arXiv:2503.01034*, 2025.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

George-Octavian Bărbulescu and Peter Triantafillou. To each (textual sequence) its own: improving memorized-data unlearning in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24, 2024.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pp. 2633–2650, 2021.

Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Tianqi Chen, Shujian Zhang, and Mingyuan Zhou. Score forgetting distillation: A swift, data-free method for machine unlearning in diffusion models, 2025. URL `https://arxiv.org/abs/2409.11219`.

Bartosz Cywiński and Kamil Deja. Saeuron: Interpretable concept unlearning in diffusion models with sparse autoencoders, 2025. URL `https://arxiv.org/abs/2501.18052`.

European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council, 2016. URL `https://data.europa.eu/eli/reg/2016/679/oj`.

European Parliament and Council of the European Union. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, OJ L 2024/1689, July 2024. URL https://eurlex.europa.eu/eli/reg/2024/1689/oj/eng.

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.

Shashwat Goel, Ameya Prabhu, Philip Torr, Ponnurangam Kumaraguru, and Amartya Sanyal. Corrective machine unlearning. *arXiv preprint arXiv:2402.14015*, 2024.

Hyeonggeun Han, Sehwan Kim, Hyungjun Joo, Sangwoo Hong, and Jungwoo Lee. Adjusting initial noise to mitigate memorization in text-to-image diffusion models. *arXiv preprint arXiv:2510.08625*, 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, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Dongjae Jeon, Dueun Kim, and Albert No. Understanding and mitigating memorization in generative models via sharpness of probability landscapes. In *Forty-second International Conference on Machine Learning*, 2025.

Dmitrii Krasheninnikov, Richard E Turner, and David Krueger. Fresh in memory: Training-order recency is linearly encoded in language model activations. In *International Conference on Learning Representations*, volume 2026, pp. 62177–62197, 2026.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Proceedings of Advances in Neural Information Processing Systems*, 2023.

Xinyue Liu, Niloofar Mireshghallah, Jane C Ginsburg, and Tuhin Chakrabarty. Alignment whack-a-mole: Finetuning activates verbatim recall of copyrighted books in large language models. *COLM*, 2026.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *Algorithmic Learning Theory*, pp. 931–962. PMLR, 2021.

Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14532–14542, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Jie Ren, Yaxin Li, Shenglai Zeng, Han Xu, Lingjuan Lyu, Yue Xing, and Jiliang Tang. Unveiling and mitigating memorization in text-to-image diffusion models through cross attention. In *European Conference on Computer Vision (ECCV)*, 2024.

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.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Stefan Schoepf, Michael Curtis Mozer, Nicole Elyse Mitchell, Alexandra Brintrup, Georgios Kaissis, Peter Kairouz, and Eleni Triantafillou. Redirection for erasing memory (rem): Towards a universal unlearning method for corrupted data. *arXiv preprint arXiv:2505.17730*, 2025.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.

Nazanin Mohammadi Sepahvand, Anvith Thudi, Berivan Isik, Ashmita Bhattacharyya, Nicolas Papernot, Eleni Triantafillou, Daniel M Roy, and Gintare Karolina Dziugaite. Leveraging per-instance privacy for machine unlearning. *arXiv preprint arXiv:2505.18786*, 2025.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6048–6058, 2023a.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023b.

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

Eleni Triantafillou, Peter Kairouz, Fabian Pedregosa, Jamie Hayes, Meghdad Kurmanji, Kairan Zhao, Vincent Dumoulin, Julio Jacques Junior, Ioannis Mitliagkas, Jun Wan, et al. Are we making progress in unlearning? findings from the first neurips unlearning competition. *arXiv preprint arXiv:2406.09073*, 2024.

Eleni Triantafillou, Ahmed Imtiaz Humayun, Monica Ribero, Alexander Matt Turner, Michael C Mozer, and Georgios Kaissis. Is your algorithm unlearning or untraining? *arXiv preprint arXiv:2604.07962*, 2026.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Ryan Webster. A reproducible extraction of training images from diffusion models. *arXiv preprint arXiv:2305.08694*, 2023.

Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Eric Zhang, Leshem Chosen, and Jacob Andreas. Unforgettable generalization in language models. *arXiv preprint arXiv:2409.02228*, 2024a.

Gong Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Forget-me-not: Learning to forget in text-to-image diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1755–1764, 2024b.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024c.

