

Memories of Forgotten Concepts

Supplementary Material

In the next sections, we provide additional details and results that further support our analysis and offer a more comprehensive understanding of the findings we presented in the main paper. Appx. A provides additional information on how the likelihood of latents affects image generation. Appx. B explains our initialization choice when searching for distant latents that can generate a given query image I_q . Appx. C provides additional results for the experiments shown in the paper, including metrics that were not discussed, such as CLIP-score and a concept detector accuracy. Appx. D contains an analysis of the distribution of the NLL of multivariate normally distribute vectors. Appx. E contains results that justify our choice of inversion method and its parameters.

A. Likelihood effect on generation

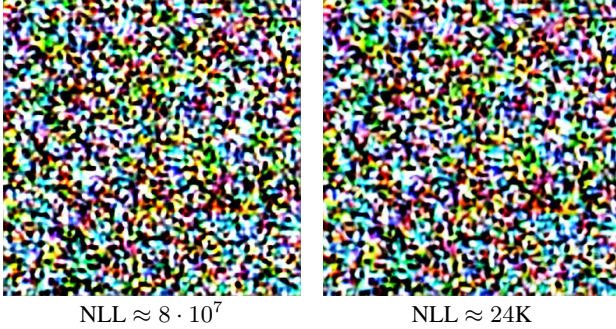


Figure 12. **Inversion of low likelihood images.** A low likelihood latent can be used to generate an image (left). The image can be inverted to find a latent that generates a similar image (right), with PSNR=19.64[dB].

In this section, we focus on further examining the effect of the likelihood of z_T on the generated image by a given diffusion model. As explained in Sec. 4, inversion is a powerful tool that can be used to generate images with different likelihoods. But, examining these generated images along with reconstruction error can give more information. For example, in Fig. 12, we see that while inversion is used to transform a *very unlikely* image to an image with reasonable likelihood, the reconstruction PSNR of this process is poor.

We are also interested in the relation between likelihood to generation quality. As shown on Fig. 13, while the $\vec{0}$ vector has the lowest (best) NLL, its generation quality is poor. This is due to the fact that the model was trained using random samples from the standard normal distribution, and (with high probability) have not been given the $\vec{0}$ as in-


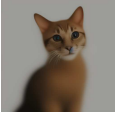


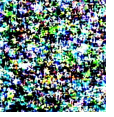
NLL α	15K 0	20K 100	23.2K 128	26.3K 150	31.2K 180
					

Figure 13. **Generation using latents with varying likelihoods:** Using the same caption “cat”, the likelihood of the initial latent seed controls the generation quality. This is done by sampling $z \sim \mathcal{N}(0, 1)$ and applying $Y = \alpha \cdot \frac{z}{\|z\|}$ (i.e., using the same vector with a scaled norm of α).

put for generation. This coincides with the work of Samuel *et al.* [37], demonstrating that diffusion models are learned using latents with a specific norm range. This conclusion is important for our analysis, as we do not use the NLL as an absolute score, but rather as a relative score compared to the NLL of the standard normal distribution (see Eq. (5)).

B. Different initializations for distant memories retrieval

In Sec. 3.3, we suggest applying our sequential inversion block (SIB), starting from arbitrary support images to retrieve distant memories of a given ablated target image. Next, we present a few straightforward alternatives and discuss their drawbacks.

Instead of performing a VAE decoder inversion, one could suggest utilizing the encoder of the VAE. Given an image I_q , the encoder returns parameters for a normal distribution, i.e., $\text{Enc}(I_q) = \mathcal{N}(\mu_{I_q}, \Sigma_{I_q})$. The distribution can be used to sample multiple different latents, in close proximity. In Sec. 3, we do not sample multiple latents, but rather we use a latent z_0 which is the mean of the distribution, μ_{I_q} . Fig. 14 shows the PSNR and distances results for these latents. The reconstruction quality is high, but all memories turn out the same, displaying an average pairwise cosine distance of 0.

In Fig. 15, in order to examine the case of more distant latents, we sample from $\mathcal{N}(\mu_{I_q}, \Sigma_{I_q})$ but add a standard normal random noise (normalized across its channels dimension) and scaled by a factor of 10. As can be seen, the average pairwise cosine distance is much higher and resembles the average pairwise cosine distance presented by our solution in Fig. 10. However, the PSNR is considerably lower, suggesting that this method did not reconstruct images that resemble I_q .