Yihua Zhang, Chongyu Fan, Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jiancheng Liu, Gaoyuan Zhang, Gaowen Liu, Ramana Rao Kompella, Xiaoming Liu, et al. Unlearncanvas: Stylized image dataset for enhanced machine unlearning evaluation in diffusion models. *arXiv preprint arXiv:2402.11846*, 2024d.

Kairan Zhao, Meghdad Kurmanji, George-Octavian Bărbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. In *Proceedings of Advances in Neural Information Processing Systems*, 2024.

Kairan Zhao, Eleni Triantafillou, and Peter Triantafillou. You don't need all that attention: Surgical memorization mitigation in text-to-image diffusion models. *Forty-third International Conference on Machine Learning*, 2026.

A Appendix

Contents

A.1 Implementation Details 13
A.1.1 Training Configuration 13
A.1.2 Additional Generalizable-Unlearning Evaluations 13
A.2 Representation Analysis 14
A.2.1 Activation Steering Along the Memorization Direction 14
A.2.2 Quantifying the Representation Structure Behind Section 4 14
A.2.3 Representation Alignment Achieved by Our Method 15
A.3 Full results for generalizable unlearning 16
A.4 Memorization-utility trade-off at matched SSCD 17
A.5 Qualitative Examples 18

A.1 Implementation Details

A.1.1 Training Configuration

We fine-tune SD v1.4 and v2.0 using AdamW optimizer with learning rate 1×10^{-5}, batch size 16, for 20000 steps. For the kNN-based local alignment loss, we use $k \in [1, 500]$ and $\lambda \in [1 \times 10^{-5}, 1 \times 10^{-4}]$. All experiments are run on NVIDIA A100 and L40 GPUs.

A.1.2 Additional Generalizable-Unlearning Evaluations

We consider three ways of partitioning the memorized set $\mathcal{M}$ into an intervention subset $\mathcal{F}$ and a held-out subset $\mathcal{H}$. In all cases, the two subsets have equal size, and only $\mathcal{F}$ is available to the unlearning algorithm. The different partitions control how similar the observed and held-out memorized examples are in terms of their initial memorization strength.

Low→High memorization. We rank examples by their SSCD under the pretrained model and assign the lower-memorization half to $\mathcal{F}$ and the higher-memorization half to $\mathcal{H}$. The method must therefore generalize from examples exhibiting relatively weaker memorization to examples exhibiting substantially stronger memorization. Among the two splits, this produces the most challenging generalization setting and is therefore used as our primary evaluation in the main paper.

High→Low memorization. We reverse the partition above: the more strongly memorized half forms $\mathcal{F}$, while the less strongly memorized half forms $\mathcal{H}$. This setting evaluates the converse direction of generalization. Because intervention is performed on the more strongly memorized examples while the held-out examples are initially easier to mitigate, it serves primarily as a complementary sanity check supporting the observation that generalization difficulty depends on which examples are available for intervention.

For both settings, we report overall SSCD, CLIP, and FID, as well as $\mathrm{SSCD}_{\mathcal{F}}$, $\mathrm{SSCD}_{\mathcal{H}}$, and subset-level CLIP scores in the full appendix tables.

A.2 Representation Analysis

A.2.1 Activation Steering Along the Memorization Direction

As an additional causal probe of the representational structure identified in Section 4.2, we directly intervene on the model's internal representations during inference. We define a fixed memorization direction as

$$v_{\mathrm{mem}} = \mu_{\mathcal{F}} - \mu_{\mathcal{R}}. \tag{6}$$

where $\mu_{\mathcal{F}}$ and $\mu_{\mathcal{R}}$ are the representation centroids of the identified memorized and retained non-memorized examples, respectively. We steer the U-Net activations by adding αv_{mem}, such that positive α moves activations toward the memorized region and negative α moves them in the opposite direction.

Table 4 summarizes the results. At moderate steering strengths, we observe the expected directional effect: relative to $\alpha = 0$ (SSCD 0.525), SSCD increases to 0.534 at $\alpha = +16$ and decreases to 0.517 at $\alpha = -16$, while CLIP remains nearly unchanged. This difference between $\alpha = +16$ and $\alpha = -16$ is statistically significant, providing complementary causal evidence that directly manipulating the memorization-related representation direction can affect output-level memorization.

At larger steering strengths, both directions eventually reduce SSCD, although negative steering consistently produces a stronger reduction than positive steering (Table 4). This indicates that memorization does not appear to be fully characterized by a single global linear steering direction. Instead, our proposed method in Section 5 uses this representational insight to define an alignment objective, while allowing nonlinear parameter updates through fine-tuning to achieve memorization reduction.

Table 4: Activation steering along the memorization direction. Changes in SSCD and CLIP are reported relative to the unsteered model ($\alpha = 0$). Positive and negative α steer activations toward and away from the memorized region, respectively.

$\lvert\alpha\rvert$	ΔSSCD ($+\alpha$)	ΔSSCD ($-\alpha$)	ΔCLIP ($+\alpha$)	ΔCLIP ($-\alpha$)
2	$+0.001$	-0.003	≈ 0	≈ 0
4	$+0.004$	-0.002	≈ 0	≈ 0
8	$+0.007$	-0.001	≈ 0	≈ 0
16	$+0.009$	-0.008	≈ 0	≈ 0
32	-0.004	-0.039	-0.002	-0.006
64	-0.049	-0.139	-0.004	-0.024
128	-0.144	-0.192	-0.025	-0.032
256	-0.195	-0.218	-0.043	-0.039
512	-0.189	-0.240	-0.035	-0.031

A.2.2 Quantifying the Representation Structure Behind Section 4

To quantify the separation between the forget and the retain set in Section 4, we measure two complementary statistics: (i) the mean pairwise cosine similarity between forget-set and retain-set representations, and (ii) the Euclidean distance between the forget-set and retain-set centroids. Lower pairwise cosine similarity and larger centroid distance indicate greater distributional separation. As shown in Table 5, before unlearning, the forget and retain examples have low cosine similarity for both verbatim memorization (0.055 ± 0.049) and template memorization (0.097 ± 0.048), while their

centroid distances are large (22.852 and 32.651, respectively). These results confirm a substantial representation gap between memorized and non-memorized examples.

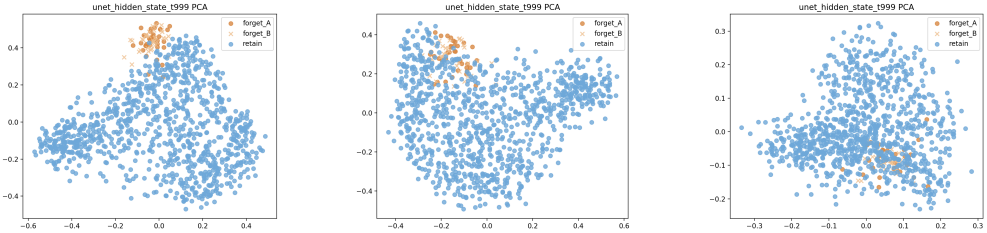
We further quantify how this gap changes after applying an existing unlearning method of Wen et al. (2024). Although this intervention reduces output-level memorization on both sets $\mathcal{F}$ and $\mathcal{H}$, it does not achieve reliable representation alignment between the forget and retain sets. As shown in Table 5, for verbatim memorization, mean pairwise forget-retain cosine similarity increases only modestly from 0.055 ± 0.049 to 0.135 ± 0.066, while the centroid distance even increases from 22.852 to 30.547; a similar pattern holds for template memorization. This indicates that the two distributions are not reliably brought closer together.

Table 5: Cosine similarity and centroid distance between memorized-set and retain-set representations in the UNet mid-block hidden state space, measured pre- and post-unlearning for a representative existing method. Higher cosine similarity and lower centroid distance indicate better alignment of the memorized distribution with the retain distribution.

	Verbatim memorization		Template memorization	
Method	$\mathcal{M}$-$\mathcal{R}$ cosine sim. (↑)	$\mathcal{M}$-$\mathcal{R}$ centroid dist. (↓)	$\mathcal{M}$-$\mathcal{R}$ cosine sim. (↑)	$\mathcal{M}$-$\mathcal{R}$ centroid dist. (↓)
Pre-unlearning	0.055 (0.049)	22.852	0.097 (0.048)	32.651
Post-unlearning	0.135 (0.066)	30.547	0.162 (0.053)	41.524

A.2.3 Representation Alignment Achieved by Our Method

Figure 2 and Table 6 further report representation visualization and statistics for our methods alongside the baselines introduced in Section 6.1. We can observe that our proposed method substantially closes the representational gap relative to other baselines. As shown in Table 6, our method achieves strong alignment on verbatim (0.826 ± 0.046, centroid distance 17.606) and template (0.774 ± 0.050, centroid distance 26.842) memorization. This shows a large improvement over both the pre-unlearning baseline ("No mitigation") and the existing method (Wen et al.).