Finally, we show in Fig. 16, that applying SIB, starting

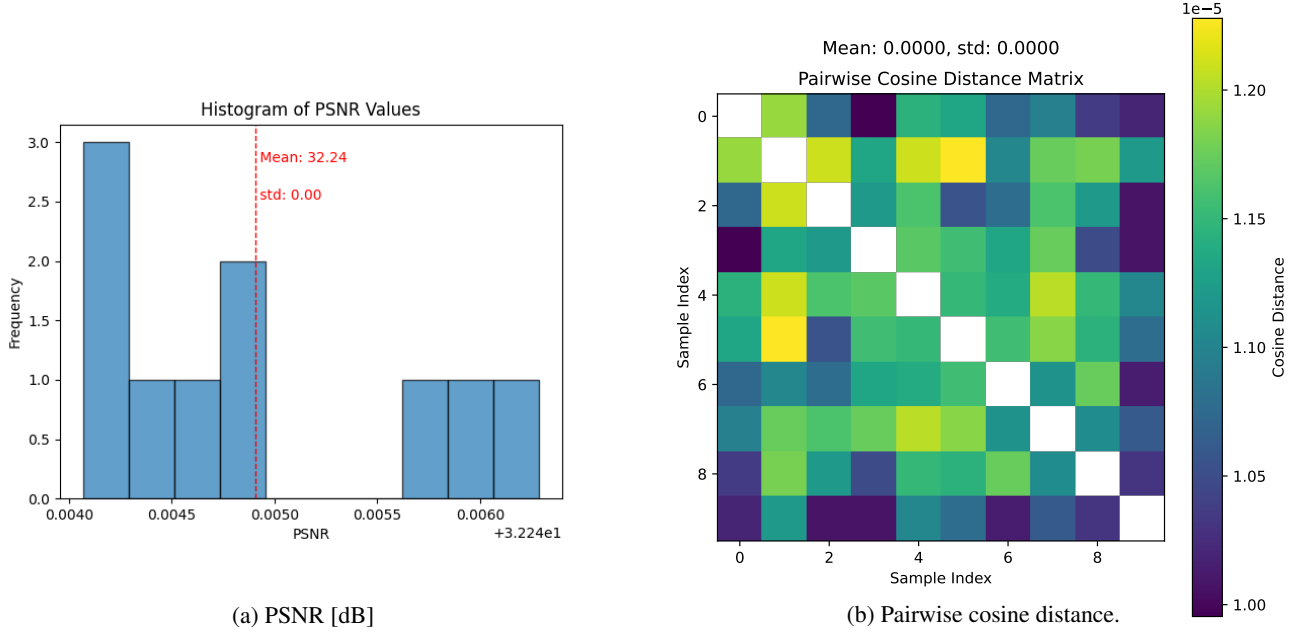


Figure 14. **Sample near:** randomly sample 10 latents from $\mathcal{N}(\mu = \text{Enc}(I_q), \Sigma_{I_q})$.

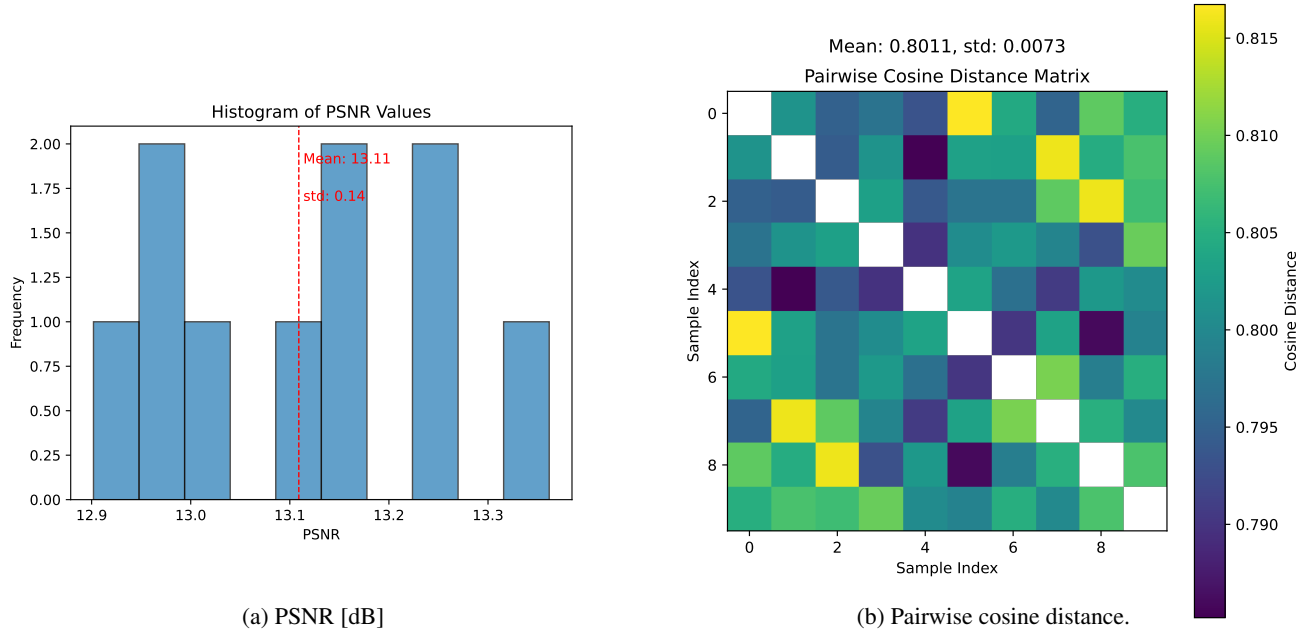


Figure 15. **Sample far:** randomly sample 10 latents from $\mathcal{N}(\mu = \text{Enc}(I_q), \Sigma_{I_q})$. For each sample, add a random noise.

from randomly sampled latents, is also suboptimal. Although the reconstruction quality is sufficient, the average pairwise cosine distance is lower than our suggested method (See Fig. 10).

We conclude that trivial random initializations are

suboptimal, and are inferior compared to our method in Sec. 3.3.

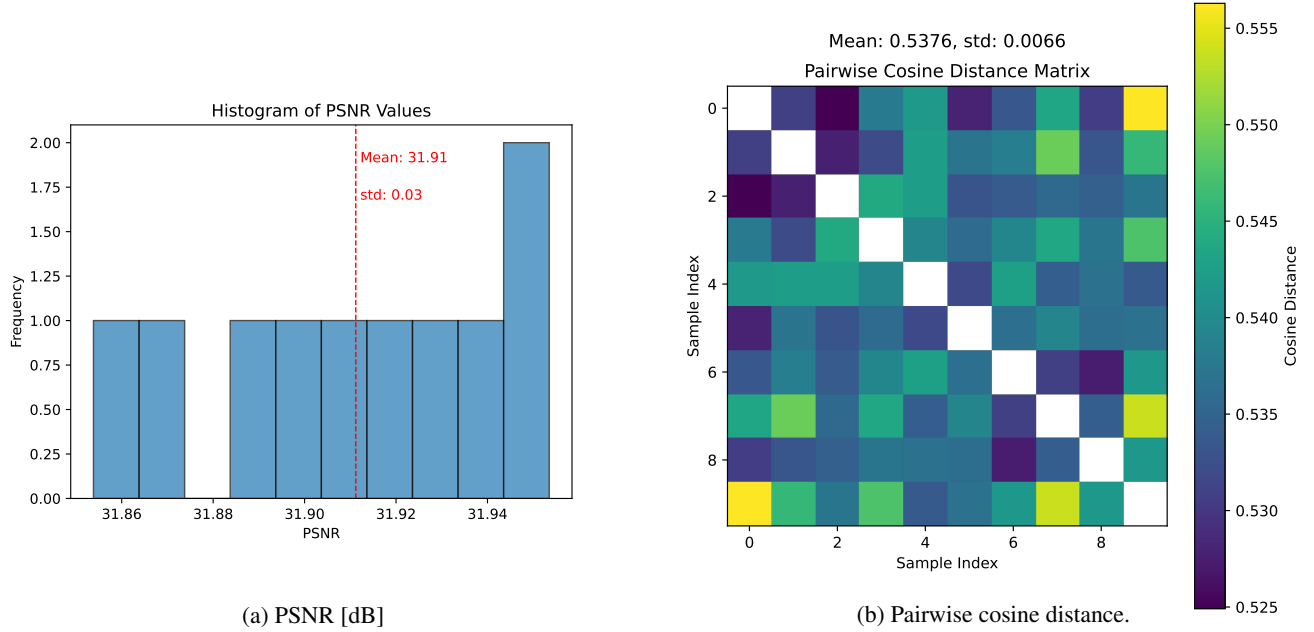


Figure 16. **SIB on random noise:** randomly sample 10 initializations for SIB.

C. Further analysis

Next, we present additional analysis regarding the experiments in Sec. 3.3. Specifically, Tabs. 1 to 6 and Tabs. 7 to 12 contain extended results for the experiments detailed in Secs. 3.2 and 3.3 (and visualized in Figs. 5 and 9), respectively. Each table contains results that correspond to one concept. These tables contain the scores discussed in the main paper, *i.e.*, PSNR and $d_{\mathcal{N}}(E, R)$, along with a concept classifier detection score, CLIP-score and the EMD between different distributions. The presented EMD results are:

1. E, \mathcal{N} — The EMD between the NLL of latents in the erased set E and the NLL of standard normal samples, *i.e.*, $\text{EMD}(\text{NLL}_{\rightarrow Z_T}(E), \text{NLL}(\mathcal{N}))$.
2. R, \mathcal{N} — The same as above, using latents from the reference set R , *i.e.*, $\text{EMD}(\text{NLL}_{\rightarrow Z_T}(R), \text{NLL}(\mathcal{N}))$.
3. E, R — The EMD between latents in the erased and reference sets, *i.e.*, $\text{EMD}(\text{NLL}_{\rightarrow Z_T}(E), \text{NLL}_{\rightarrow Z_T}(R))$.