(a) Pre-unlearning (b) Post-unlearning: Wen et al. (c) Post-unlearning: ours

Figure 2: Visualization of the U-Net hidden states for the memorized and non-memorized examples. (a) Before unlearning (i.e., using the pretrained model), the memorized and non-memorized examples are quite clearly separated. (b,c) After applying an unlearning algorithm: the method of Wen et al. (2024) / our proposed method, the memorized and non-memorized examples become less separated.

Table 6: Cosine similarity and centroid distance between memorized-set and retain-set representations in the UNet mid-block hidden state space, measured using the pretrained model ("No mitigation") and different unlearning methods. Higher cosine similarity and lower centroid distance indicate better alignment of the memorized distribution with the retain distribution.

	Verbatim memorization		Template memorization	
Method	$\mathcal{M}$-$\mathcal{R}$ cosine sim. (↑)	$\mathcal{M}$-$\mathcal{R}$ centroid dist. (↓)	$\mathcal{M}$-$\mathcal{R}$ cosine sim. (↑)	$\mathcal{M}$-$\mathcal{R}$ centroid dist. (↓)
No mitigation	0.055 (0.049)	22.852	0.097 (0.048)	32.651
Wen et al.	0.135 (0.066)	30.547	0.162 (0.053)	41.524
Ours	0.826 (0.046)	17.606	0.774 (0.050)	26.842

Table 7: Full generalizable-unlearning results under different intervention/held-out splits for **SD v1.4 - verbatim memorization**. We report mean ± 95% confidence interval for SSCD and CLIP. Overall metrics are evaluated on $\mathcal{F} \cup \mathcal{H}$, while subset-level metrics are evaluated separately on $\mathcal{F}$ and $\mathcal{H}$.

(a) Low → High memorization.

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.581 ± 0.052	0.338 ± 0.008	163.946	0.292 ± 0.042	0.330 ± 0.011	0.866 ± 0.022	0.345 ± 0.011
BLUR (ours)	$\mathbf{0.177 \pm 0.033}$	0.290 ± 0.011	**117.232**	$\mathbf{0.085 \pm 0.017}$	0.268 ± 0.016	$\mathbf{0.267 \pm 0.056}$	0.312 ± 0.015
Fine-tune	0.429 ± 0.046	0.330 ± 0.009	143.502	0.217 ± 0.031	0.319 ± 0.013	0.638 ± 0.053	0.340 ± 0.011
SISS	0.325 ± 0.036	0.318 ± 0.003	151.376	0.178 ± 0.025	0.315 ± 0.013	0.469 ± 0.048	0.321 ± 0.011
NegGrad+	0.349 ± 0.041	0.320 ± 0.009	134.732	0.186 ± 0.027	0.307 ± 0.013	0.509 ± 0.058	0.332 ± 0.011
RTA	0.423 ± 0.047	0.324 ± 0.009	142.957	0.210 ± 0.034	0.311 ± 0.014	0.634 ± 0.055	0.336 ± 0.012
Wen et al.	$\underline{0.299 \pm 0.041}$	0.315 ± 0.008	$\underline{128.814}$	$\underline{0.166 \pm 0.026}$	0.303 ± 0.011	$\underline{0.429 \pm 0.065}$	0.327 ± 0.010

(b) High → Low memorization.

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.581 ± 0.052	0.338 ± 0.008	163.946	0.866 ± 0.022	0.345 ± 0.011	0.292 ± 0.042	0.330 ± 0.011
BLUR (ours)	$\mathbf{0.121 \pm 0.015}$	0.292 ± 0.010	**119.802**	$\mathbf{0.091 \pm 0.016}$	0.287 ± 0.013	$\mathbf{0.152 \pm 0.024}$	0.297 ± 0.015
Fine-tune	0.429 ± 0.046	0.330 ± 0.009	143.502	0.638 ± 0.053	0.340 ± 0.011	0.217 ± 0.031	0.319 ± 0.013
SISS	$\underline{0.246 \pm 0.025}$	0.314 ± 0.008	143.212	$\underline{0.325 \pm 0.037}$	0.317 ± 0.011	0.167 ± 0.023	0.311 ± 0.012
NegGrad+	0.370 ± 0.039	0.326 ± 0.008	136.047	0.524 ± 0.049	0.337 ± 0.010	0.214 ± 0.033	0.314 ± 0.012
Wen et al.	0.312 ± 0.043	0.318 ± 0.009	$\underline{125.605}$	0.457 ± 0.066	0.333 ± 0.010	$\underline{0.165 \pm 0.029}$	0.302 ± 0.013
RTA	0.441 ± 0.046	0.325 ± 0.008	145.043	0.648 ± 0.052	0.335 ± 0.010	0.230 ± 0.035	0.315 ± 0.012

A.3 Full results for generalizable unlearning

Table 7, Table 8, and Table 9 compare the two splitting strategies introduced in Section A.1.2. The Low→High and High→Low splits are more challenging, as the intervention and held-out sets differ systematically in their initial memorization level.

The difference is most pronounced in the Low→High setting, where the unlearning algorithm observes only the less strongly memorized half of the forget set and must generalize to the more strongly memorized half. Our method achieves an overall SSCD of 0.177, substantially lower than the strongest baseline, Wen et al. (0.299), while maintaining a good overall CLIP score. It also achieves the lowest FID (117.2 vs. 128.8 for the next-best method). More importantly, on the held-out high-memorization subset $\mathcal{H}$, our method reduces SSCD from 0.866 to 0.267, compared with 0.429 for Wen et al. and substantially higher values for the remaining baselines. This indicates a considerably stronger ability to generalize the unlearning effect to memorized examples that are systematically harder than those observed during intervention.

The High→Low setting exhibits a complementary pattern. Our method again achieves the best overall SSCD (0.121) and FID (119.8) while preserving CLIP (0.292). Since the held-out subset in this setting contains the less strongly memorized examples, several baselines already achieve relatively low SSCD on $\mathcal{H}$, and the advantage of our method on the held-out subset alone is consequently smaller. Nevertheless, our method suppresses memorization consistently across both subsets, achieving SSCDs of 0.091 and 0.152 on $\mathcal{F}$ and $\mathcal{H}$, respectively. In contrast, competing methods exhibit a much larger imbalance between the two subsets. Similar trends can also be found in Table 8 and 9.

Together, these results show that our method achieves the strongest performance not only in the standard unlearning setting, but also in the generalizable unlearning settings. Its advantage is particularly pronounced under the more challenging memorization-stratified splits, especially when generalizing from weakly to strongly memorized examples, where our local representation alignment yields substantially stronger suppression on held-out memorized examples while maintaining competitive generation quality.

We emphasize that CLIP measures text-image alignment only at an aggregate level and can be insensitive to perceptually important differences between individual generations. We therefore use it as a coarse utility constraint rather than as the sole measure of generation quality, and report FID alongside CLIP in the main evaluation. Qualitative examples illustrating this limitation are provided in Appendix A.5.

Table 8: Full generalizable-unlearning results under different intervention/held-out splits for **SD v1.4 - template memorization**. We report mean ± 95% confidence interval for SSCD and CLIP. Overall metrics are evaluated on $\mathcal{F} \cup \mathcal{H}$, while subset-level metrics are evaluated separately on $\mathcal{F}$ and $\mathcal{H}$.

(a) Low → High memorization.

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.671 ± 0.021	0.317 ± 0.004	223.766	0.529 ± 0.024	0.315 ± 0.007	0.813 ± 0.009	0.318 ± 0.005
BLUR (ours)	$\mathbf{0.248 \pm 0.019}$	0.278 ± 0.006	**156.054**	$\mathbf{0.195 \pm 0.020}$	0.269 ± 0.009	$\mathbf{0.301 \pm 0.026}$	0.287 ± 0.007
Fine-tune	0.513 ± 0.024	0.309 ± 0.005	194.960	0.396 ± 0.027	0.310 ± 0.007	0.630 ± 0.028	0.308 ± 0.006
SISS	$\underline{0.312 \pm 0.012}$	0.278 ± 0.005	198.666	$\underline{0.256 \pm 0.014}$	0.281 ± 0.009	$\underline{0.368 \pm 0.017}$	0.275 ± 0.007
NegGrad+	0.521 ± 0.024	0.307 ± 0.005	199.630	0.397 ± 0.027	0.307 ± 0.008	0.646 ± 0.026	0.308 ± 0.006
RTA	0.543 ± 0.024	0.287 ± 0.006	205.745	0.440 ± 0.032	0.281 ± 0.011	0.645 ± 0.024	0.293 ± 0.006
Wen et al.	0.356 ± 0.022	0.286 ± 0.005	$\underline{183.856}$	0.303 ± 0.026	0.294 ± 0.008	0.409 ± 0.035	0.278 ± 0.007

(b) High → Low memorization.