Items 1 and 2 serve as the numerator and denominator of $d_{\mathcal{N}}(\cdot, \cdot)$ (see Eq. (5)), respectively. Tabs. 7 to 12 also contain the average distances between all $z_T^{(s_i \rightarrow q)}$ and z_T^q (see Fig. 10). These values are shown in both euclidean distance and cosine similarity.

Fig. 17 contains a full comparison of Fig. 6 for all available erasing methods and concepts.

D. What is the distribution of the NLL of a Normal Random Vector?

As we described in Sec. 3.1, we use the Negative-Log-Likelihood (NLL) to analyze latents w.r.t. normal distributions. Recall that as explained in Sec. 2.2, the distribution that was used to train the model is multivariate standard normal, *i.e.*, with i.i.d. components. Next, we present why in our case we can treat this distribution as Gaussian.

For a multivariate random vector $Z \in \mathbb{R}^k \sim \mathcal{N}(\vec{\mu}, \Sigma)$ with i.i.d. variables $Z_i \in \mathbb{R}$, its Probability Density Function (PDF) is:

$$p_Z(Z) = (2\pi)^{-\frac{k}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(Z - \mu)^T \Sigma^{-1}(Z - \mu)\right). \quad (8)$$

In our case, all the i.i.d. univariate Gaussians have the same parameters, meaning that $\forall i : Z_i \sim \mathcal{N}(\mu, \sigma^2)$. Thus, the NLL of Z , which is $-\log p_Z(Z)$, can be expressed as:

$$\text{NLL}(Z) = \frac{k}{2} \log(2\pi\sigma^2) + \frac{1}{2\sigma^2} \sum_{i=1}^k (Z_i - \mu)^2. \quad (9)$$

We denote $Y_i = \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2\sigma^2} (Z_i - \mu)^2$ and we get that:

$$\text{NLL}(Z) = \sum_{i=1}^k Y_i. \quad (10)$$

	Detection (%)	PSNR [dB]	EMD			$d_{\mathcal{N}}(E, R)$	CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R		
EraseDiff [45]	100	34.13	1565.2K	2896.0K	269.1K	0.54	0.31
ESD [7]	100	34.24	281.7K	194.9K	15.6K	1.45	0.31
FMN [46]	100	34.23	249.0K	185.9K	14.4K	1.34	0.31
Salun [4]	100	34.05	408.5K	294.4K	13.9K	1.39	0.31
Scissorhands [44]	100	34.32	2090.3K	2603.4K	67.1K	0.80	0.32
SPM [23]	100	34.22	257.3K	182.3K	14.0K	1.41	0.31
UCE [8]	100	34.22	263.8K	192.6K	13.3K	1.37	0.31
AdvUnlearn [48]	100	34.21	258.1K	192.1K	12.2K	1.34	0.31
Vanilla [33]	100	34.21	287.0K	210.1K	14.0K	1.37	0.31

Table 1. Ablated concept: Nudity.

	Detection (%)	PSNR [dB]	EMD			$d_{\mathcal{N}}(E, R)$	CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R		
EraseDiff [45]	96	32.44	521.6K	256.2K	47.3K	2.04	0.32
ESD [7]	98	32.47	386.0K	159.1K	49.5K	2.43	0.32
FMN [46]	96	32.33	507.6K	238.9K	50.5K	2.12	0.32
Salun [4]	96	32.50	631.1K	356.7K	38.9K	1.77	0.32
Scissorhands [44]	94	32.48	484.0K	217.7K	52.5K	2.22	0.32
SPM [23]	96	32.46	436.9K	201.4K	45.5K	2.17	0.32
AdvUnlearn [48]	96	32.29	439.6K	204.8K	44.5K	2.15	0.32
Vanilla [33]	96	32.47	453.1K	209.1K	47.0K	2.17	0.32

Table 3. Ablated concept: Parachute.