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.671 ± 0.021	0.317 ± 0.004	223.766	0.813 ± 0.009	0.318 ± 0.005	0.529 ± 0.024	0.315 ± 0.007
BLUR (ours)	$\mathbf{0.205 \pm 0.014}$	0.281 ± 0.007	**156.006**	$\mathbf{0.207 \pm 0.020}$	0.279 ± 0.008	$\mathbf{0.203 \pm 0.020}$	0.283 ± 0.010
Fine-tune	0.513 ± 0.024	0.309 ± 0.005	194.960	0.630 ± 0.028	0.308 ± 0.006	0.396 ± 0.027	0.310 ± 0.007
SISS	0.269 ± 0.014	0.273 ± 0.006	192.802	0.242 ± 0.019	0.261 ± 0.007	0.296 ± 0.019	0.285 ± 0.008
NegGrad+	$\underline{0.254 \pm 0.019}$	0.304 ± 0.005	$\underline{158.564}$	$\underline{0.229 \pm 0.028}$	0.303 ± 0.007	0.280 ± 0.025	0.305 ± 0.008
RTA	0.514 ± 0.021	0.307 ± 0.005	208.773	0.587 ± 0.027	0.306 ± 0.006	0.442 ± 0.026	0.308 ± 0.007
Wen et al.	0.307 ± 0.020	0.285 ± 0.005	200.579	0.372 ± 0.028	0.282 ± 0.006	$\underline{0.242 \pm 0.025}$	0.288 ± 0.009

Table 9: Full generalizable-unlearning results under different intervention/held-out splits for **SD v2.0 - template memorization**. We report mean ± 95% confidence interval for SSCD and CLIP. Overall metrics are evaluated on $\mathcal{F} \cup \mathcal{H}$, while subset-level metrics are evaluated separately on $\mathcal{F}$ and $\mathcal{H}$.

(a) Low → High memorization

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.545 ± 0.029	0.307 ± 0.005	218.950	0.366 ± 0.036	0.313 ± 0.008	0.722 ± 0.011	0.302 ± 0.005
BLUR (ours)	$\mathbf{0.121 \pm 0.006}$	0.279 ± 0.009	**139.552**	$\mathbf{0.119 \pm 0.010}$	0.272 ± 0.012	$\mathbf{0.123 \pm 0.007}$	0.286 ± 0.007
Fine-tune	0.499 ± 0.028	0.311 ± 0.005	210.774	0.328 ± 0.035	0.318 ± 0.008	0.668 ± 0.016	0.305 ± 0.005
SISS	0.296 ± 0.014	0.290 ± 0.005	207.295	0.213 ± 0.015	0.295 ± 0.008	0.379 ± 0.011	0.286 ± 0.006
NegGrad+	0.266 ± 0.017	0.285 ± 0.006	$\underline{180.318}$	$\underline{0.166 \pm 0.018}$	0.296 ± 0.009	0.366 ± 0.015	0.273 ± 0.006
RTA	0.381 ± 0.022	0.286 ± 0.005	199.433	0.314 ± 0.033	0.299 ± 0.009	0.447 ± 0.023	0.272 ± 0.006
Wen et al.	$\underline{0.234 \pm 0.017}$	0.275 ± 0.006	300.831	0.224 ± 0.026	0.295 ± 0.009	$\underline{0.244 \pm 0.021}$	0.256 ± 0.006

(b) High → Low memorization

Method	Overall SSCD ↓	Overall CLIP ↑	FID ↓	$\text{SSCD}_{\mathcal{F}}$ ↓	$\text{CLIP}_{\mathcal{F}}$ ↑	$\text{SSCD}_{\mathcal{H}}$ ↓	$\text{CLIP}_{\mathcal{H}}$ ↑
No mitigation	0.545 ± 0.029	0.307 ± 0.005	218.950	0.722 ± 0.011	0.302 ± 0.005	0.366 ± 0.036	0.313 ± 0.008
BLUR (ours)	$\mathbf{0.188 \pm 0.015}$	0.283 ± 0.008	**147.016**	$\mathbf{0.123 \pm 0.007}$	0.258 ± 0.007	$\underline{0.254 \pm 0.025}$	0.308 ± 0.010
Fine-tune	0.499 ± 0.028	0.311 ± 0.005	210.774	0.668 ± 0.016	0.305 ± 0.005	0.328 ± 0.035	0.318 ± 0.008
SISS	0.290 ± 0.011	0.276 ± 0.005	185.820	0.324 ± 0.009	0.259 ± 0.005	0.255 ± 0.019	0.293 ± 0.008
NegGrad+	0.382 ± 0.019	0.318 ± 0.005	$\underline{180.821}$	0.478 ± 0.016	0.317 ± 0.005	0.284 ± 0.025	0.320 ± 0.009
RTA	0.379 ± 0.022	0.275 ± 0.006	247.694	0.465 ± 0.023	0.253 ± 0.005	0.292 ± 0.030	0.296 ± 0.009
Wen et al.	$\underline{0.279 \pm 0.017}$	0.298 ± 0.006	292.290	$\underline{0.319 \pm 0.022}$	0.281 ± 0.006	$\mathbf{0.238 \pm 0.023}$	0.315 ± 0.009

A.4 Memorization-utility trade-off at matched SSCD

Our main evaluation compares methods under a minimum acceptable CLIP range, asking how much memorization can be removed without substantial degradation in generation quality. Here, we consider the complementary evaluation: we ask how much generation utility competing methods retain when pushed toward the memorization level achieved by BLUR.

For each setting, we take the overall SSCD achieved by BLUR as the target and perform an extended, more aggressive hyperparameter sweep for each baseline. We report CLIP and FID when a baseline reaches approximately the same SSCD. A dash indicates that no configuration in our extended sweep reached the target SSCD. Importantly, this evaluation allows configurations with severe utility degradation; its purpose is to expose the cost required to obtain comparable memorization reduction rather than to identify a practically desirable operating point.

Table 10: **Memorization-utility trade-off at matched SSCD.** For each setting, we use BLUR's overall SSCD as the target and aggressively tune competing methods toward the same memorization level. CLIP and FID are reported at the approximately matched operating point. "—" indicates that no configuration in our extended sweep reached the target SSCD.

(a) SD v1.4 – Verbatim memorization

Method	**Overall SSCD ↓**	**Overall CLIP ↑**	**FID ↓**
BLUR (ours)	0.177	0.290	117.232
Fine-tune	—	—	—
NegGrad+	0.077	0.131	316.0
SISS	—	—	—
RTA	—	—	—
Wen et al.	—	—	—

(b) SD v1.4 – Template memorization

Method	**Overall SSCD ↓**	**Overall CLIP ↑**	**FID ↓**
BLUR (ours)	0.248	0.278	156.054
Fine-tune	—	—	—
NegGrad+	0.075	0.134	313.3
SISS	0.070	0.138	315.2
RTA	—	—	—
Wen et al.	0.231	0.265	197.578

Table 10 shows that even when utility preservation is removed as a constraint, most baselines cannot reach the memorization level achieved by BLUR. NegGrad+ and SISS can be pushed toward comparable SSCD only under substantially more aggressive configurations, at which point generation utility deteriorates sharply, as reflected by CLIP and FID. Together with the CLIP-constrained evaluation in the main paper, these results show that BLUR achieves a more favorable memorization-utility trade-off rather than reducing SSCD simply by degrading generation quality.

A.5 Qualitative Examples

We provide qualitative generations for the three evaluation settings: *SD v1.4 - verbatim memorization*, *SD v1.4 - template memorization*, and *SD v2.0 - template memorization*. Figure 3 compares generations from the original model, existing baselines, and our method. Across the three settings, our method substantially reduces visual similarity to the memorized reference while preserving plausible generations that remain consistent with the input prompt.

The examples in Figure 4 also illustrate why we use CLIP primarily as a *coarse guardrail* against severe degradation rather than as a fine-grained measure of perceptual generation quality. Small differences in average CLIP score within the acceptable region do not necessarily correspond to visible improvements in the generated images: methods with slightly higher CLIP can produce samples of comparable or worse perceptual quality. Our evaluation therefore constrains methods to an acceptable CLIP region and compares how much memorization reduction can be achieved within this region. Under this criterion, our method achieves substantially lower SSCD while maintaining qualitatively comparable generation quality.