The expectation of Y_i is:

$$\begin{aligned}\mathbb{E}[Y_i] &= \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2\sigma^2} \mathbb{E}[(Z_i - \mu_i)^2] \\ &= \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2}.\end{aligned}\quad (11)$$

To compute the variance of Y_i we first compute $\mathbb{E}[Y_i^2]$, denoting $C = \frac{1}{2} \log(2\pi\sigma^2)$ for short:

$$\begin{aligned}\mathbb{E}[Y_i^2] &= C^2 + \frac{C}{\sigma^2} \text{Var}(Z_i) + \frac{1}{4\sigma^4} \mathbb{E}[(Z_i - \mu)^4] \\ &= C^2 + C + \frac{3}{4},\end{aligned}\quad (12)$$

where in $=^1$ we use the definition of the 4th central moment for normal distribution. Using all the above, we can compute $\text{Var}(Y_i) = \mathbb{E}[Y_i^2] - \mathbb{E}[Y_i]^2$:

$$\text{Var}(Y_i) = C^2 + C + \frac{3}{4} - (C^2 + C + \frac{1}{4}) = 0.5 \quad (13)$$

When k is large, the sum $\sum_{i=1}^k Y_i$ can be approximated by a normal distribution due to the Central Limit Theorem (CLT). We use this assumption in our case, as the latent dimension of our vectors is $4 \times 4 \times 64 \approx 16K$. Specifically, for standard normal distribution, we get:

$$\mathbb{E}[Y_i] \approx 1.42. \quad (14)$$

This means that for $\text{NLL}(Z)$ we assume:

$$\text{NLL}(Z) \sim \mathcal{N}(1.42k, 0.5k) \approx \mathcal{N}(23.3K, 8192) \quad (15)$$

	Detection (%)	PSNR [dB]	EMD			$d_{\mathcal{N}}(E, R)$	CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R		
EraseDiff [45]	96	28.73	434.5K	229.8K	34.2K	1.89	0.31
ESD [7]	96	28.76	347.7K	173.9K	33.7K	2.00	0.31
FMN [46]	96	28.75	436.4K	223.2K	37.0K	1.96	0.31
Salun [4]	96	28.74	568.1K	318.5K	38.5K	1.78	0.31
Scissorhands [44]	96	28.71	402.9K	185.5K	42.9K	2.17	0.31
SPM [23]	96	28.74	413.7K	205.9K	37.1K	2.01	0.31
AdvUnlearn [48]	96	28.74	410.7K	215.6K	33.4K	1.90	0.31
Vanilla [33]	96	28.71	413.7K	201.0K	39.4K	2.06	0.31

Table 2. Ablated concept: Church.

	Detection (%)	PSNR [dB]	EMD			$d_{\mathcal{N}}(E, R)$	CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R		
EraseDiff [45]	76	31.41	125.6K	269.0K	41.4K	0.47	0.31
ESD [7]	78	31.41	86.5K	206.7K	53.1K	0.42	0.31
FMN [46]	78	31.41	130.1K	245.2K	27.7K	0.53	0.31
Salun [4]	76	31.41	167.2K	341.2K	40.8K	0.49	0.31
Scissorhands [44]	76	31.41	264.7K	256.4K	8.1K	1.03	0.31
SPM [23]	78	31.41	108.4K	214.8K	29.4K	0.50	0.31
AdvUnlearn [48]	78	31.40	118.2K	229.3K	32.3K	0.52	0.31
Vanilla [33]	78	31.41	111.0K	212.7K	26.1K	0.52	0.31

Table 4. Ablated concept: Tench.

The \mathcal{N} distribution in Fig. 3 is an example of the NLL for standard normal samples, along with other different normal distributions.

E. Inversion parameters and generalization to different DiT based architectures

In this section we describe different aspects for our choice of inversion method, along with its chosen parameters. In addition, we demonstrate how our model generalizes to DiT based architectures. As explained in Sec. 3, we use Renoise [10] as our inversion method. Fig. 18 shows the effect of the number of renoising steps, the number of internal optimization iterations between the scheduler steps, on the likelihood of the output latent. In our experiments, using 10 renoising steps results in lower PSNR values (e.g., 16.9 dB between the church image in the left panel of Fig. 10a and its reconstruction), although the likelihoods are low. To utilize Renoise for our analysis, we use 5 iterations, which results in a reconstruction with low likelihoods, and high PSNR (e.g. 26.3 dB between the church image in the left panel of Fig. 10a and its reconstruction).

Furthermore, we perform our analysis using an additional inversion method, Null Text Inversion (NTI) [26]. This method optimizes the textual embeddings of the null text, in order to achieve a more consistent inverse image. We demonstrate concept-level retrieval on a handful of erasure methods using NTI (Tab. 13) instead of Renoise (Tab. 1). A drop in PSNR values can be attributed to the inversion superiority of Renoise when compared to NTI.

	Detection (%)	PSNR [dB]	EMD				CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R	$d_{\mathcal{N}}(E, R)$	
EraseDiff [45]	80	28.98	447.3K	261.1K	25.0K	1.71	0.30
ESD [7]	84	28.93	369.9K	199.3K	26.3K	1.86	0.30
FMN [46]	82	29.00	447.0K	244.1K	30.5K	1.83	0.30
Salun [4]	82	28.98	618.6K	362.4K	34.1K	1.71	0.30
Scissorhands [44]	86	29.01	298.0K	283.1K	11.2K	1.05	0.30
SPM [23]	82	28.95	374.0K	193.2K	29.6K	1.94	0.30
AdvUnlearn [48]	82	28.99	370.2K	201.6K	25.5K	1.84	0.30
Vanilla [33]	82	28.97	421.7K	212.7K	35.4K	1.98	0.30

Table 5. Ablated concept: Garbage Truck.

	Detection (%)	PSNR[dB]	EMD				CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R	$d_{\mathcal{N}}(E, R)$			
EraseDiff [45]	100	30.32	3187.1K	3844.1K	145.2K	0.83	0.28	0.72	169.20
ESD [7]	98	30.04	45.4K	96.7K	135.6K	0.47	0.28	0.78	159.92
FMN [46]	100	29.59	39.8K	89.2K	131.5K	0.45	0.28	0.79	160.81
Salun [4]	100	29.22	45.8K	114.6K	152.8K	0.40	0.28	0.77	159.04
Scissorhands [44]	100	30.34	3429.5K	3557.9K	166.9K	0.96	0.28	0.76	173.58
SPM [23]	100	29.29	37.8K	84.9K	122.1K	0.45	0.28	0.79	160.73
UCE [8]	100	29.66	34.6K	82.3K	120.3K	0.42	0.28	0.78	160.50
AdvUnlearn [48]	100	28.74	28.8K	84.8K	107.0K	0.34	0.28	0.78	160.16
Vanilla [33]	100	29.49	39.8K	83.8K	126.3K	0.47	0.28	0.79	160.77

Table 7. Ablated Images: Nudity.

However, our analysis holds when NTI is used as well.

Previous erasure methods and benchmarks [27, 47, 49] have focused exclusively on SD1.4. However, we extend our analysis to DiT based architectures, specifically using Flux¹. We utilized a Flux adaptation of UCE [8] according to EraseAnything [9]. To handle DiT based models, we utilized RF-Inversion [34]. This extends our explored inversion methods to DiT based models, for a total of 3 methods: Null Text Inversion, Renoise and RF-Inversion. To establish a benchmark on Flux, we run our analysis on the vanilla Flux model. Moreover, we apply a Flux adaptation to UCE and erase two different concepts. Our analysis in Tab. 14 shows high PSNR (> 34 dB) and low $d_{\mathcal{N}}(\cdot, \cdot)$ values (below 0.8), indicating the concepts remain as likely as, or even more likely than, the reference set. These findings align with EraseAnything’s conclusion that the Flux adaptation of UCE is ineffective for concept erasure.

These results reinforce our overarching conclusion about the limitations of current erasure methods.

	Detection (%)	PSNR [dB]	EMD				CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R	$d_{\mathcal{N}}(E, R)$	
ESD [7]	88	27.27	369.0K	208.1K	850.2K	1.77	0.34
FMN [46]	88	27.46	292.1K	246.8K	792.6K	1.18	0.34
SPM [23]	88	27.59	293.1K	213.6K	740.7K	1.37	0.34
UCE [8]	88	27.56	304.0K	201.1K	756.2K	1.51	0.34
AC [20]	88	27.60	358.4K	211.1K	837.2K	1.70	0.34
AdvUnlearn [48]	88	27.17	299.6K	195.0K	738.8K	1.54	0.34
Vanilla [33]	88	27.61	333.8K	189.5K	751.1K	1.76	0.34

Table 6. Ablated concept: Van Gogh.

	Detection (%)	PSNR [dB]	EMD				CLIP-Score
			E, \mathcal{N}	R, \mathcal{N}	E, R	$d_{\mathcal{N}}(E, R)$	
FMN [46]	99	30.16	746.3K	672.9K	7.9K	1.11	0.31
Salun [4]	86	26.98	963.9K	828.5K	8.9K	1.16	0.30
Scissorhands [44]	97	28.92	932.5K	1202.3K	45.9K	0.78	0.31
UCE [8]	98	30.02	725.9K	629.6K	8.0K	1.15	0.31
Vanilla [33]	99	30.08	718.0K	622.1K	8.4K	1.15	0.32

Table 13. NTI [26] Ablated concept: Nudity.