As a complementary analysis, Appendix A.4 instead matches methods at similar SSCD levels. There, forcing competing methods toward the memorization levels achieved by our method exposes a substantially larger cost in CLIP and/or FID, further showing the trade-off between memorization removal and generation quality.

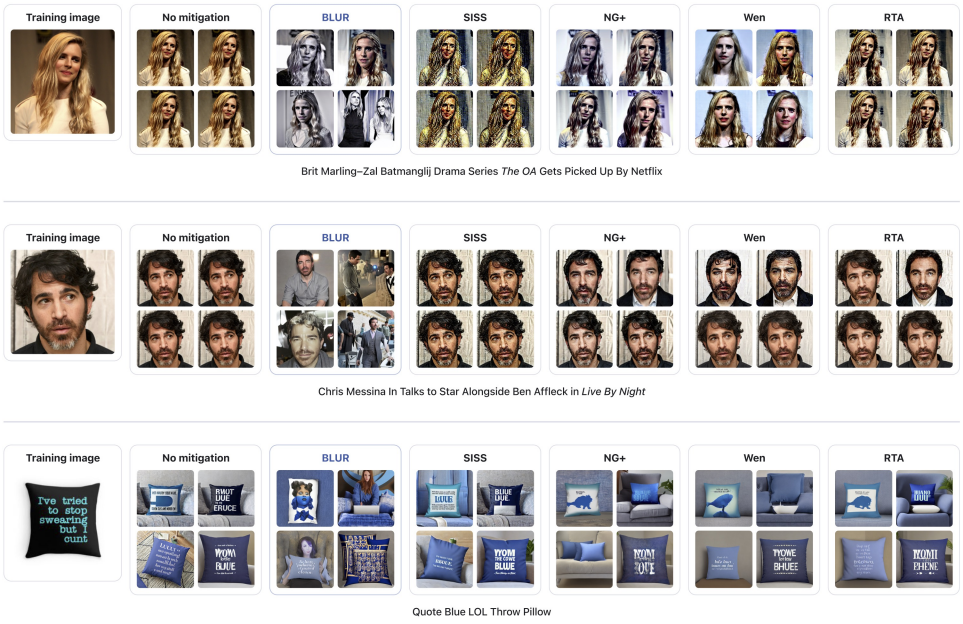


Figure 3: Qualitative generation examples from different unlearning methods.

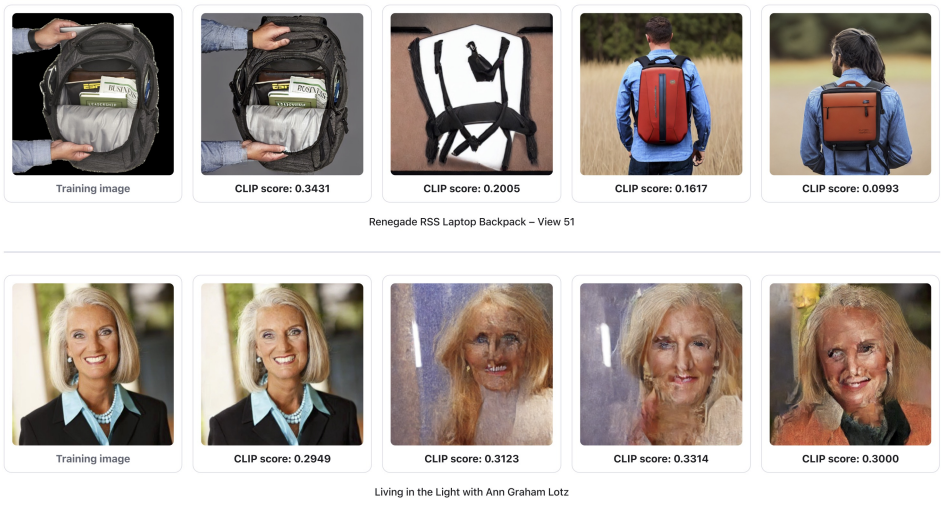


Figure 4: Qualitative generation examples to show that CLIP provides only a coarse measure of generation quality. Although the outputs are produced from the same prompt, their CLIP scores do not consistently reflect visual quality or prompt fidelity. In the bottom example, severely distorted generations receive higher CLIP scores than the near-verbatim output, while in the top example, plausible backpack generations receive lower scores than a malformed image.