F. How many solutions exist?

In our effort to truly erase an image, we explored how many distant memories exist for a single image. To this end, we significantly increased the number of retrieved latents. As shown in Fig. 19, we identified 1,000 distant likely latents that successfully reconstruct an image of a Garbage Truck, in an ESD model that erased this concept. Their mean pairwise cosine distance is 0.71, and the standard deviation is 0.02, comparable to the mean distance for 10 latents in Fig. 10. We limited our analysis to 1,000 latents but suspect the actual number is higher. Thus, as raised in the geometric interpretation of the retrieved memories in Sec. 3.3, truly forgetting an image remains challenging, highlighting the persistence of distant memories of supposedly forgotten concepts. We encourage future work to adopt our analysis as a benchmark for single-image erasure, advancing broader concept erasure.

¹<https://github.com/black-forest-labs/flux>

	Detection (%)	PSNR[dB]	EMD			$d_N(E, R)$	CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R				
EraseDiff [45]	90	23.29	167.6K	136.7K	477.0K	1.23	0.31	0.62	145.48
ESD [7]	86	23.08	456.1K	102.0K	641.8K	4.47	0.31	0.62	147.88
FMN [46]	86	23.15	261.6K	97.9K	509.1K	2.67	0.31	0.62	146.99
Salun [4]	86	23.10	140.7K	146.7K	463.4K	0.96	0.31	0.62	145.61
Scissorhands [44]	94	23.51	194.1K	73.4K	341.4K	2.64	0.31	0.62	145.07
SPM [23]	86	22.92	251.6K	84.0K	469.3K	2.99	0.31	0.62	147.23
AdvUnlearn [48]	88	22.99	255.0K	90.6K	475.3K	2.81	0.31	0.62	146.97
Vanilla [33]	88	22.96	257.9K	87.6K	471.0K	2.94	0.31	0.63	147.31

Table 8. Ablated Images: Church.

	Detection (%)	PSNR[dB]	EMD			$d_N(E, R)$	CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R				
EraseDiff [45]	86	28.83	33.3K	141.0K	79.3K	0.24	0.31	0.80	160.99
ESD [7]	80	28.23	25.7K	79.2K	64.7K	0.32	0.30	0.80	162.28
FMN [46]	84	28.36	20.0K	98.0K	78.4K	0.20	0.30	0.80	162.29
Salun [4]	84	28.83	57.6K	179.0K	66.7K	0.32	0.31	0.80	159.92
Scissorhands [44]	82	29.43	23.1K	67.0K	21.7K	0.34	0.31	0.79	159.62
SPM [23]	84	27.85	21.3K	91.5K	70.8K	0.23	0.31	0.80	162.42
AdvUnlearn [48]	74	27.60	20.7K	86.3K	65.0K	0.24	0.30	0.80	162.29
Vanilla [33]	84	28.06	20.7K	87.7K	74.8K	0.24	0.31	0.80	162.55

Table 9. Ablated Images: Parachute.

	Detection (%)	PSNR[dB]	EMD			$d_N(E, R)$	CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R				
EraseDiff [45]	58	28.18	88.9K	129.1K	167.8K	0.69	0.33	0.74	155.13
ESD [7]	46	27.74	140.0K	87.3K	212.0K	1.60	0.33	0.74	156.84
FMN [46]	58	27.57	127.8K	96.5K	205.7K	1.32	0.33	0.74	156.63
Salun [4]	50	28.03	89.0K	161.5K	149.5K	0.55	0.33	0.74	154.42
Scissorhands [44]	58	28.34	138.2K	64.8K	139.6K	2.13	0.33	0.73	154.10
SPM [23]	54	27.40	119.8K	88.5K	192.3K	1.35	0.33	0.74	156.90
AdvUnlearn [48]	42	27.07	116.7K	82.3K	185.1K	1.42	0.33	0.74	156.42
Vanilla [33]	50	27.41	122.2K	83.9K	192.4K	1.46	0.32	0.74	156.97

Table 10. Ablated Images: Tench.

	Detection (%)	PSNR[dB]	EMD			$d_N(E, R)$	CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R				
EraseDiff [45]	86	24.07	41.0K	113.6K	218.2K	0.36	0.29	0.71	153.61
ESD [7]	78	23.42	136.5K	90.7K	276.2K	1.50	0.29	0.71	155.38
FMN [46]	76	24.00	101.4K	98.7K	272.2K	1.03	0.29	0.71	155.56
Salun [4]	76	24.06	9.3K	166.1K	208.8K	0.06	0.29	0.71	152.98
Scissorhands [44]	80	24.24	58.2K	124.0K	241.7K	0.47	0.29	0.69	152.15
SPM [23]	80	23.28	91.0K	80.5K	243.3K	1.13	0.29	0.71	155.43
AdvUnlearn [48]	78	22.91	95.3K	91.8K	257.0K	1.04	0.29	0.71	155.34
Vanilla [33]	76	23.84	92.4K	89.7K	244.6K	1.03	0.29	0.71	155.65

Table 11. Ablated Images: Garbage Truck.

	Detection (%)	PSNR[dB]	EMD			$d_N(E, R)$	CLIP-Score	Cosine distance	Euclidean distance
			E, \mathcal{N}	R, \mathcal{N}	E, R				
ESD [7]	90	22.84	591.0K	89.9K	870.8K	6.57	0.32	0.61	147.20
FMN [46]	92	23.30	424.6K	94.1K	746.6K	4.51	0.32	0.61	146.73
SPM [23]	90	23.13	419.8K	86.6K	702.0K	4.85	0.32	0.61	146.76
UCE [8]	94	23.09	452.9K	87.4K	741.9K	5.18	0.32	0.61	146.87
AC [20]	92	23.28	545.8K	89.9K	837.2K	6.07	0.32	0.61	147.23
AdvUnlearn [48]	90	23.03	481.6K	83.2K	759.6K	5.79	0.32	0.61	146.99
Vanilla [33]	94	23.22	421.4K	86.2K	700.0K	4.89	0.32	0.61	146.89

Table 12. Ablated Images: Van Gogh.



Figure 17. Erased concepts generations. Arbitrary latents vs. our retrieved latents for different ablating methods.

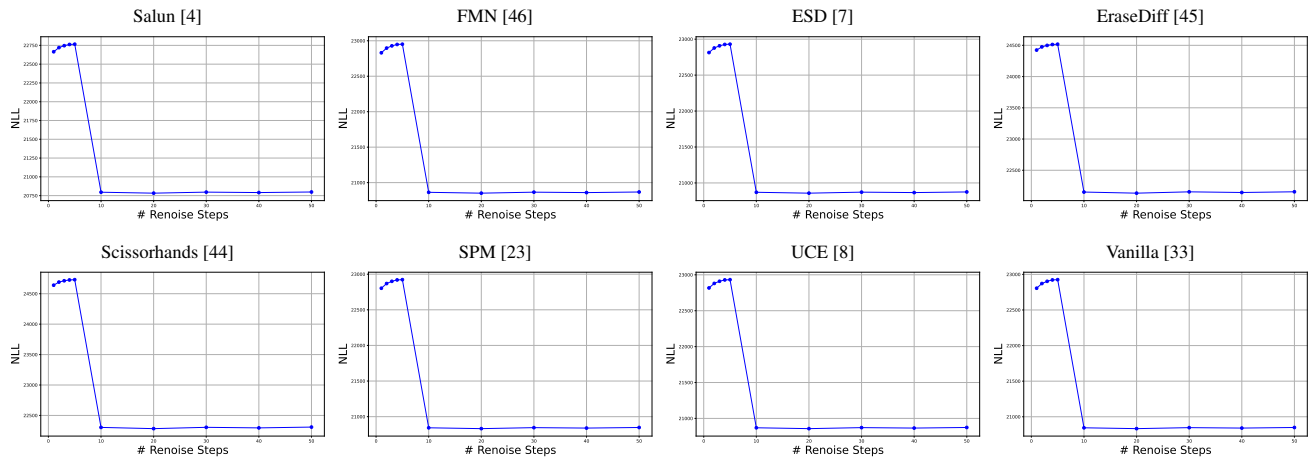


Figure 18. **Choosing the right renoising parameter.** Using Renoise [10], we see that after a certain amount of iterations, the NLL drops dramatically, making it harder to perform a likelihood analysis.

Concept	Vanilla		UCE [8]	
	PSNR[dB]	$d_{\mathcal{N}}(E, R)$	PSNR[dB]	$d_{\mathcal{N}}(E, R)$
Nudity	35.77	0.76	35.76	0.57
Parachute	34.02	0.59	34.01	0.75

Table 14. **Our analysis on Flux using RF-Inversion.**

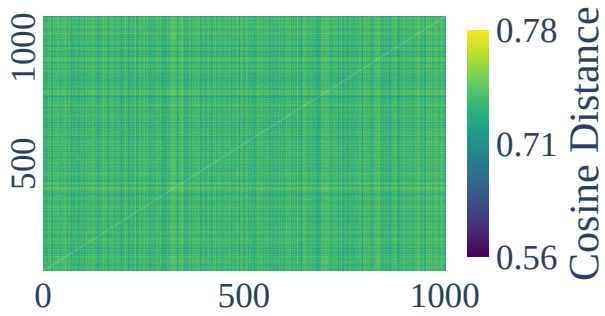


Figure 19. **Erasing is a challenging task.** We produce 1K distant memories of a *single* image. The mean pair-wise cosine distance between the latents is 0.71, and minimal distance is 0.56.